MU887002A
送受信テストモジュール
取扱説明書

第7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，MT8870A ユニバーサルワイヤレステストセット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

管理番号：M-W4048AW-7.0

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。記述内容を十分に理解した上で機器を操作してください。
下記の表示およびシンボルは，そのすべてが本製品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があることを示します。
⚠ 警告	回避しなければ，死亡または重傷に至るおそれがある潜在的な危険があることを示します。
⚠ 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おそ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があることを示します。

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，安全上および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に理解して，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。

MU887002A
送受信テストモジュール
取扱説明書

2020年（令和２年）８月20日（初　版）
2023年（令和５年）２月14日（第７版）

・予告なしに本書の製品操作・取り扱いに関する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 © 2020-2023, ANRITSU CORPORATION
Printed in Japan

安全にお使いいただくために

⚠ 警告



・ 左のアラートマークを表示した箇所の操作をするときは，必ず取扱説明書を参照してください。取扱説明書を読まないで操作などを行った場合は，負傷するおそれがあります。また，本器の特性劣化の原因にもなります。なお，このアラートマークは，危険を示すほかのマークや文言と共に用いられることもあります。

校正



・ 機器本体またはユニットには，出荷時の品質を保持するために性能保証シールが貼られています。このシールは，所定の訓練を受け，火災や感電事故などの危険を熟知した当社または当社代理店のサービスエンジニアによってのみ開封されます。お客様自身で機器本体またはユニットを開け，性能保証シールを破損しないよう注意してください。第三者によってシールが開封，破損されると機器の性能保証を維持できないおそれがあると判断される場合があります。

⚠ 注意

住宅環境での使用について

・ 本器は，工業環境用に設計されています。住宅環境で使用すると，無線障害を起こすことがあります。その場合，使用者は適切な対策を施す必要が生じます。

品質証明

アンリツ株式会社は，本製品が出荷時の検査により公表規格を満足していること，ならびにそれらの検査には，産業技術総合研究所（National Institute of Advanced Industrial Science and Technology）および情報通信研究機構（National Institute of Information and Communications Technology）などの国立研究所によって認められた公的校正機関にトレーサブルな標準器を基準として校正した測定器を使用したことを証明します。

保証

アンリツ株式会社は，納入後 1 年以内に製造上の原因に基づく故障が発生した場合は，無償で修復することを保証します。
ソフトウェアの保証内容は別途「ソフトウェア使用許諾書」に基づきます。
ただし，次のような場合は上記保証の対象外とさせていただきます。

- この取扱説明書に別途記載されている保証対象外に該当する故障の場合。
- お客様の誤操作，誤使用または無断の改造もしくは修理による故障の場合。
- 通常の使用を明らかに超える過酷な使用による故障の場合。
- お客様の不適当または不十分な保守による故障の場合。
- 火災，風水害，地震，落雷，降灰またはそのほかの天災地変による故障の場合。
- 戦争，暴動または騒乱など破壊行為による故障の場合。
- 本製品以外の機械，施設または工場設備の故障，事故または爆発などによる故障の場合。
- 指定外の接続機器もしくは応用機器，接続部品もしくは応用部品または消耗品の使用による故障の場合。
- 指定外の電源または設置場所での使用による故障の場合。
- 特殊環境における使用 (注) による故障の場合。
- 昆虫，くも，かび，花粉，種子またはそのほかの生物の活動または侵入による故障の場合。

また，この保証は，原契約者のみ有効で，お客様から再販売されたものについては保証しかねます。

なお，本製品の使用，あるいは使用不能によって生じた損害およびお客様の取引上の損失については，責任を負いかねます。ただし，その損害または損失が，当社の故意または重大な過失により生じた場合はこの限りではありません。

注：
「特殊環境における使用」には，以下のような環境での使用が該当します。
- 直射日光が当たる場所
- 粉じんが多い環境
- 屋外
- 水，油，有機溶剤もしくは薬液などの液中，またはこれらの液体が付着する場

所

- 潮風，腐食性ガス（亜硫酸ガス，硫化水素，塩素，アンモニア，二酸化窒素，塩化水素など）がある場所
- 静電気または電磁波の強い環境
- 電源の瞬断または異常電圧が発生する環境
- 部品が結露するような環境
- 潤滑油からのオイルミストが発生する環境
- 高度 2000 m を超える環境
- 車両，船舶または航空機内など振動または衝撃が多く発生する環境

当社へのお問い合わせ

本製品の故障については，紙版説明書では巻末，電子版説明書では別ファイルに記載の「本製品についてのお問い合わせ窓口」へすみやかにご連絡ください。お問い合わせ窓口は、当社ホームページのお問い合わせページでも確認できます。お問い合わせページにある「電子計測器に関するお問い合わせ」からもご連絡いただけます。

国外持出しに関する注意

- 本製品は，特定の外国の安全規格などに準拠していない場合がありますので，当社の承諾なく日本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。
- 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。詳細は当社ホームページを参照してください。
  https://www.anritsu.com/support/export-procedures
  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，破砕または裁断処理していただきますようお願い致します。

商標・登録商標

CDMA2000®は米国電気通信工業会（TIA USA）の米国及びその他の国における登録商標です。

ソフトウェア使用許諾

お客様は，ご購入いただいたソフトウェア（プログラム，データベース，電子機器の動作・設定などを定めるシナリオ等を含み，以下「本ソフトウェア」と総称します）を使用（実行，インストール，複製，記録等を含み，以下「使用」と総称します）する前に，本「ソフトウェア使用許諾」（以下「本使用許諾」といいます）をお読みください。お客様から本使用許諾の規定にご同意いただいた場合のみ，お客様は，本使用許諾に定められた範囲において本ソフトウェアをアンリツが推奨または指定する装置（以下，「本装置」といいます）に使用することができます。お客様が本ソフトウェアを使用したとき，当該ご同意をいただいたものとします。

第 1 条 （許諾，禁止内容）

1. お客様は，本ソフトウェアを有償・無償にかかわらず第三者へ販売，開示，移転，譲渡，賃貸，リース，頒布し，または再使用させる目的で複製，開示，使用許諾することはできません。
2. お客様は，本ソフトウェアをバックアップの目的で，1 部のみ複製を作成できます。
3. 本ソフトウェアのリバースエンジニアリング，逆アセンブルもしくは逆コンパイル，または改変もしくは派生物（二次的著作物）の作成は禁止させていただきます。
4. お客様は，本ソフトウェアを本装置 1 台で使用できます。

第 2 条 （免責）

アンリツは，お客様による本ソフトウェアの使用または使用不能から生ずる損害，第三者からお客様に請求された損害を含め，一切の損害について責任を負わないものとします。ただし，当該損害がアンリツの故意または重大な過失により生じた場合はこの限りではありません。

第 3 条 （修補）

1. お客様が，取扱説明書に書かれた内容に基づき本ソフトウェアを使用していたにもかかわらず，本ソフトウェアが取扱説明書もしくは仕様書に書かれた内容どおりに動作しない場合（以下「不具合」といいます）には，アンリツは，アンリツの判断に基づいて，本ソフトウェアを無償で修補，交換し，または不具合回避方法のご案内をするものとします。ただし，以下の事項による本ソフトウェアの不具合および破損，消失したお客様のいかなるデータの復旧を除きます。
   a) 取扱説明書・仕様書に記載されていない使用目的での使用
   b) アンリツが指定した以外のソフトウェアとの相互干渉
   c) アンリツの承諾なく，本ソフトウェアまたは本装置の修理，改造がされた場合
   d) 他の装置による影響，ウイルスによる影響，災害，その他の外部要因などアンリツの責めとみなすことができない要因があった場合
2. 前項に規定する不具合において，アンリツが，お客様ご指定の場所で作業する場合の移動費，宿泊費および日当に係る現地作業費については有償とさせていただきます。
3. 本条第 1 項に規定する不具合に係る保証責任期間は本ソフトウェア購入後 6 か月または修補後 30 日いずれか遅い方の期間とさせていただきます。

第 4 条 （法令の遵守）

お客様は，本ソフトウェアを，直接，間接を問わず，核，化学・生物兵器およびミサイルなど大量破壊兵器および通常兵器，ならびにこれらの製造設備等・関連資機材等の拡散防止の観点から，日本国の「外国為替及び外国貿易法」およびアメリカ合衆国「輸出管理法」その他国内外の関係する法律，規則，規格等に違反して，いかなる仕向け地，自然人もしくは法人に対しても輸出しないものとし，また輸出させないものとします。

第 5 条 （規定の変更）

アンリツは，本使用許諾の規定の変更が，お客様の一般の利益に適う場合，または本使用許諾の目的および変更に係る諸事情に照らして合理的な場合に，お客様の承諾を得ることなく変更を実施することができます。変更にあたりアンリツは，原則として 45 日前までに，その旨（変更後の内容および実施日）を自己のホームページに掲載し，またはお客様に書面もしくは電子メールで通知します。

第 6 条 （解除）

1. アンリツは，お客様が，本使用許諾のいずれかの条項に違反したとき，アンリツの著作権およびその他の権利を侵害したとき，暴力団等反社会的な団体に属しもしくは当該団体に属する者と社会的に非難されるべき関係があることが判明したとき，または法令に違反したとき等，本使用許諾を継続できないと認められる相当の事由があるときは，直ちに

本使用許諾を解除することができます。

2. お客様またはアンリツは，30 日前までに書面で相手方へ通知することにより，本使用許諾を終了させることができます。

第 7 条 （損害賠償）

お客様が本使用許諾の規定に違反した事に起因してアンリツが損害を被った場合，アンリツはお客様に対して当該損害の賠償を請求することができます。

第 8 条 （解除後の義務）

お客様は，第 6 条により，本使用許諾が解除されまたは終了したときは直ちに本ソフトウェアの使用を中止し，アンリツの求めに応じ，本ソフトウェアおよびそれらに関する複製物を含めアンリツに返却または廃棄するものとします。

第 9 条 （協議）

本使用許諾の条項における個々の解釈について生じた疑義，または本使用許諾に定めのない事項について，お客様およびアンリツは誠意をもって協議のうえ解決するものとします。

第 10 条（準拠法）

本使用許諾は，日本法に準拠し，日本法に従って解釈されるものとします。本使用許諾に関する紛争の第一審の専属的合意管轄裁判所は，東京地方裁判所とします。

(改定履歴)

2020 年 2 月 29 日

計測器のウイルス感染を防ぐための注意

- ファイルやデータのコピー
  当社より提供する，もしくは計測器内部で生成されるもの以外，計測器にはファイルやデータをコピーしないでください。
  前記のファイルやデータのコピーが必要な場合は，メディア（USB メモリ，CF メモリカードなど）も含めて事前にウイルスチェックを実施してください。
- ソフトウェアの追加
  当社が推奨または許諾するソフトウェア以外をダウンロードしたりインストールしたりしないでください。
- ネットワークへの接続
  接続するネットワークは，ウイルス感染への対策を施したネットワークを使用してください。

ウイルス感染を防ぐための注意

インストール時

本ソフトウェア，または当社が推奨，許諾するソフトウェアをインストールする前に，PC (パーソナルコンピュータ) および PC に接続するメディア (USB メモリ，CF メモリカードなど) のウイルスチェックを実施してください。

本ソフトウェア使用時および計測器と接続時

- ファイルやデータのコピー
  次のファイルやデータ以外を PC にコピーしないでください。
  - 当社より提供するファイルやデータ
  - 本ソフトウェアが生成するファイル
  - 本書で指定するファイル

  前記のファイルやデータのコピーが必要な場合は，メディア (USB メモリ，CF メモリカードなど) も含めて事前にウイルスチェックを実施してください。
- ネットワークへの接続
  PC を接続するネットワークは，ウイルス感染への対策を施したネットワークを使用してください。
- マルウェア (ウイルスなど悪意のあるソフトウェア) からの保護
  PC をネットワークへ接続する場合は，以下のことを推奨します。
  - ファイアウォールを有効にする
  - Windows の重要な更新プログラムをインストールする
  - アンチウイルスソフトウェアを利用する

ソフトウェアを安定してお使いいただくための注意

本ソフトウェアの動作中に，PC 上にて以下の操作や機能を実行すると，ソフトウェアが正常に動作しないことがあります。

- 当社が推奨または許諾するソフトウェア以外のソフトウェアを同時に実行
- ふたを閉じる (ノート PC の場合)
- スクリーンセーバ
- バッテリ節約機能 (ノート PC の場合)

各機能の解除方法は，使用している PC の取扱説明書を参照してください。

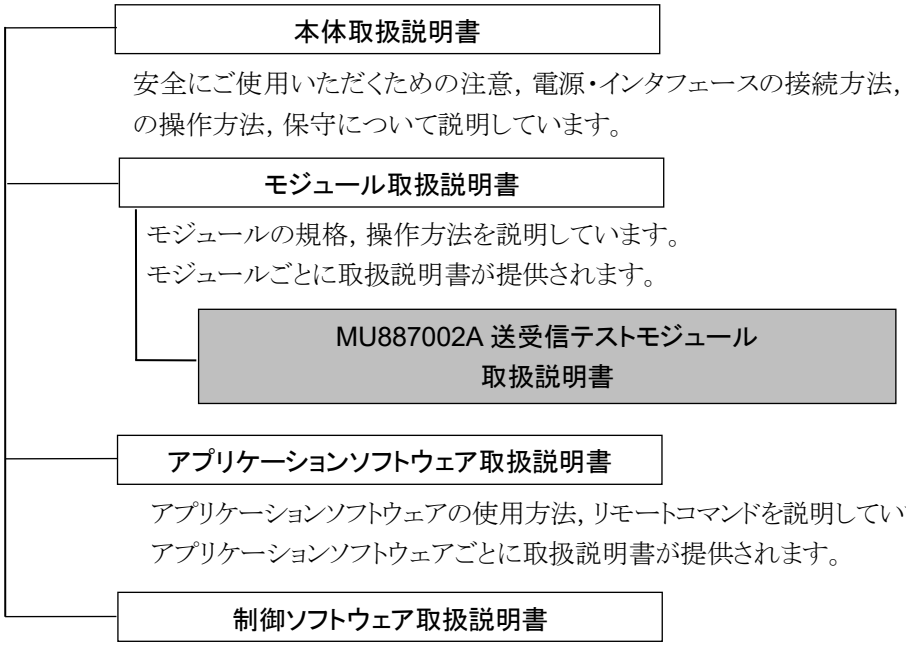
はじめに

この説明書は，MU887002A 送受信テストモジュールの使用方法，パネル，規格について記述したものです。

MT8870A ユニバーサルワイヤレステストセットに関連する製品には，以下があります。

- MT8870A ユニバーサルワイヤレステストセット（本体）
- MT8870A に装着されるモジュール
- MT8870A にインストールされるアプリケーションソフトウェア
- パーソナルコンピュータにインストールして MT8870A をリモート制御する制御ソフトウェア

これらの製品群をユニバーサルワイヤレステストセットシリーズと呼びます。ユニバーサルワイヤレステストセットシリーズの取扱説明書は，以下のように，本体，モジュール，アプリケーションソフトウェア，および制御ソフトウェアに分かれて構成されています。▭ は，本書を示します。



安全上の注意，保守については，
『MT8870A ユニバーサルワイヤレステストセット 取扱説明書』を参照してください。
性能試験については，
各アプリケーションソフトウェアの取扱説明書を参照してください。

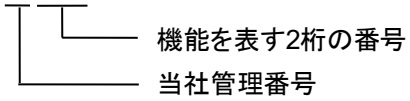
このマニュアルの表記について

本書では，特に支障のない限り，MT8870A での使用を前提に説明をします。
MT8872A を使用される場合は，読み替えてご使用ください。

また，オプション形名の当社管理番号は，理由のない限り「0」として記述します。

オプション形名：下記のように 3 桁の数字で表示されます。

MU88700XA- x x x



目次

安全にお使いいただくために iii

はじめに I

第1章 概要 1-1

1.1 概要 1-2
1.2 特長 1-3
1.3 製品構成 1-4
1.4 省略語 1-9

第2章 ご使用になる前に 2-1

2.1 各部の名称と機能 2-2
2.2 モジュールの取り付け 2-5
2.3 接続を確認する 2-7
2.4 静電気について 2-9
2.5 移動機の接続 2-11

第3章 基本操作 3-1

3.1 コマンドの言語モード 3-2
3.2 通信設定 3-3
3.3 ファイル転送 3-8
3.4 システム設定 3-9
3.5 測定前の準備 3-13
3.6 ステータスレジスタ 3-19

第4章 ベクトル信号発生器 4-1

4.1 ベクトル信号発生器 (VSG) の紹介 4-2
4.2 動作モード 4-3
4.3 ノーマルモード 4-4
4.4 シーケンスモード 4-15

第5章 SCPIコマンドリファレンス 5-1

5.1 コマンド一覧 5-2
5.2 コマンドの詳細 5-27

第6章 Nativeコマンドリファレンス..... 6-1

6.1 コマンド一覧.....6-2
6.2 コマンドの詳細.....6-12

第7章 性能試験..... 7-1

7.1 概要.....7-2
7.2 性能試験用機器一覧表.....7-3
7.3 各測定に対する性能試験詳細.....7-8
7.4 サービスについて.....7-44

第8章 ユーティリティツール..... 8-1

8.1 ユーティリティツールの概要.....8-2
8.2 インストール.....8-8
8.3 操作方法.....8-15

付録 A 規格.....A-1

付録 B ソフトウェアライセンスについて.....B-1

付録 C 性能試験記録表..... C-1

付録 D ランプの状態表示..... D-1

付録 E MX887092A.....E-1

索引 索引-1

ここでは，MU887002A 送受信テストモジュールの機能の概要と製品構成について説明します。性能・機能仕様については「付録 A 仕様」を参照してください。

1.1　概要 1-2
1.2　特長 1-3
1.3　製品構成 1-4
1.3.1　標準構成 1-4
1.3.2　オプション 1-5
1.3.3　アプリケーションソフトウェア 1-6
1.3.4　応用部品 1-8
1.3.5　保証サービス 1-8
1.4　省略語 1-9

1.1　概要

MU887002A 送受信テストモジュール（以下，モジュールと呼びます）は，MT8870A ユニバーサルワイヤレステストセット（以下，MT8870A と呼びます）に装着するモジュールです。

- スロットサイズ:　2 スロットサイズ 2Tx/2Rx
  1 TRX/1 スロットとして独立して使用可能
- 測定ポート数:　12 Port × 2 スロット
- 送受信周波数:　0.4～6 GHz
- 信号帯域幅:　Downlink/Uplink とも 200 MHz
- 同時出力:　12 Port/1 TRX
- 実装:　MU887000A/01A との同時実装も可能

モジュールの機能は次のとおりです。

- 信号発生器
  周波数，レベルを設定して高周波信号を出力します。
  IQ 変調情報が記載されたデータファイルに従って信号に変調をかけたり，AWGN (Additive White Gaussian Noise) を付加したりできます。
- シーケンス機能
  信号の周波数，レベル，変調信号，タイムスケジュールなどを記載したリストに従って，信号を出力します。
  リストは 16 個まで保持できます。リストの繰り返し回数，1 つのリストの処理が終了した後ジャンプ先，リストの実行開始条件などが編集できます。
- システム設定
  日時，時刻，ネットワークアドレスの設定や，ライセンスキー，およびソフトウェアのバージョンを管理します。
  添付の MX887900A ユーティリティツールを使用すると，パーソナルコンピュータからモジュールのシステム設定ができます。
- ステータスレジスタ
  SCPI (Standard Commands for Programming Interfaces) に規定されたステータスレジスタを持ち，イベントやエラーを検出できます。

別売のアプリケーションソフトウェアを購入することで，通信規格に適合した信号を送信できます。また，シグナルアナライザの機能が追加され，受信信号を解析できます。別売のアプリケーションソフトウェアは 1.3.3 項を参照してください。

モジュールを制御するには，イーサネット経由でコマンドを送信します。
コマンドの書式には，次の 2 通りがあります。

- SCPI　SCPI 規格で規定されたの書式のコマンドです。
- Native　当社の既存製品と互換性があるコマンドです。

1.2 特長

モジュールの特長は次のとおりです。

(1) 高速な測定

(2) ソフトウェアの切り替えにより，多くの無線規格に対応

(3) DL 信号は，校正されたレベルで任意の
複数のテストポート（TRX 1/2 それぞれ最大 12 Port）から同時に出力可能

(4) 高レベル（0 dBm）出力，200 MHz 帯域幅により，試験系，試験効率を改善

(5) MU887000A/01A との高い互換性をもつことにより評価プログラム作成・再評価が不要

(6) 既存製造設備，製造ライン設計をそのまま利用可能な MT8870A に実装する 2 スロット構成

モジュールにインストールしたアプリケーションソフトウェアを切り替えることにより，様々な通信規格の試験ができます。

1.3　製品構成

この節では，モジュールの標準構成のほか，モジュールを活用するためのオプション，アプリケーションソフトウェア，応用部品，保証サービスを紹介します。

1.3.1　標準構成

モジュールの標準構成を次の表に示します。

表1.3.1-1　標準構成

項目	形名・記号	品名	数量
本体	MU887002A	送受信テストモジュール	1
添付品	W4048AW	MU887002A 送受信テストモジュール 取扱説明書	1
	KUWM-32-M4-16-OR 	ウィングノブ MU887002A を MT8870A および MT8872A のスロットから取り外すためのネジです。	1

取扱説明書は，添付されるメディア（DVD など）に含まれます。
複数の電子ファイルが 1 つまたは複数のメディア（DVD など）にまとめて収納されることがあります。

1.3.2 オプション

モジュールの機能を拡張するためのオプションを次の表に示します。オプションはすべて別売りとなります。注文の際は，形名・記号，品名，数量をご指定ください。

本器のオプションは「表 1.3.2-1　MU887002A オプション」のとおりです。
追加されているオプションの形名は，正面パネルのラベルに記載されています。

本書では，オプション形名の当社管理番号は，理由のない限り「0」として記述します。

表1.3.2-1　MU887002A オプション

形名・記号	品名	備考
MU887002A-007	7GHz 拡張機能	出荷時取り付け
MU887002A-107	7GHz 拡張機能後付	工場引き取り*
MU887002A-207	7GHz 拡張機能後付	現法引き取り*
MU887002A-097	7GHz 拡張ハードウェア	出荷時取り付け
MU887002A-197	7GHz 拡張ハードウェア後付	工場引き取り*
MU887002A-297	7GHz 拡張ハードウェア後付	現法引き取り*

＊:　シリアル番号により取り付け作業内容が異なります。
　　詳細は，当社営業担当までお問い合わせください。

オプション形名：下記のように 3 桁の数字で表示されます。

MU887002A- x x x

機能を表す2桁の番号です。
当社管理番号です。
0:　出荷時取り付け
1:　後付けオプション。取り付けには，当社への引き取りが必要です。
2:　後付けオプション。取り付けには，現法への引き取りが必要です。

1.3.3　アプリケーションソフトウェア

モジュールで使用できるアプリケーションソフトウェアと，その波形ファイルは別売です。アプリケーションソフトウェアの詳細な機能や性能については，当社または当社代理店にお問い合わせください。
アプリケーションソフトウェアと波形ファイルを以下の表に示します。

表1.3.3-1　アプリケーションソフトウェア

形名・記号	品名
MX887010A	セルラ規格用シーケンス測定
MX887011A	W-CDMA/HSPA Uplink 送信測定
MX887012A	GSM/EDGE Uplink 送信測定
MX887013A	LTE FDD Uplink 送信測定
MX887014A	LTE TDD Uplink 送信測定
MX887015A	CDMA2000 Reverse Link 送信測定
MX887016A	1xEV-DO Reverse Link 送信測定
MX887017A	TD-SCDMA Uplink 送信測定
MX887018A	NR FDD sub-6GHz Uplink 送信測定
MX887019A	NR TDD sub-6GHz Uplink 送信測定
MX887021A	W-CDMA/HSPA Downlink 送信測定
MX887023A	LTE FDD Downlink 送信測定
MX887030A	WLAN 802.11b/g/a/n 送信測定
MX887031A	WLAN 802.11ac 送信測定
MX887032A	WLAN 802.11p 送信測定
MX887033A	WLAN 802.11ax 送信測定
MX887034A	WLAN 802.11be 送信測定
MX887040A	Bluetooth 送信測定
MX887050A	近距離無線パワー/周波数測定機能
MX887060A	IEEE 802.15.4 送信測定
MX887061A	Z-Wave 送信測定
MX887065A	Category M FDD Uplink 送信測定
MX887067A	NB-IoT Uplink 送信測定
MX887068A	LTE-V2X 送信測定
MX887090A	マルチ DUT メジャメントスケジューラ
MX887092A	MU887002A パスロス測定機能

表1.3.3-2 波形ファイル

形名・記号	品名
MV887011A	W-CDMA/HSPA Downlink 波形ファイル
MV887012A	GSM/EDGE Downlink 波形ファイル
MV887013A	LTE FDD Downlink 波形ファイル
MV887014A	LTE TDD Downlink 波形ファイル
MV887015A	CDMA2000 Forward Link 波形ファイル
MV887016A	1xEV-DO Forward Link 波形ファイル
MV887017A	TD-SCDMA Downlink 波形ファイル
MV887018A	NR FDD sub-6GHz Downlink 波形ファイル
MV887019A	NR TDD sub-6GHz Downlink 波形ファイル
MV887021A	W-CDMA/HSPA Uplink 波形ファイル
MV887023A	LTE FDD Uplink 波形ファイル
MV887030A	WLAN 802.11b/g/a/n 波形ファイル
MV887031A	WLAN 802.11ac 波形ファイル
MV887032A	WLAN 802.11p 波形ファイル
MV887033A	WLAN 802.11ax 波形ファイル
MV887034A	WLAN 802.11be 波形ファイル
MV887040A	Bluetooth 波形ファイル
MV887060A	IEEE 802.15.4 波形ファイル
MV887061A	Z-Wave 波形ファイル
MV887065A	Category M FDD Downlink 波形ファイル
MV887067A	NB-IoT Downlink 波形ファイル
MV887068A	LTE-V2X 波形ファイル
MV887100A	GPS 波形ファイル
MV887101A	Galileo 波形ファイル
MV887102A	GLONASS 波形ファイル
MV887103A	BeiDou 波形ファイル
MV887104A	QZSS 波形ファイル
MV887110A	DVB-H 波形ファイル
MV887111A	ISDB-T 波形ファイル

表1.3.3-3　アプリケーションソフトウェアオプション

形名・記号	品名
MX887013A-001	LTE-Advanced FDD Uplink CA 送信測定
MX887014A-001	LTE-Advanced TDD Uplink CA 送信測定
MX887018A-001	NR FDD Contiguous ENDC 送信測定
MX887019A-001	NR TDD Contiguous ENDC 送信測定
MX887040A-001	DLE 送信測定
MX887040A-002	2LE 送信測定
MX887040A-003	BLR 送信測定
MX887040A-004	BLE AoA/AoD 送信測定
MX887068A-001	LTE-V2X PSCCH 送信測定

表1.3.3-4　波形ファイルオプション

形名・記号	品名
MV887040A-001	DLE 波形ファイル
MV887040A-002	2LE 波形ファイル
MV887040A-003	BLR 波形ファイル
MV887040A-004	BLE AoA/AoD 波形ファイル
MV887100A-002	GPS L5 波形ファイル

1.3.4　応用部品

必要に応じて応用部品（アクセサリー）をご利用ください。応用部品はすべて別売りとなります。注文の際は，形名・記号，品名，数量をご指定ください。

表1.3.4-1　応用部品

形名・記号	品名	備考
W4048AW	MU887002A 取扱説明書	和文，メディア（DVD など）
J0576B	同軸コード，1.0 m	N-P・5D-2W・N-P
J0576D	同軸コード，2 m	N-P・5D-2W・N-P
J0127A	同軸コード，1 m	BNC-P・RG58A/U・BNC-P
J0127C	同軸コード，0.5 m	BNC-P・RG58A/U・BNC-P

1.3.5　保証サービス

表1.3.5-1　MU887002A 保証サービス

形名・記号	品名	備考
MU887002A-ES210	2 年保証延長サービス	
MU887002A-ES310	3 年保証延長サービス	
MU887002A-ES510	5 年保証延長サービス	

1.4 省略語

本書の説明，およびモジュールのパネルで使用している省略語を次の表に示します。

表1.4-1 省略語

省略語	正式名
ARB	Arbitrary Waveform generator
AWGN	Additive White Gaussian Noise
C/N	Carrier to Noise Ratio
CW	Continuous Wave
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
FPGA	Field Programmable Gate Array
FTP	File Transfer Protocol
GPIB	General Purpose Interface Bus
GPRF	General Purpose Radio Frequency
IP	Internet Protocol
L	Left
MAC	Media Access Control
Max.	Maximum
NR	New Radio
R	Right
rms	Root mean square
SA	Signal Analyzer
SG	Signal Generator
TRX	Transceiver
VISA	Virtual Interface Software Architecture
VSG	Vector Signal Generator
V2X	Vehicle-to-Everything
WLAN	Wireless Local Area Network

この章では，各部の名称と機能，モジュールの設置方法，そして外部機器の接続方法について説明しています。

⚠ 注意

MU887002A を購入後，初めてお使いになるとき，および長期間使用されなかったときはレベル確度に影響を与えるおそれがあるため，十分に予熱してからお使いください。

2.1 各部の名称と機能 2-2
2.2 モジュールの取り付け 2-5
2.2.1 モジュールを取り付ける 2-5
2.2.2 モジュールを取り外す 2-6
2.3 接続を確認する 2-7
2.3.1 イーサネット 2-7
2.4 静電気について 2-9
2.5 移動機の接続 2-11

2.1　各部の名称と機能

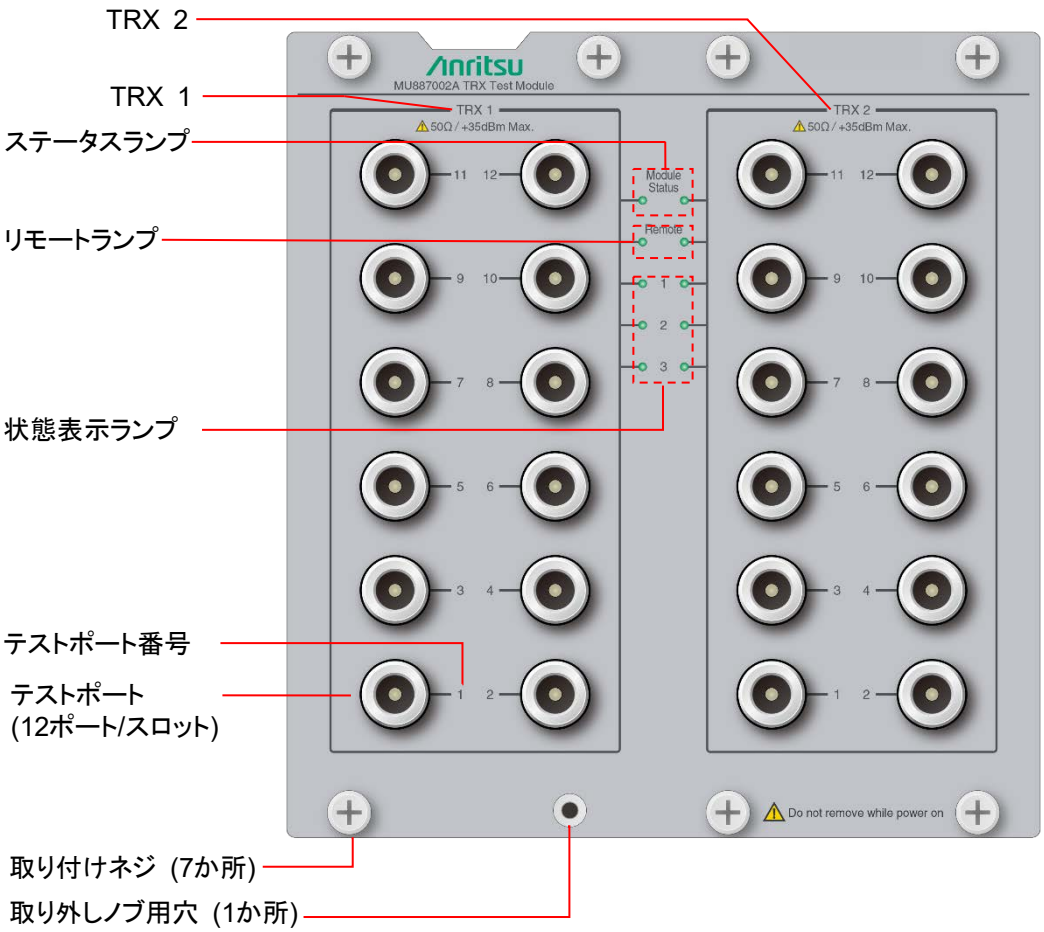


図 2.1-1　正面パネルの名称

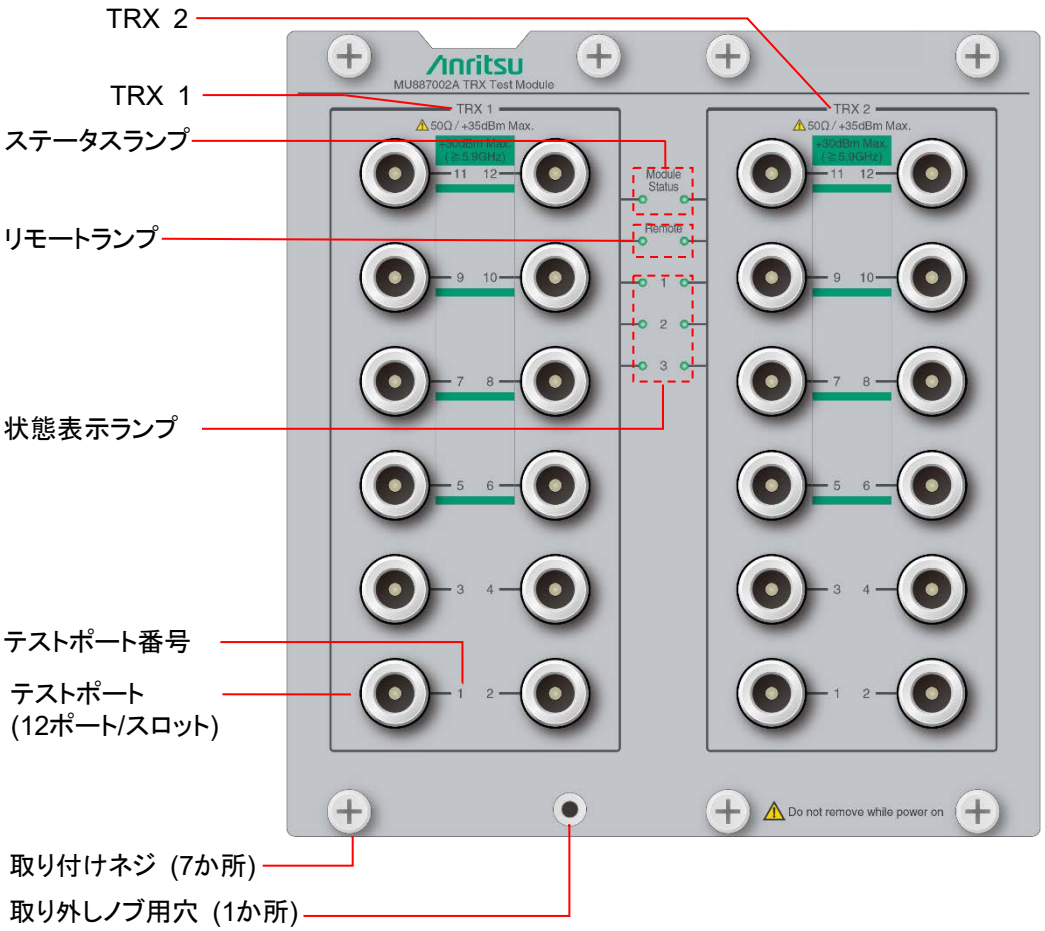


図 2.1-2　正面パネルの名称 (MU887002A-007 搭載時)

表 2.1-1　正面パネルの機能

名称	機能
取り付けネジ（7 か所）	モジュールを MT8870A または MT8872A に固定します。
取り外しノブ用穴（1 か所）	モジュールを取り外すとき，ウィングノブを挿入する穴です。
ステータスランプ (Module Status)*	モジュールの状態を表示します。 詳細は「付録 D ランプの状態表示」を参照してください。
リモートランプ (Remote)*	リモート制御されているときに点灯します。 詳細は「付録 D ランプの状態表示」を参照してください。
状態表示ランプ (1～3)*	モジュール，アプリケーションソフトウェアの状態を表示します。 詳細は「付録 D ランプの状態表示」を参照してください。
テストポート 1～12	高周波信号用のコネクタです。 信号の入力と出力を同時にできます。
MU887002A-007 搭載時 テストポート 1～4	高周波信号用のコネクタです。 6.0 GHz 以下の周波数において，信号の入力と出力を同時にできます。
MU887002A-007 搭載時 テストポート 5～12	高周波信号用のコネクタです。 信号の入力と出力を同時にできます。 ただし，5.9 GHz 以上の周波数において，下記となります。 ・ 信号の入力と出力を同時にできません。 ・ 最大入力レベル: +30 dBm

＊:　電源投入時はすべてのランプが橙色に点灯します。

⚠ 注意

必ず取り付けネジを締めて使用してください。

取り付けネジが緩んでいる状態で MT8870A を使用すると，MU887002A が脱落してけがをするおそれがあります。

2.2　モジュールの取り付け

⚠ 注意

モジュールの取り付け，および取り外しをする前に，MT8870A の電源を切断してください。

2.2.1　モジュールを取り付ける

モジュールはスロットのどの位置にも取り付けられます。

1. 2.5 mm のプラスドライバを使用して，ブランクパネルのネジを緩めます。
2. ネジを手前に引いて，ブランクパネルを外します。
3. モジュールをスロットの奥まで差し込みます。
4. プラスドライバを使用して，パネルのネジを締めます。
   ネジが入りにくい場合，他のネジを持ってモジュールを持ち上げ，ネジを締めます。

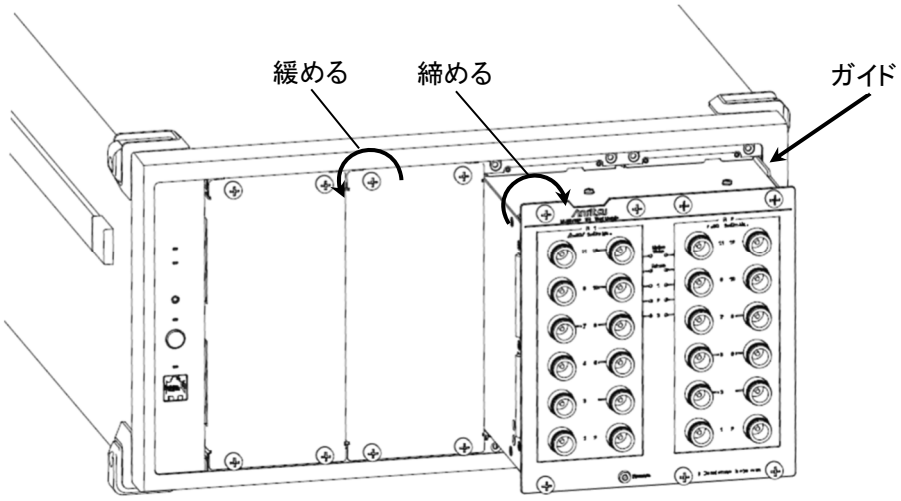


図 2.2.1-1　モジュールの取り付け

⚠ 注意

- モジュールを取り付けまたは取り外しをするときは，リストストラップを装着するなど，静電気対策をとってください。
- モジュールを取り付けまたは取り外しをするときは，モジュールのパネルから同軸ケーブルを外してください。モジュールを取り付けまたは取り外しの作業中に誤ってモジュールに信号が印加されて，モジュールを破損するおそれがあります。

2.2.2　モジュールを取り外す

1. プラスドライバを使用して，パネルの 7 つのネジを緩めます。
2. 取り外しノブ用穴にウィングノブを入れ，ウィングノブを右に回して，モジュールが手前に抜けて来るまで右に回します。

注：

ウィングノブに指を掛けて引っ張らないでください。

3. ウィングノブを左に回して，モジュールからウィングノブを取り外します。

注：

ウイングノブを取り外さないと取り付け時にモジュールを挿入できません。無理に挿入しようとすると破損するおそれがあります。

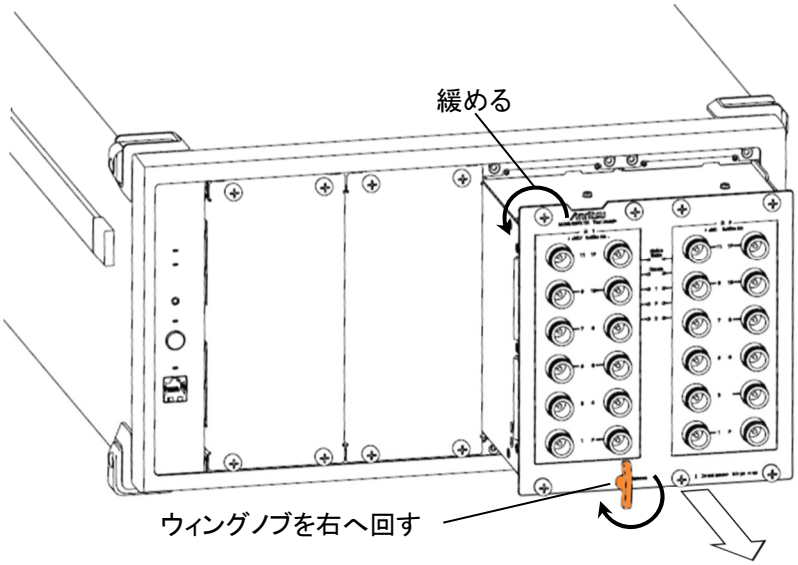


図 2.2.2-1　モジュールの取り外し

4. モジュールを持って，モジュールを引き出します。
5. モジュールを外したスロットにブランクパネルをはめます。
6. プラスドライバを使用して，ブランクパネルのネジを締めます。

⚠ 注意

使用しないスロットにはブランクパネルを取り付けてください。使用しないスロットにブランクパネルを取り付けないと，ほかのスロットに装着されているモジュール，および MT8870A 内部が冷却されず，十分な性能が得られなかったり，動作が不安定になったりするおそれがあります。

モジュールを引き出すときには，重量があるため，モジュール本体に両手を添えて引き出してください。モジュールが落下し，思わぬけがをするおそれがあります。

2.3　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 MT8870A を認識できるか確認します。
モジュールが MT8870A のスロット 1，2 に装着されていて，インタフェースの設定が次のとおりであるという前提で説明します。
IPv4 アドレス:　スロット 1:　192.168.1.1，スロット 2:　192.168.1.2
ポート番号:　　56001

2.3.1　イーサネット

ここでは，フリーソフトウェアの Tera Term を使用する方法を説明します。

IPv4 アドレスごとに，別々の Tera Term を起動し，設定を行います。

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 MT8870A 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. Tera Term が MT8870A を認識すると，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。
5. [OK] をクリックします。



6. *IDN?を送信します。
MT8870A からのレスポンスが表示されることを確認します。

2.4　静電気について

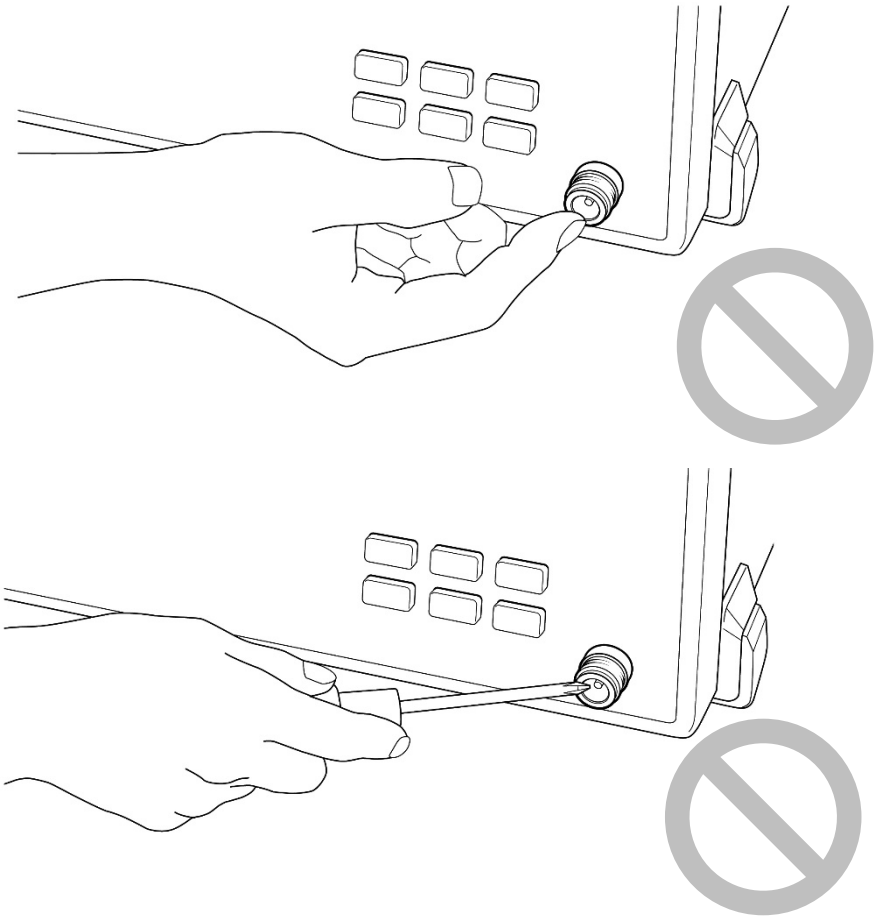
ここでは，静電気による機器の破損を防止するための一般的な注意を説明します。

⚠ 注意

- 必ず 3 芯電源コードを用いて MT8870A，および被測定物（実験回路を含む）の両方をアースに接続してください。双方がアースに接続されていることを確認してから，MT8870A および被測定物（実験回路を含む）を同軸ケーブルで接続してください。

  MT8870A と被測定物がアースに接続されていない状態で，MT8870A と被測定物を接続すると，静電気により MT8870A の入力回路を破損するおそれがあります。

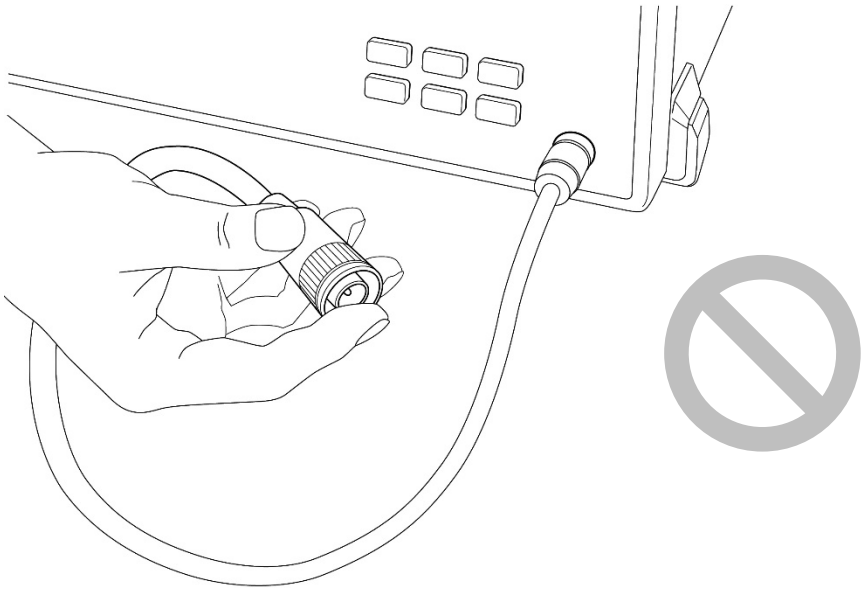
- コネクタの中心導体を触ったり，金属を接触させたりしないでください。MT8870A の入力回路を破損するおそれがあります。



⚠ 注意

MT8870A の入力コネクタに接続している同軸ケーブルの中心導体を触ったり，金属を接触させたりしないでください。

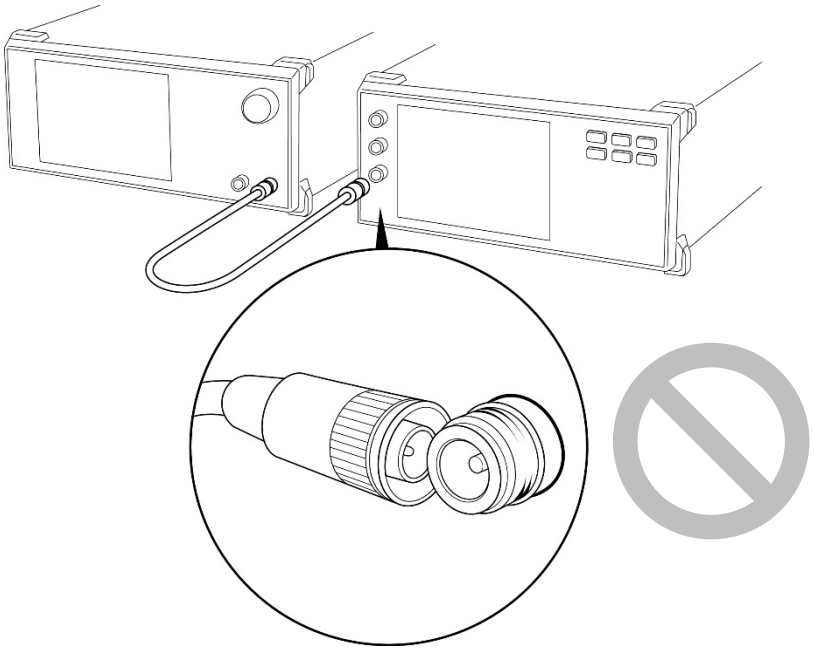
MT8870A の入力回路を破損するおそれがあります。



⚠ 注意

同軸ケーブルをコネクタに接続するときに，中心導体を金属に接触させないでください。

MT8870A の入力回路を破損するおそれがあります。



2.5　移動機の接続

モジュールと測定対象である移動機との接続方法は，移動機のRFコネクタの形状やアサイメントによって異なりますが，ここでは基本的な接続の方法を説明します。

注:

RF 信号の測定を行う場合は，使用するモジュールの出力コネクタが，「3.5.4 ポートの設定」で指定したコネクタと一致していることを確認してください。

テストポート 1 と移動機を接続する場合

正面パネルにある Test Port 1 と，移動機の RF 信号入出力部を RF ケーブルで接続します。直接つなぐことができない場合は別途アダプタを用意する必要があります。

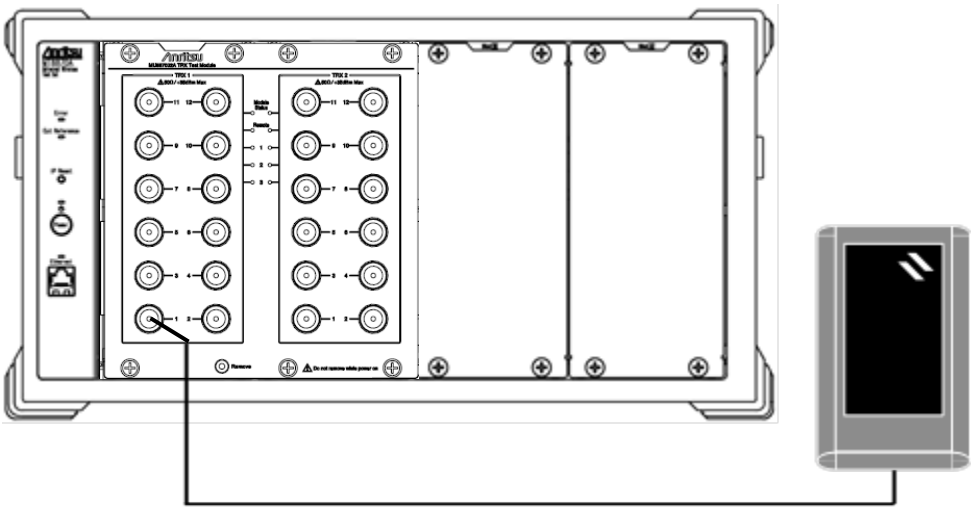


図 2.5-1　移動機との接続例

この章では，モジュールの基本操作を説明します。
この章で紹介するコマンドの詳細は，「第 5 章 SCPI コマンドリファレンス」，または「第 6 章 Native コマンドリファレンス」を参照してください。

3.1 コマンドの言語モード 3-2
3.2 通信設定 3-3
3.2.1 イーサネットの設定 3-3
3.2.2 終端文字 3-7
3.2.3 設定項目の反映 3-7
3.3 ファイル転送 3-8
3.4 システム設定 3-9
3.4.1 一般設定 3-9
3.4.2 製品情報の確認 3-11
3.5 測定前の準備 3-13
3.5.1 周波数の設定 3-13
3.5.2 レベル校正 3-14
3.5.3 ケーブルの損失補正 3-16
3.5.4 ポートの設定 3-18
3.5.5 トリガコネクタの設定 3-18
3.6 ステータスレジスタ 3-19
3.6.1 レジスタの構成 3-19
3.6.2 ステータスバイトレジスタ 3-22
3.6.3 標準イベントステータスレジスタ 3-24
3.6.4 イベントレジスタ 3-25
3.6.5 実行状態レジスタ 3-26
3.6.6 クエスチョナブルレジスタ 3-29

3.1　コマンドの言語モード

モジュールをリモート制御するコマンドは，SCPI コマンドモードと，Native コマンドモードの 2 つの言語モードがあります。

・ SCPI コマンドモード
SCPI 規格（http://www.ivifoundation.org/docs/SCPI-99.PDF）に規定された書式のコマンドです。
SCPI のコマンドは次の例のように，コロンで区切られ機能別に階層構造にコマンドが設計されています。小文字の部分，および角括弧で囲まれた部分は省略できます。

```
:SYSTem:ERRor:CODE[:NEXT]
```

SCPI コマンドモードでは，機器の状態をモニタするために実行状態レジスタ，クエスチョナブルステータスレジスタを使用します。

・ Native コマンドモード
SCPI 規格の書式には従わないコマンドです。Native のコマンドは，次の例のように SCPI 規格のコマンドに比べて少ない文字数でコマンドを記述できます。SCPI のコマンドと違って，コマンドの文字を省略できません。

```
SYSERRCODE
```

Native コマンドモードでは，機器の状態をモニタするためにイベントステータスレジスタを使用します。

動作モードを設定するには，次のコマンドを使用します。
Native コマンドモードから SCPI コマンドモードに切り替える場合

```
SYST:LANG SCPI
```

SCPI コマンドモードから Native コマンドモードに切り替える場合

```
:SYSTem:LANGuage NAT
```

IEEE488.2 で規定されている以下の共通コマンドは，どちらの言語モードでも使用できます。

```
*CLS, *ESE, *ESR, *IDN, *OPC, *RST, *SRE, *STB, *TRG, *TST,
*WAI
```

3.2 通信設定

モジュールは，イーサネット経由で制御します。
イーサネットは，IPv4 (Internet Protocol version 4) と IPv6 (Internet Protocol version 6) のどちらか片方，または両方を使用できます。

3.2.1 イーサネットの設定

IPv4，IPv6 のどちらを使用するか，両方を使用するか設定します。

IPv4 では，次の項目を設定します。
アドレスの設定方法
IP アドレスを自動的に取得するか，指定した IP アドレスを使用するか選択します。
指定した IP アドレスを使用する場合，IP アドレス，サブネットマスク，デフォルトゲートウェイを設定します。
IP アドレスを自動的に取得する場合，取得した IP アドレス，サブネットマスク，デフォルトゲートウェイを確認できます。
DNS の利用
モジュールがホスト名を使用して他の機器と通信する場合，DNS (Domain Name System) をオンに設定します。
DNS サーバを利用する場合，DNS のプライマリアドレス，セカンダリアドレス，ホスト名，ドメイン名を設定します。
アプリケーションソフトウェア取扱説明書で指示がある場合に，オンに設定します。

IPv6 では，次の項目を設定します。
アドレスの設定方法
IPv6 アドレスを自動的に取得するか，指定した IPv6 アドレスを使用するか選択します。
指定した IP アドレスを使用する場合，IP アドレス，プリフィクス長，デフォルトルータを設定します。
IP アドレスを自動的に取得する場合，取得した IP アドレス，プリフィクス長，デフォルトルータ，リンクローカルアドレスを確認できます。
DNS の利用
モジュールがホスト名を使用して他の機器と通信する場合，DNS (Domain Name System) をオンに設定します。
DNS サーバを利用する場合，DNS のプライマリアドレス，セカンダリアドレス，ホスト名，ドメイン名を設定します。
アプリケーションソフトウェア取扱説明書で指示がある場合に，オンに設定します。

設定の初期化

イーサネットの設定は，次の手順で初期化できます。

1. MT8870A の電源を切断します。
2. MT8870A の IP Reset ボタンを押しながら電源を投入します。
3. MT8870A の Error ランプが赤色で点滅するまで，IP Reset ボタンを押し続けます。
4. Error ランプが点滅したら，IP Reset ボタンを放します。

初期化した後の設定値を次の表に示します。

表 3.2.1-1　IP アドレスの初期値

MT8870A のスロット番号	IPv4 アドレス	IPv6 アドレス/プリフィクス長
1	192.168.1.1	0/0
2	192.168.1.2	0/0
3	192.168.1.3	0/0
4	192.168.1.4	0/0

表 3.2.1-2　イーサネットの初期値

	項目	設定値
IPv4	サブネットマスク	255.255.255.0
	デフォルトゲートウェイ	(未設定)
	DNS 設定	使用しない
	DNS プライマリアドレス	(未設定)
	DNS セカンダリアドレス	(未設定)
IPv6	リンクローカルアドレス/プリフィクス長	MAC アドレスにより自動割り付け
	デフォルトルータ	(未設定)
	DNS 設定	使用しない
	DNS プライマリアドレス	(未設定)
	DNS セカンダリアドレス	(未設定)
共通	ホスト名	HOST-n*
	ドメイン名	DOMAIN

＊：　n は 1～4 のスロット番号です。
MU887002A がスロット 3 に装着されている場合，HOST-3 となります。

⚠ 注意

イーサネット経由の制御を終了する場合は，セッションをクローズして，パネルの Remote ランプが消灯してからコンピュータの制御ソフトウェアを終了してください。

イーサネットで MU887002A に接続している場合，パネルの Remote ランプが点灯した状態で制御ソフトウェアを終了すると，その後イーサネット経由の制御ができないことがあります。この場合は MT8870A の電源を入れなおしてください。

イーサネットの設定，問い合わせに使用するコマンドは次のとおりです。

IPV4/IPv6 共通

- IPv4 アドレスの使用/不使用，IPv6 アドレスの使用/不使用
  IPVER
  :SYSTem:COMMunicate:NET:IPVersion
- ドメイン名
  DOMAINNAME
  :SYSTem:COMMunicate:NET:DNS:DOMain
- ホスト名
  HOSTNAME
  :SYSTem:COMMunicate:NET:DNS:HOST

IPv4

- IPv4 アドレスの設定方法
  IPV4_TYPE
  :SYSTem:COMMunicate:NET:IPVFour:ADDRess:TYPE
- 固定 IPv4 アドレス
  IPV4_STATIC_ADDR_ALLMODULE
  :SYSTem:COMMunicate:NET:IPVFour:STATic:IPADdress
- 固定サブネットマスク
  IPV4_STATIC_SUBNETMASK
  :SYSTem:COMMunicate:NET:IPVFour:STATic:SMASk
- 固定デフォルトゲートウェイ
  IPV4_STATIC_DEFAULTGATEWAY
  :SYSTem:COMMunicate:NET:IPVFour:STATic:GIP
- 現在の IPv4 アドレス問い合わせ
  IPV4_CURRENT_ADDR
  :SYSTem:COMMunicate:NET:IPVFour:CURRent:IPADdress
- 現在のサブネットマスク問い合わせ
  IPV4_CURRENT_SUBNETMASK
  :SYSTem:COMMunicate:NET:IPVFour:CURRent:SMASk
- 現在のデフォルトゲートウェイ問い合わせ
  IPV4_CURRENT_DEFAULTGATEWAY
  :SYSTem:COMMunicate:NET:IPVFour:CURRent:GIP

- DNS の使用
  IPV4_DNS_AUTO_SW
  :SYSTem:COMMunicate:NET:IPVFour:DNS:AUTO:ENABle
- DNS のプライマリ・アドレス
  IPV4_DNS_PRIMARY
  :SYSTem:COMMunicate:NET:IPVFour:DNS:PRIMary
- DNS のセカンダリ・アドレス
  IPV4_DNS_SECONDARY
  :SYSTem:COMMunicate:NET:IPVFour:DNS:SECondary

IPv6

- IPv6 アドレスの設定方法
  IPV6_TYPE
  :SYSTem:COMMunicate:NET:IPVSix:ADDRess:TYPE
- 固定 IPv6 アドレス
  IPV6_STATIC_ADDR_ALLMODULE
  :SYSTem:COMMunicate:NET:IPVSix:STATic:IPADdress
- 固定デフォルトルータ
  IPV6_STATIC_DEFAULTROUTER
  :SYSTem:COMMunicate:NET:IPVSix:STATic:DEFRouter
- 現在の IPv6 アドレス問い合わせ
  IPV6_CURRENT_ADDR
  :SYSTem:COMMunicate:NET:IPVSix:CURRent:IPADdress
- 現在のリンクローカルアドレス問い合わせ
  IPV6_CURRENT_LINK_LOCAL_ADDR
  :SYSTem:COMMunicate:NET:IPVSix:CURRent:LOCaladdress
- 現在のデフォルトルータ問い合わせ
  IPV6_CURRENT_DEFAULTROUTER
  :SYSTem:COMMunicate:NET:IPVSix:CURRent:DEFRouter
- DNS の使用
  IPV6_DNS_AUTO_SW
  :SYSTem:COMMunicate:NET:IPVSix:DNS:AUTO:ENABle
- DNS のプライマリ・アドレス
  IPV6_DNS_PRIMARY
  :SYSTem:COMMunicate:NET:IPVSix:DNS:PRIMary
- DNS のセカンダリ・アドレス
  IPV6_DNS_SECONDARY
  :SYSTem:COMMunicate:NET:IPVSix:DNS:SECondary

3.2.2 終端文字

モジュールが送信する通信メッセージの末尾に付ける文字列を，次のコマンドで設定します。LF，CRLF，文字無しから選択できます。

DELM
:SYSTem:COMMunicate:GPIB[:SELF]:DELimiter

TRM
:SYSTem:COMMunicate:GPIB[:SELF]:TERMinator

3.2.3 設定項目の反映

イーサネット，または終端文字の項目を変更した場合，新しい設定を有効にするにはMT8870Aに装着されているモジュールを再起動します。
再起動する場合は，下記のコマンドをイーサネットの項目を変更したモジュールで使用してください（別のモジュールで使用した場合は更新されません）。

NETRESTART
:SYSTem:COMMunicate:NET:RESTart

電源スイッチの入れ直しや，IEEE488.2 共通コマンドの*RST では通信設定が更新されません。

3.3　ファイル転送

イーサネット経由でモジュールを制御する場合，モジュール内部の記憶媒体と FTP を使用してファイル転送ができます。
このためには，ファイル転送を開始する前に次の項目をモジュールに登録します。

- ユーザ名　　:MT8870A
- パスワード　:anritsu

なお，アカウント数はひとつだけです。

ユーザ名，パスワードを管理するには，以下のコマンドを使用します。

- ユーザ名の問い合わせ
  FTPUSER_LIST
  :SYSTem:COMMunicate:NET:FTP:USER:CATalog
- ユーザ名の変更
  FTPUSER_CHANGE
  :SYSTem:COMMunicate:NET:FTP:USER:CHANge:NAME
- パスワードの変更
  FTPUSER_PASSWD
  :SYSTem:COMMunicate:NET:FTP:USER:CHANge:PASS

ファイルを転送するには，制御用コンピュータに FTP クライアントソフトウェアをインストールします。

FTP クライアントソフトウェアに，モジュールの IP アドレス，ユーザ名，パスワードを入力すると，FTP による通信ができます。

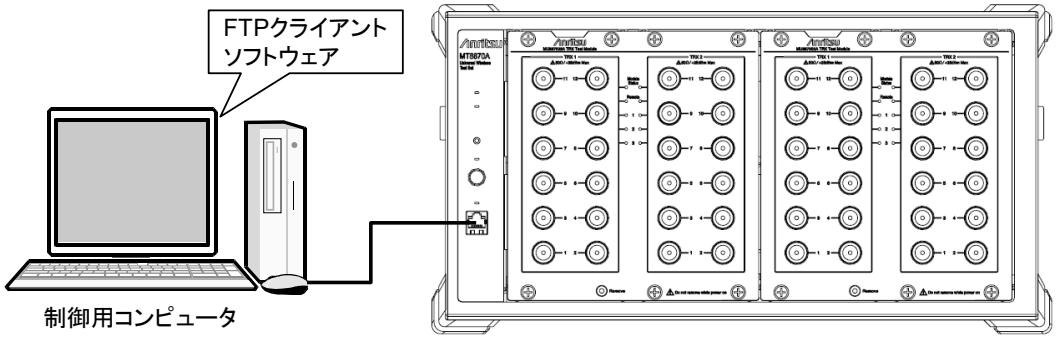


図 3.3-1　ファイル転送時の接続例

パスワードを忘れた場合は，MT8870A の IP Reset ボタンを押しながら電源を投入します。ユーザ名とパスワードが初期設定値に変更されます。

3.4 システム設定

3.4.1 一般設定

システム全般に関わる設定項目，問い合わせ項目には以下があります。

- 自己診断結果の問い合わせ
- 日付・時刻
- ブザー
- 言語モード
- アプリケーションソフトウェアの切り替え
- エラーの問い合わせ
- ログの取得
- ファームウェアの更新
- ライセンスキーのインストール
- パラメータの初期化
- 再起動
- アプリケーションソフトウェアの切り替え

システム全般に関わる設定項目，問い合わせに使用するコマンドは次のとおりです。

- 自己診断結果の問い合わせ
  `*TST`
  `SYSTEST`
  `:SYSTem:SELF:TEST`
- 日付・時刻
  `DATETIME`
  `:SYSTem:DATE`
- ブザー
  `BEEP`
  `:SYSTem:BEEPer`
- 言語モード
  `SYST:LANG`
  `:SYSTem:LANGuage`
- エラーの問い合わせ
  `SYSERR`
  `:SYSTem:ERRor`
- エラー数の問い合わせ
  `SYSERRCNT`
  `:SYSTem:ERRor:COUNt`
- エラーコードの問い合わせ
  `SYSERRCODE`
  `:SYSTem:ERRor:CODE[:NEXT]`
- すべてのエラーの問い合わせ
  `SYSERRALL`
  `:SYSTem:ERRor:ALL`
- すべてのエラーコードの問い合わせ
  `SYSERRCODEALL`
  `:SYSTem:ERRor:CODE:ALL`

- ログの消去
  LOGCLEAR
  :MMEMory:LOG:CLEar
- ログの取得
  LOGREAD
  :MMEMory:LOG:LOAD
- ログサイズの取得
  LOGSIZE
  :MMEMory:LOG:SIZE
- ファームウェアの情報取得
  FIRMCHECK
  :SYSTem:BASE:UPDate:FIRMware:CHECk
- ファームウェアの更新
  FIRMUPDATE
  :SYSTem:BASE:UPDate:FIRMware
- ライセンスキーインストール前の情報取得
  KEYCHECK
  :SYSTem:BASE:LICense:CHECk
- ライセンスキーのインストール
  KEYINST
  :SYSTem:BASE:LICense:INSTall
- パラメータの初期化
  *RST
- 再起動
  REBOOT
  :SYSTem:REBoot
- シャットダウン
  SHUTDOWN
  :SYSTem:SHUTdown
- アプリケーションソフトウェアの切り替え
  SYSSEL
  :INSTrument[:SELect]
  STDSEL
  :CONFigure:CELLular:MEASurement:STANdard
  アプリケーションソフトウェアの切り替えに関するコマンドの説明は,
  使用するアプリケーションソフトウェアの取扱説明書を参照してください。

3.4.2 製品情報の確認

確認できる製品情報には，下記があります。

- MT8870A ユニバーサルワイヤレステストセットの情報
  製品形名，製造番号，ハードウェアオプション，FPGA バージョン，ソフトウェアライセンス，ソフトウェアオプション，内部温度，電源投入回数，稼働時間
- MU887002A 送受信テストモジュールの情報
  製品形名，製造番号，TRX1/2，ハードウェアオプション，FPGA バージョン，電源投入回数，稼働時間，挿抜回数，内部温度，MAC アドレス
- モジュールが装着されている MT8870A のスロット番号
- パッケージバージョン
- ソフトウェアライセンスの期限

MT8870A ユニバーサルワイヤレステストセットの情報を確認するには，以下のコマンドを使用します。

- 製品形名，製造番号，メーカ名
  `*IDN`
- 製品形名，製品名，ハードウェアオプション，FPGA バージョン，ソフトウェアライセンス，ソフトウェアオプション
  `MFINFO`
  `:SYSTem:INFormation:MAINframe`
- 製造番号
  `MFSERIAL`
  `:SYSTem:INFormation:MAINframe:DEVice:ID`
- 内部温度
  `MFTEMP`
  `:SYSTem:INFormation:MAINframe:TEMPerature`
- 電源投入回数
  `MFPOWERONCNT`
  `:SYSTem:INFormation:MAINframe:POWeron:COUNt`
- 稼働時間
  `MFTMCNT`
  `:SYSTem:INFormation:MAINframe:RTIMe`
- ファームウェアバージョン
  `MCFV`
  `:SYSTem:INFormation:MAINframe:PACKage:VERSion`
- ソフトウェア形名
  `MCSOPT`
  `:SYSTem:INFormation:MAINframe:SOFTware`
- ソフトウェアのオプション番号
  `MCOPT`
  `:SYSTem:INFormation:MAINframe:SOFTware:OPTion`
- 波形ライセンス
  `MCWOPT`
  `:SYSTem:INFormation:MAINframe:WAVeform`

MU887002A 送受信テストモジュールの情報を確認するには，以下のコマンドを使用します。

- 製品形名，製品名，ハードウェアオプション，FPGA バージョン
  SYSINFO
  :SYSTem:INFormation
- 製造番号
  SYSSERIAL
  :SYSTem:INFormation:DEVice:ID
- TRX 1/2
  SYSTRX
  :SYSTem:INFormation:MAINframe:TRX
- 内部温度
  SYSTEMP
  :SYSTem:INFormation:TEMPerature
- 電源投入回数
  SYSPOWERONCNT
  :SYSTem:INFormation:POWeron:COUNt
- 稼働時間
  SYSTMCNT
  :SYSTem:INFormation:RTIMe
- 挿抜回数
  SYSPLUGCNT
  :SYSTem:INFormation:PLUG:COUNt
- MAC アドレス
  MACADDR
  :SYSTem:COMMunicate:NET:HWADdress

モジュールが装着されている MT8870A のスロット番号は，以下のコマンドで確認します。

- スロット番号
  SYSSLOT
  :SYSTem:INFormation:SLOT

パッケージバージョンは，次のソフトウェアの組み合わせに対する管理番号です。

- MT8870A の内蔵プログラム
- MU887002A の内蔵プログラム
- MU887002A にインストールされるアプリケーションソフトウェア

MU887002A 送受信テストモジュールの情報を確認するには，以下のコマンドを使用します。

SYSVER
:SYSTem:VERSion

MT8870A にインストールされるアプリケーションソフトウェアは，ソフトウェアライセンスとして提供されます。ソフトウェアライセンスには有効期限が設定される場合があります。使用しているソフトウェアライセンスの有効期限は，以下のコマンドで確認できます。

- ライセンスキーの情報取得
  KEYINFO
  :SYSTem:BASE:LICense:INFormation

3.5　測定前の準備

3.5.1　周波数の設定

モジュールの周波数基準信号は，次から選択できます。この基準信号の設定は，MT8870A に装着されるすべてのモジュールで共通です。

- MT8870A 内蔵の信号（内部基準信号）
- MT8870A のコネクタ（Ref Input）に入力される信号（外部基準信号）

周波数基準信号を設定するには，次のコマンドを使用します。

```
FREQREF
:SYSTem:BASE:REFerence:FREQuency
REF
:SYSTem:BASE:REFerence:FREQuency:CONFigure
```

使用している基準信号の種類，および MT8870A 内部回路が周波数基準信号に対して位相同期が取れているかの情報は，次のコマンドで確認できます。
また，MT8870A の Ext. Reference ランプでも確認できます。
位相同期が取れていない場合，周波数確度が規格外になることがあります。

```
FREQREFSOURCE
:SYSTem:BASE:REFerence:FREQuency:SOURce
EXTREF
:SYSTem:BASE:REFerence:FREQuency:CONFigure:SOURce
```

内部基準信号を選択している場合，次のコマンドで周波数を補正できます。
モジュールの周波数を変化させる場合に使用します。

```
ADJREFCLK
:SYSTem:BASE:REFerence:FREQuency:ADJust
```

周波数の補正値を工場出荷時の値に戻す場合，次のコマンドで工場出荷時の補正値を確認できます。
このコマンドを実行しただけでは，周波数の補正値は工場出荷時の値に戻りません。工場出荷時の値に戻すには，読み出した値を上記のコマンドで再設定してください。

```
ADJREFCLKDEF
:SYSTem:BASE:REFerence:FREQuency:ADJust:DEFault
```

3.5.2　レベル校正

モジュールの校正では，入力レベルと出力レベルに対するレベル確度の周波数特性をフラットにし，内部の温度変化によるレベル確度のずれを調整します。

次の場合に校正を実行します。
- 電源を入れてウォーミングアップした後
- テストする端末を替えたとき
- 周囲温度が大きく温度が変化したとき

この節で説明している校正はモジュール自身で行うもので，全バンド校正とバンド校正の 2 種類があります。
全バンド校正は，モジュールの RF 入出力全帯域で校正を行います。
バンド校正は，アプリケーションソフトウェアが対応している通信システムの帯域で校正を行います。バンド校正は，全バンド校正よりも短い時間で終了します。

注:

電源を入れて 60 分以上経過してから全バンド校正を実行してください。全バンド校正を行った後は，温度変動時，および移動機の入れ替えなど，測定の区切りにバンド校正を実行してください。

全バンド校正の実行，情報の取得には，次のコマンドを使用します。

全バンド校正の実行
```
FULLCAL
:CALCulate:CALibration:FULL:STARt
```
全バンド校正実施時刻の取得
```
FULLCAL_TM
:CALCulate:CALibration:FULL:LASTtime
```
全バンド校正が失敗した時刻の取得
```
FULLCALERR_TM
:CALCulate:CALibration:FULL:ERRor:LASTtime
```
全バンド校正が失敗した回数の取得
```
FULLCALERRCNT
:CALCulate:CALibration:FULL:ERRor:COUNt
```
全バンド校正結果の取得
```
FULLCALRESULT
:CALCulate:CALibration:FULL:RESult
```
全バンド校正時からの経過時刻の取得
```
CALTIME
:CALCulate:CALibration:ELAPsed:TIME
```
全バンド校正時の温度の取得
```
CALTEMP
:CALCulate:CALibration:TEMPerature
```

バンド校正の実行，情報の取得には，次のコマンドを使用します。

バンド校正の実行

BANDCAL

:CALCulate:CALibration:BAND:STARt

内部温度の変化に対応してバンド校正の実行

BANDCAL_TEMP

:CALCulate:CALibration:BAND:STARt:TEMPerature

バンド校正を実施時刻の取得

BANDCAL_TM

:CALCulate:CALibration:BAND:LASTtime

バンド校正が失敗した時刻の取得

BANDCALERR_TM

:CALCulate:CALibration:BAND:ERRor:LASTtime

バンド校正の失敗回数

BANDCALERRCNT

:CALCulate:CALibration:BAND:ERRor:COUNt

バンド校正結果の取得

BANDCALRESULT

:CALCulate:CALibration:BAND:RESult

3.5.3　ケーブルの損失補正

MU887002A は，モジュールのコネクタに接続する同軸ケーブルの損失補正データを 16 個保持できます。
補正データは，ポート別に周波数対損失のテーブル形式で保持されます。

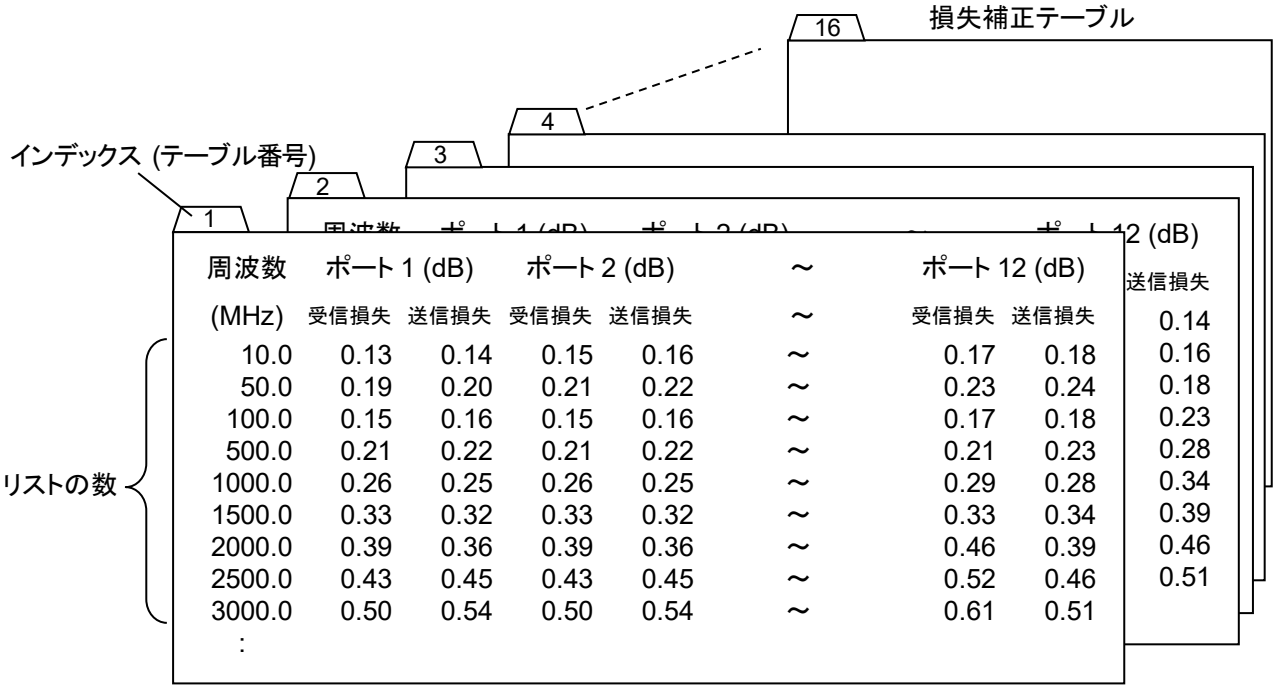


図 3.5.3-1　損失補正データの構成

各ポートの損失補正データには，次の 2 種類があります。

- MU887002A が受信する信号（アップリンク）に対する補正データ（受信損失）
- MU887002A が送信する信号（ダウンリンク）に対する補正データ（送信損失）

方向性結合器など方向によって損失が違う部品が測定系に存在する場合に，信号の方向別に補正係数を設定できます。

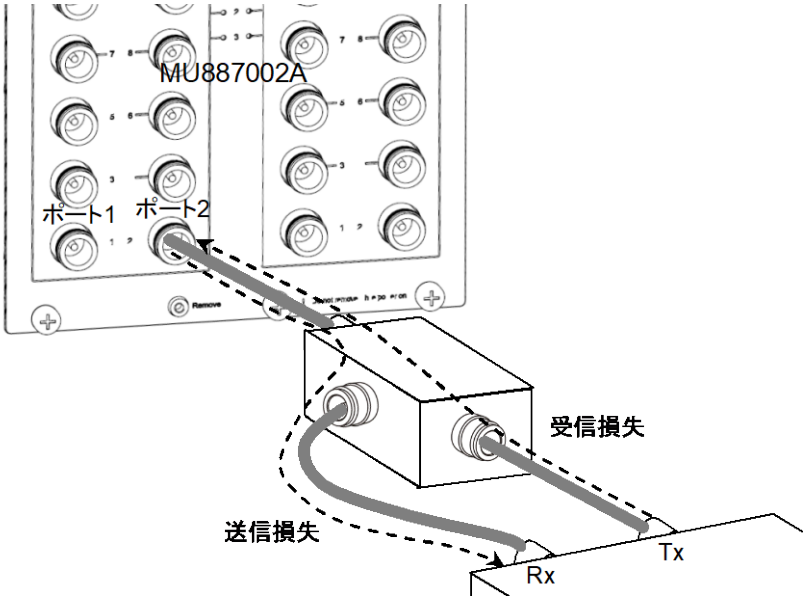


図 3.5.3-2　損失補正データの構成

次のコマンドで損失補正データの記載内容を編集します。

指定した周波数の Port 1～Port 4 の外部損失を設定

LOSSTBLVAL

:CALCulate:EXTLoss:TABLe:VALue

指定した周波数の Port 1～Port 12 の外部損失を設定

LOSSTBLVALALL

:CALCulate:EXTLoss:TABLe:VALue:ALL

外部損失データを削除

DELLOSSTBL

:CALCulate:EXTLoss:TABLe:DELete

外部損失のデータ数を問い合わせる

LOSSTBLCNT

:CALCulate:EXTLoss:TABLe:COUNt

16 個の損失補正データのうち，適用する損失補正テーブルのインデックスを次のコマンドで選択します。

LOSSTBL

:CALCulate:EXTLoss:TABLe:SETTing

全ポートの損失補正をオン/オフします。

EXTLOSSW

[:ROUTe]:EXTLoss:TABLe:SWITch

複数のテストポートから信号出力する場合，テストポート間の損失補正テーブル値の差が 12 dB を超えると，信号は設定した出力レベルよりも低く出力されます。この場合，ランプ 2 が橙色に点灯します。

ただし，8 dB を超えた場合でも，出力レベルの誤差が大きくなる可能性があります。

3.5.4　ポートの設定

信号の入力/出力に使用する正面パネルのテストポートを次のコマンドで指定します。

```
PORT
:ROUTe:PORT:CONNect:DIRection
```

- 入力: MU887002A に信号を入力するテストポートを設定します。
- 出力: MU887002A から信号を出力するテストポートを設定します。
- オフ: 全テストポート入力オフを設定します。

入力に設定できるテストポートは常に 1 TRX に 1 個のみです。
出力に設定できるテストポートは 1 TRX に最大 12 個です。
すべてのテストポートを入力と出力の両方に設定できますが, 同時に入力と出力に設定できるのは 1 つの TRX に 1 個だけです。
ただし, MU887002A-007 搭載時,

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では, 入力または出力のどちらかにしか設定できません。

複数のテストポートから信号出力する場合, テストポート間の損失補正テーブル値の差が 12 dB を超えると, 信号は設定した出力レベルよりも低く出力されます。この場合, ランプ 2 が橙色に点灯します。
ただし, 8 dB を超えた場合でも, 出力レベルの誤差が大きくなる可能性があります。

3.5.5　トリガコネクタの設定

測定中に背面パネルのトリガコネクタを使用する場合, コネクタの入力/出力設定, トリガ信号の種類を次のコマンドで指定します。

トリガコネクタの入出力

```
MFTRIGPORT
:ROUTe:TRIGger:PORT
```

トリガ入力信号の種類

```
SYSTRIGINSRC
:ROUTe:TRIGger:INPut:SOURce
```

トリガ出力信号の種類

```
SYSTRIGOUTSRC
:ROUTe:TRIGger:OUTPut:SOURce
```

トリガコネクタの入出力は, モジュール別, TRX 別に設定します。

3.6　ステータスレジスタ

モジュールには，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。

3.6.1　レジスタの構成

レジスタの構成を図 3.6.1-1，および図 3.6.1-2 に示します。

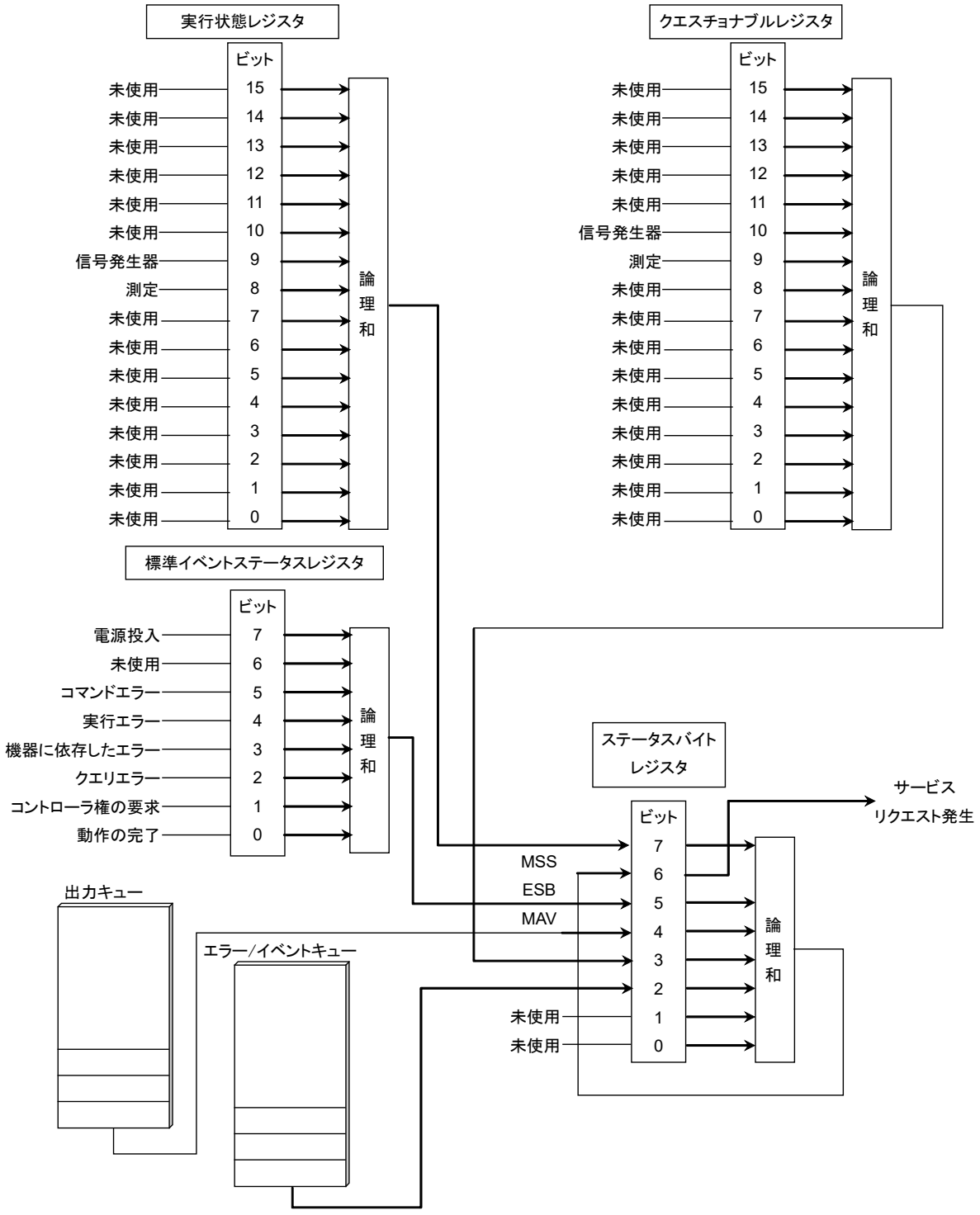


図 3.6.1-1　レジスタの構成（SCPI コマンドモード）

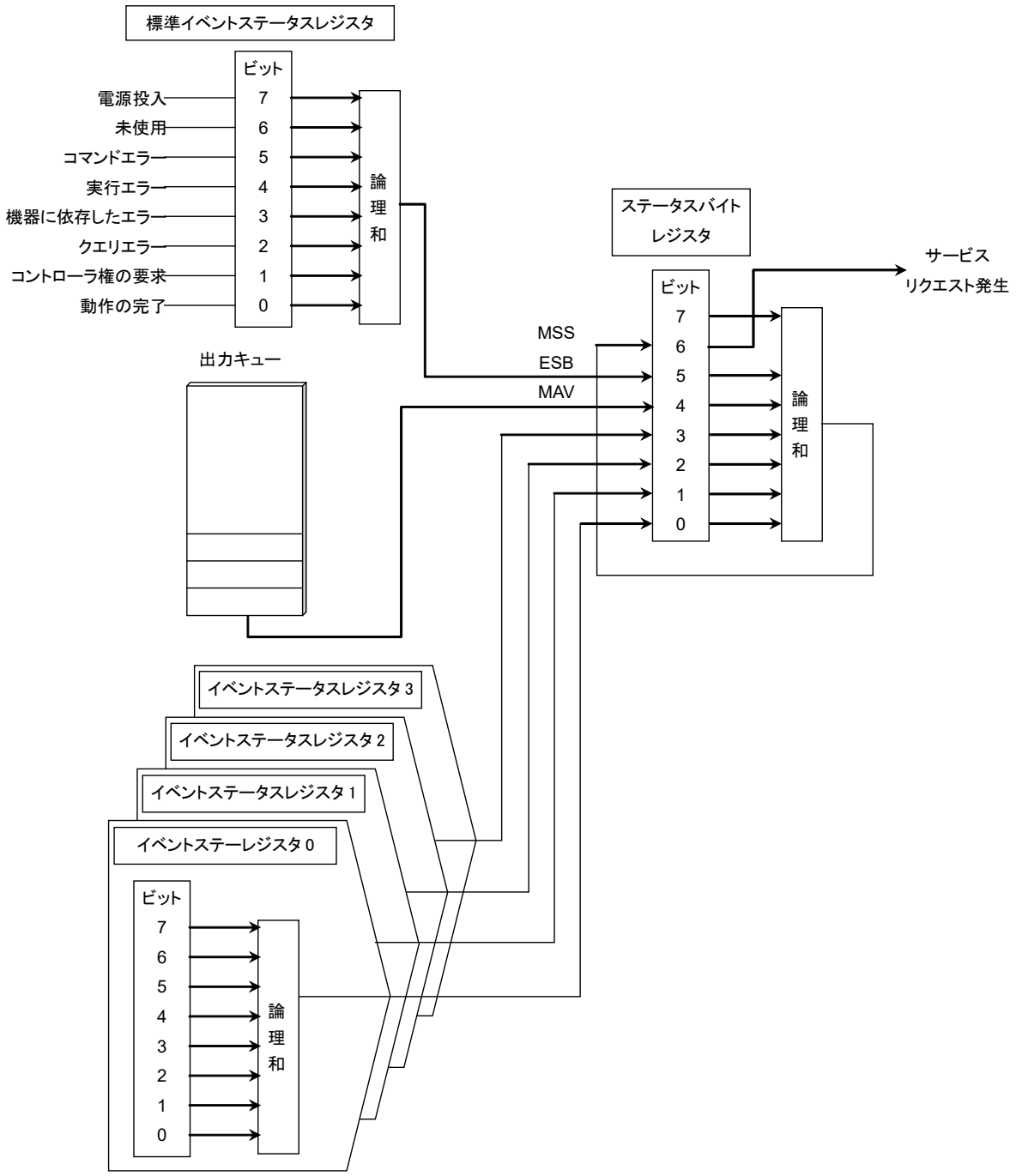


図 3.6.1-2　レジスタの構成 (Native コマンドモード)

各レジスタは 8 ビットまたは 16 ビットのデータを持ちます。レジスタの出力値は，表 3.6.1-1 に示す各ビットの 10 進数の値を合計した値です。

表 3.6.1-1　レジスタのビットの 10 進数変換値

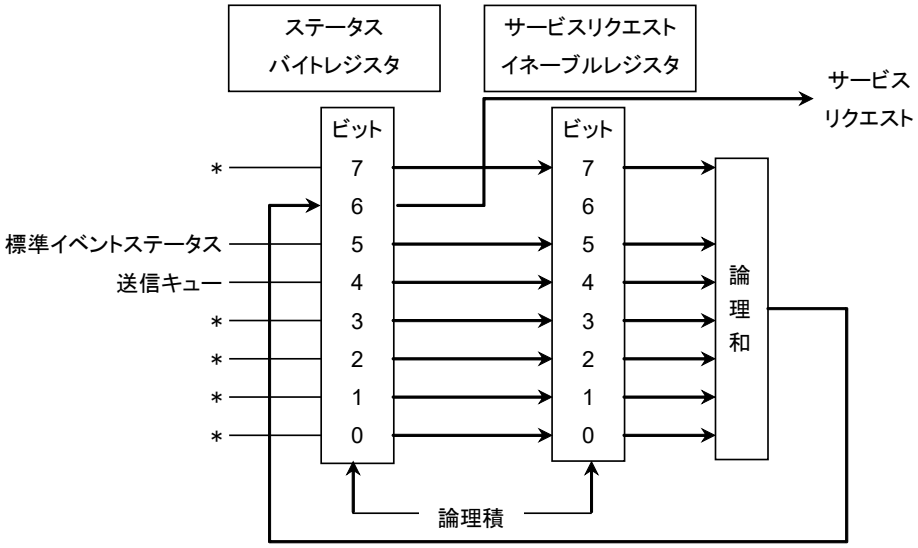
ビット	10 進数の値	ビット	10 進数の値
0	1	8	256
1	2	9	512
2	4	10	1024
3	8	11	2048
4	16	12	4096
5	32	13	8192
6	64	14	16384
7	128	15	32768

各レジスタにはビットイネーブルレジスタが対になって用意されています。

3.6.2　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，制御用コンピュータに対して割り込みを発生することができます。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。



＊：ビット7，3～0の信号はコマンドモードによって変わります。

図 3.6.2-1　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，イーサネットでは使用できません。

ステータスバイトレジスタは，次の方法で読み取ります。

- 共通コマンドの*STB?を使用する

サービスリクエストイネーブルレジスタの設定と読み取りには，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 3.6.2-1　ステータスバイトレジスタのビット定義

ビット	説明
7	SCPI コマンドモード：実行状態レジスタ Native コマンドモード：未使用　常に 0 になります。
6	MSS (Master Summary Status) 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	標準イベントステータスレジスタ
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	SCPI コマンドモード：クエスチョナブルステータスレジスタ Native コマンドモード：イベントステータスレジスタ 3
2	SCPI コマンドモード：測定器のエラーイベントキューに，データがあるときに 1 になります。 Native コマンドモード：イベントステータスレジスタ 2
1	SCPI コマンドモード：未使用　常に 0 になります。 Native コマンドモード：イベントステータスレジスタ 1
0	SCPI コマンドモード：未使用　常に 0 になります。 Native コマンドモード：イベントステータスレジスタ 0

ステータスバイトレジスタのビット 7 は，実行状態レジスタの情報を示します。
情報の詳細は，「3.6.5 実行状態レジスタ」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 6 は，マスターサマリーステータス (MSS) と呼びます。MSS が 1 のとき，モジュールから制御用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 は標準イベントステータスレジスタの情報を示します。
情報の詳細は，「3.6.3 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 7，5 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。

3.6.3　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。

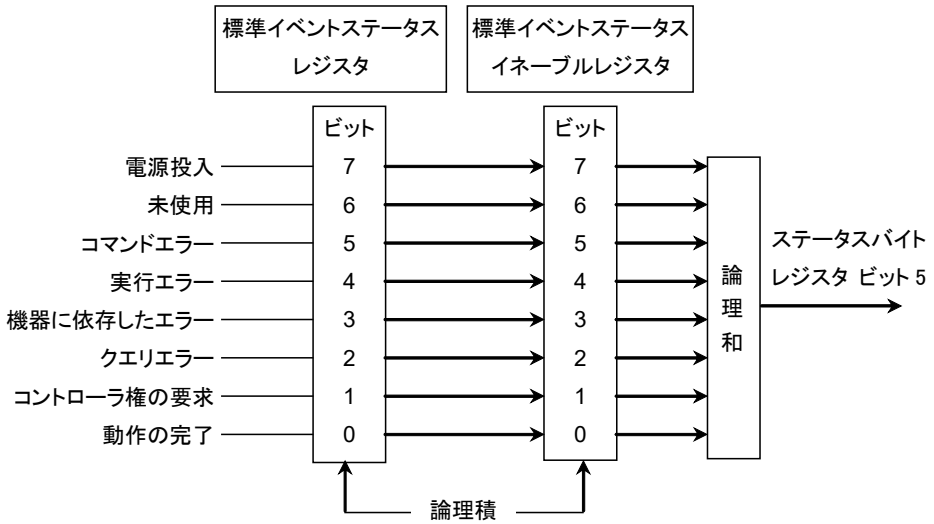


図 3.6.3-1　標準イベントステータスレジスタの構成

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 3.6.3-1　標準イベントステータス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。一度読み取ると 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	機器に依存したエラー コマンドエラー，実行エラーおよび問い合わせエラー以外のエラーが発生したときに 1 になります。
2	クエリエラー レスポンスデータが無いときに読み取り処理をした場合，レスポンデータがキューからオーバーフローして失われた場合に 1 になります。
1	コントローラ権の要求 モジュールがコントローラになることを要求する場合に 1 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 0 は，共通コマンド*OPC で読み取ることができます。
標準レジスタは，共通コマンド*CLS で 0 にできます。

3.6.4　イベントステータスレジスタ

イベントステータスレジスタには，標準イベントステータスレジスタと同様にイネーブルレジスタがあります。
モジュールのイベントステータスレジスタ 0～3 には，次の機能が有ります。

0: 終了イベントステータスレジスタ（信号発生器）
1: エラーイベントステータスレジスタ（信号発生器）
2: 終了イベントステータスレジスタ（測定）
3: エラーイベントステータスレジスタ（測定）

表 3.6.4-1　終了イベントステータスレジスタ（信号発生器）のビット定義

ビット	説明
7～1	未使用　常に 0 になります。
0	ファイル読み取り中に 1 になります。

表 3.6.4-2　エラーイベントステータスレジスタ（信号発生器）のビット定義

ビット	説明
7～1	未使用　常に 0 になります。
0	読み取ったファイルにエラーがある場合に 1 になります。

表 3.6.4-3　終了イベントステータスレジスタ（測定）のビット定義

ビット	説明
7～1	未使用　常に 0 になります。
0	測定実行中に 1 になります。

表 3.6.4-4　エラーイベントステータスレジスタ（測定）のビット定義

ビット	説明
7～3	未使用　常に 0 になります。
2	時間内に測定できなかった場合（タイムアウト）に 1 になります。
1	測定結果がレベルアンダーの場合に 1 になります。
0	測定結果がレベルオーバーの場合に 1 になります。

各レジスタの設定，問い合わせには次のコマンドを使用します。
終了イベントステータスレジスタ（信号発生器）: `ESE0,ESR0`
エラーイベントステータスレジスタ（信号発生器）:`ESE1,ESR1`
終了イベントステータスレジスタ（測定）:　`ESE2,ESR2`
エラーイベントステータスレジスタ（測定）:　`ESE3,ESR3`

3.6.5　実行状態レジスタ

実行状態レジスタ（OSR）は，次のレジスタで構成されます。

- 実行状態コンディションレジスタ
- 遷移フィルタ
- 実行状態イベントレジスタ
- 実行状態イネーブルレジスタ（OSER）

実行状態コンディションレジスタは，状態の変化を表示します。状態が変化すると，このレジスタの値も変化します。

実行状態イベントレジスタは，実行状態コンディションレジスタの値の変化を記録します。実行状態イベントレジスタの前には，書き込む条件を定義する遷移フィルタがあります。遷移フィルタは，次のどれかに条件にあてはまったときに実行状態イベントレジスタのビットを 1 にします。

- ビットが 0 から 1 に変化したとき
- ビットが 1 から 0 に変化したとき
- ビットが 0 から 1 に変化したとき，またはビットが 1 から 0 に変化したとき

実行状態イネーブルレジスタは，実行状態イベントレジスタの出力をビットごとに設定します。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 7 に出力されます。

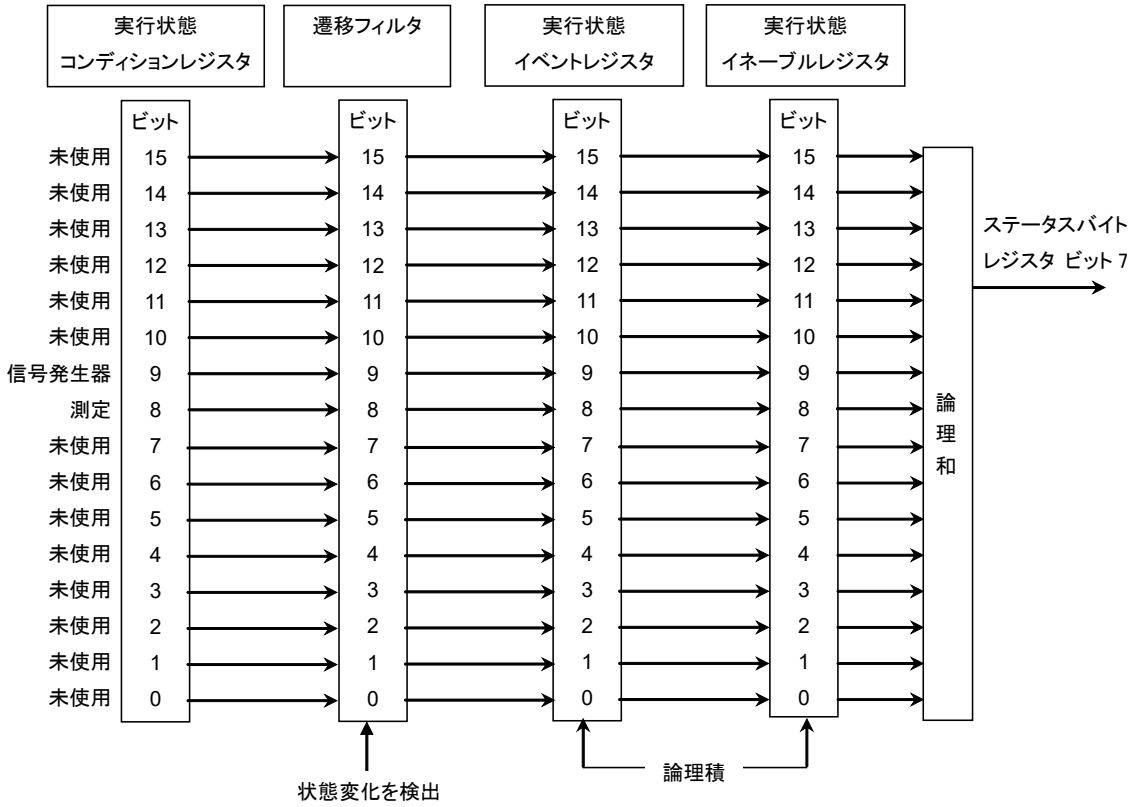


図 3.6.5-1　実行状態レジスタの構成

実行状態コンディションレジスタの，各ビットの定義は次のとおりです。

表 3.6.5-1　実行状態コンディションレジスタのビット定義

ビット	説明
15～10	未使用　常に 0 になります。
9	信号発生器実行状態レジスタの論理和
8	測定実行状態レジスタの論理和
7～0	未使用　常に 0 になります。

実行状態コンディションレジスタのビット 9，ビット 8 には実行状態レジスタと同じ構成を持った信号発生器実行状態レジスタ，測定実行状態レジスタの論理和が設定されます。

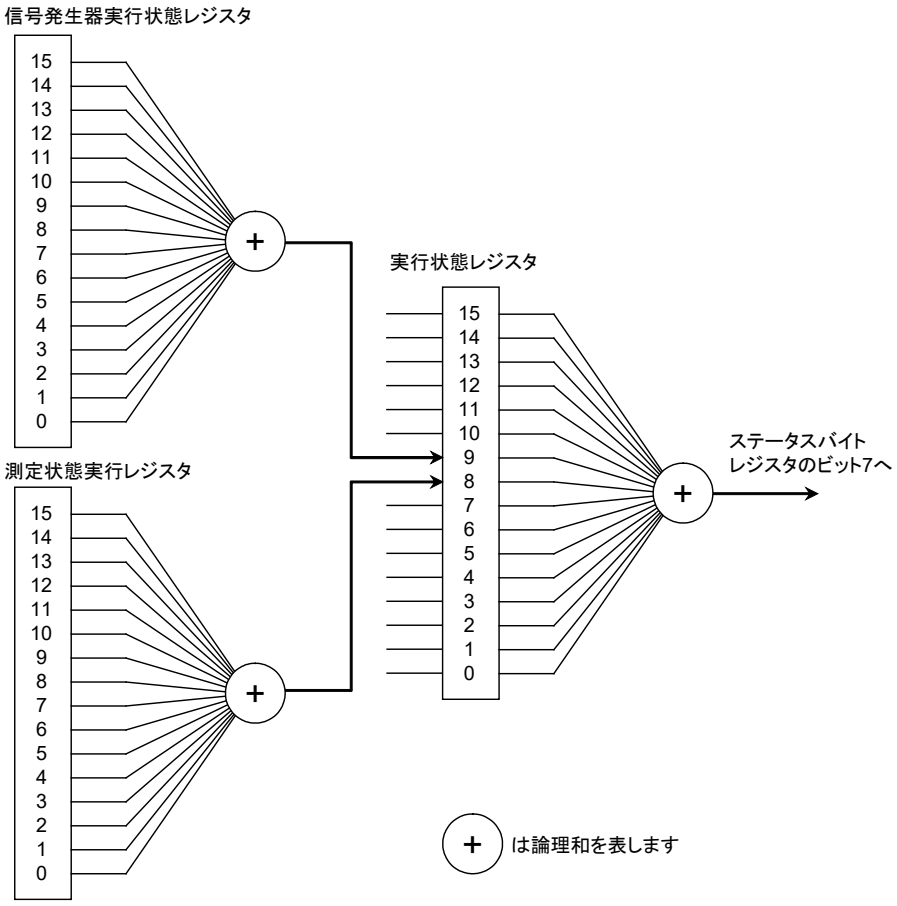


図 3.6.5-2　実行状態レジスタの接続図

信号発生器実行状態レジスタ，測定実行状態レジスタのビット定義を次の表に示します。
未使用のビットは，アプリケーションソフトウェアで機能を割り当てられることがあります。

表 3.6.5-2　信号発生器実行状態レジスタのビット定義

ビット	説明
15～3	未使用　常に 0 になります。
2	波形パターンの同期出力準備中に 1 になります。
1	デフラグ実行中に 1 になります。
0	ファイル読み取り中に 1 になります。

表 3.6.5-3　測定実行状態レジスタのビット定義

ビット	説明
15～1	未使用　常に 0 になります。
0	測定実行中に 1 になります。

実行状態レジスタの設定，問い合わせには次のコマンドを使用します。

```
:STATus:OPERation[:EVENt]
:STATus:OPERation:CONDition
:STATus:OPERation:ENABle
:STATus:OPERation:NTRansition
:STATus:OPERation:PTRansition
```

信号発生器実行状態レジスタの設定，問い合わせには次のコマンドを使用します。

```
:STATus:OPERation:GENerator[:EVENt]
:STATus:OPERation:GENerator:CONDition
:STATus:OPERation:GENerator:ENABle
:STATus:OPERation:GENerator:NTRansition
:STATus:OPERation:GENerator:PTRansition
```

測定実行状態レジスタの設定，問い合わせには次のコマンドを使用します。

```
:STATus:OPERation:MEASure[:EVENt]
:STATus:OPERation:MEASure:CONDition
:STATus:OPERation:MEASure:ENABle
:STATus:OPERation:MEASure:NTRansition
:STATus:OPERation:MEASure:PTRansition
```

3.6.6　クエスチョナブルステータスレジスタ

クエスチョナブルステータスコンディションレジスタは，次の可能性がある場合に変化します。

- 機器が出力する信号が正しく設定されていない。
- 測定結果が正しくない。

たとえば，実際に出力される信号のレベルや周波数が設定した値どおりでない場合，オーバーレンジやアンダーレンジで正しくレベルが測定されなかった場合に，クエスチョナブルレジスタのビットが設定されます。

クエスチョナブルレジスタは，実行状態レジスタと同様に次のレジスタで構成されます。

- クエスチョナブルステータスコンディションレジスタ
- 遷移フィルタ
- クエスチョナブルステータスイベントレジスタ
- クエスチョナブルステータスイネーブルレジスタ（QER）

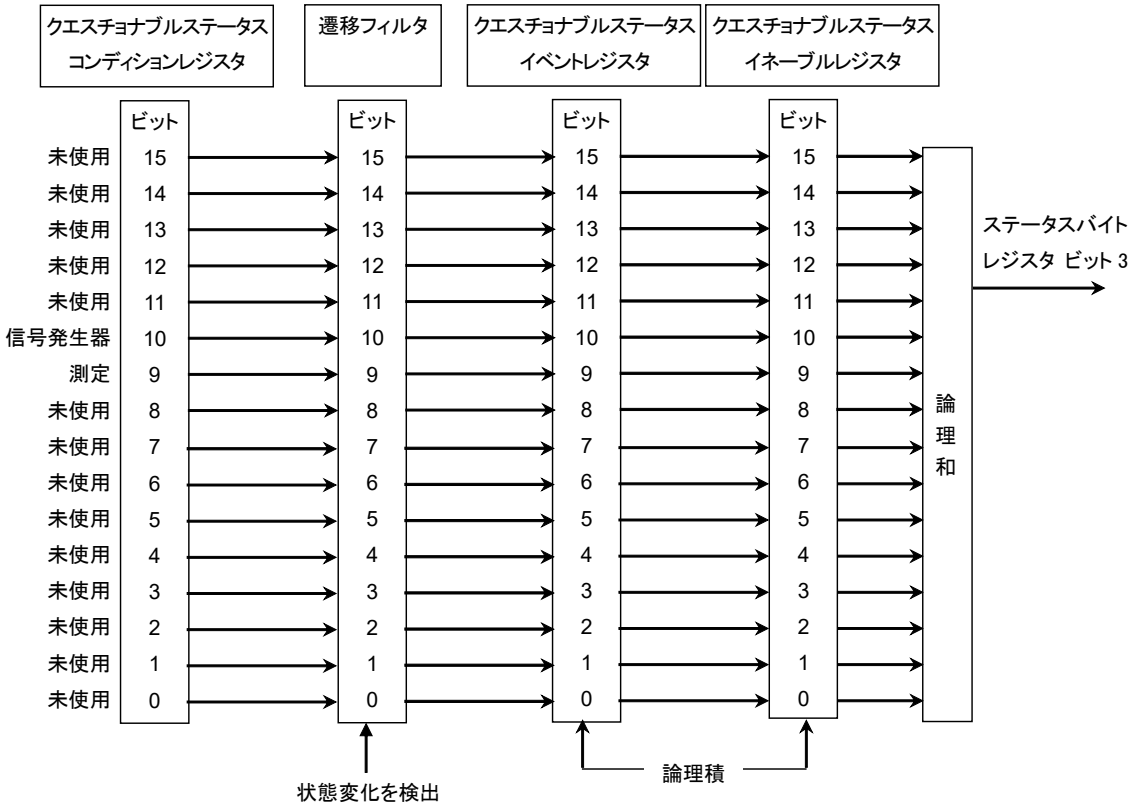


図 3.6.6-1　クエスチョナブルステータスレジスタの構成

クエスチョナブルステータスコンディションレジスタの，各ビットの定義は次のとおりです。

表 3.6.6-1　クエスチョナブルステータスコンディションレジスタのビット定義

ビット	説明
15～11	未使用　常に 0 になります。
10	信号発生器クエスチョナブルステータスレジスタの論理和
9	測定クエスチョナブルステータスレジスタの論理和
8～0	未使用　常に 0 になります。

クエスチョナブルステータスコンディションレジスタのビット 10，ビット 9 にはクエスチョナブルステータスレジスタと同じ構成を持った信号発生器クエスチョナブルステータスレジスタ，測定クエスチョナブルステータスレジスタの論理和が設定されます。

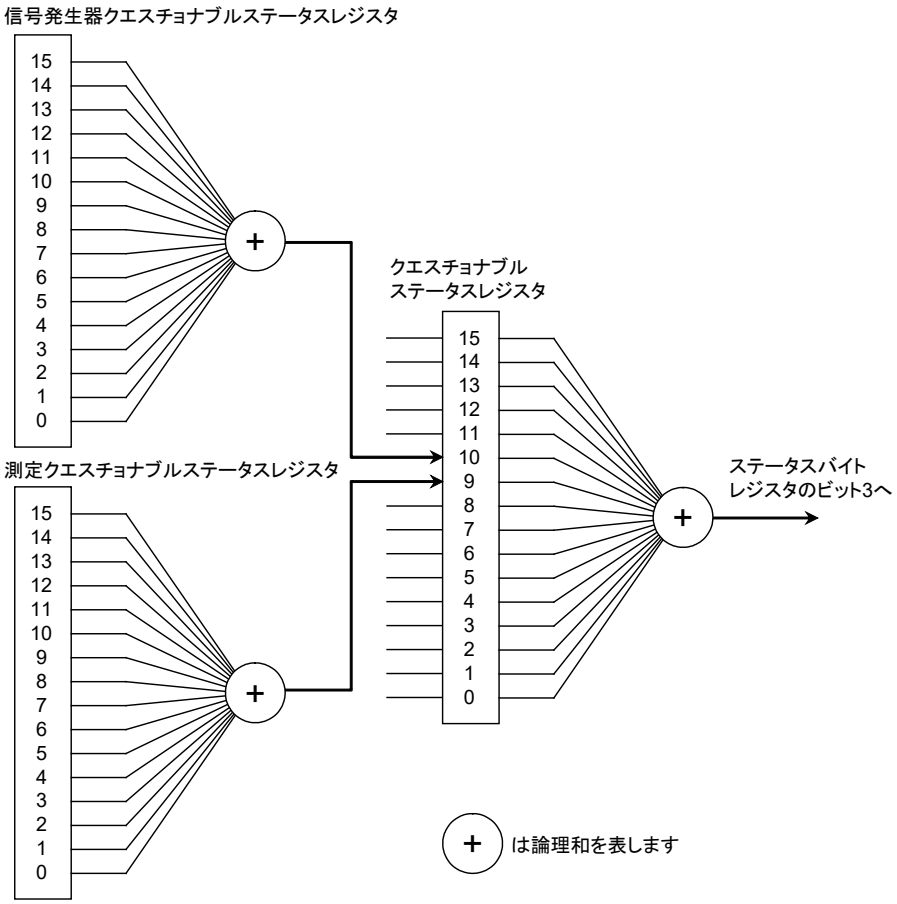


図 3.6.6-2　クエスチョナブルステータスレジスタの接続図

信号発生器クエスチョナブルステータスレジスタ，測定クエスチョナブルステータスレジスタのビット定義を次の表に示します。
未使用のビットは，アプリケーションソフトウェアで機能を割り当てられることがあります。

表 3.6.6-2　信号発生器クエスチョナブルステータスレジスタのビット定義

ビット	説明
15～1	未使用　常に 0 になります。
0	読み取ったファイルにエラーがある場合に 1 になります。

表 3.6.6-3　測定クエスチョナブルステータスレジスタのビット定義

ビット	説明
15～3	未使用　常に 0 になります。
2	時間内に測定できなかった場合（タイムアウト）に 1 になります。
1	測定結果がレベルアンダーの場合に 1 になります。
0	測定結果がレベルオーバーの場合に 1 になります。

クエスチョナブルステータスレジスタの設定，問い合わせには次のコマンドを使用します。

```
:STATus:QUEStionable[:EVENt]
:STATus:QUEStionable:CONDition
:STATus:QUEStionable:ENABle
:STATus:QUEStionable:NTRansition
:STATus:QUEStionable:PTRansition
```

信号発生器クエスチョナブルステータスレジスタの設定，問い合わせには次のコマンドを使用します。

```
:STATus:QUEStionable:GENerator[:EVENt]
:STATus:QUEStionable:GENerator:CONDition
:STATus:QUEStionable:GENerator:ENABle
:STATus:QUEStionable:GENerator:NTRansition
:STATus:QUEStionable:GENerator:PTRansition
```

測定クエスチョナブルステータスレジスタの設定，問い合わせには次のコマンドを使用します。

```
:STATus:QUEStionable:MEASure[:EVENt]
:STATus:QUEStionable:MEASure:CONDition
:STATus:QUEStionable:MEASure:ENABle
:STATus:QUEStionable:MEASure:NTRansition
:STATus:QUEStionable:MEASure:PTRansition
```

第4章 ベクトル信号発生器

この章では，MU887002A 送受信テストモジュールをベクトル信号発生器として使用する方法について説明します。
コマンドの詳細は「第 5 章 SCPI コマンドリファレンス」を参照してください。

4.1 ベクトル信号発生器 (VSG) の紹介 4-2
4.2 動作モード 4-3
4.3 ノーマルモード....4-4
4.3.1 基本設定 4-4
4.3.2 変調 4-6
4.3.3 TRX間の波形パターン同期....4-10
4.4 シーケンスモード 4-15
4.4.1 動作内容 4-15
4.4.2 シーケンステーブルの構成....4-16
4.4.3 ファイル操作 4-21
4.4.4 設定項目のエラーチェック 4-22
4.4.5 シーケンスの実行....4-23
4.4.6 TRX間の波形パターン同期....4-24

4.1　ベクトル信号発生器（VSG）の紹介

信号発生器（SG: Signal Generator）は，周波数，レベル，変調信号を設定して試験用信号を発生する装置です。特に位相と振幅を変調できる信号発生器をベクトル信号発生器（VSG: Vector Signal Generator）と呼びます。

信号発生器の出力電圧 V は次の式で表されます。

$$V = R\sin(2\pi ft + \varphi)$$

R:　振幅
f:　周波数（Hz）
φ:　位相差（rad.）

VSG では，位相がπ/2（rad.）違っている正弦波信号を合成して位相変調を実現しています。

$$R\sin(2\pi ft + \varphi) = I\cos(2\pi ft) + Q\sin(2\pi ft)$$

$$I = R\cos(\varphi),\ Q = R\sin(\varphi)$$

位相変調の振幅成分をそれぞれ I (In phase)，Q (Quadrature phase) と呼びます。位相変調するための IQ 振幅信号をベースバンド信号と呼びます。

MU887002A 送受信テストモジュールでは，波形ファイルに保存されているベースバンド信号を再生することにより位相変調をかけます。

波形ファイルの仕様については下記を参照してください。
『セルラアプリケーション用波形ファイル 取扱説明書』
『SRW アプリケーション用波形ファイル 取扱説明書』
『GNSS & Broadcast 用波形ファイル 取扱説明書』

4.2　動作モード

VSG には，2 つの動作モードがあります。

- ノーマルモード
  リモート制御によって，信号発生器として使用できます。
- シーケンスモード
  シーケンス機能を実行するモードです。
  シーケンスを実行している状態（Active）と，停止している状態（Inactive）があります。

設定できるモード間の変更を次の図に示します。

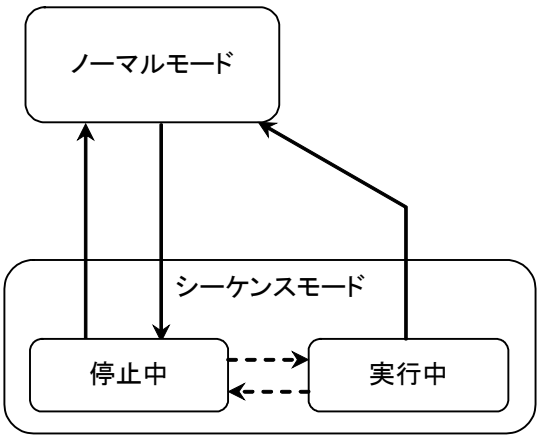


図4.2-1　設定可能な動作モードの変更方向

動作モードを設定するには，:SOURce:GPRF:GENerator:MODE コマンドを使用します。
シーケンスモードの実行/停止を切り替えるには，次のコマンドを使用します。

```
:SOURce:GPRF:GENerator:SEQuence:EXECute
:SOURce:GPRF:GENerator:SEQuence:CANCel
```

なお，Native コマンドモードで VSG を制御する場合は，SCPI コマンドの省略形を使用します。SCPI コマンドの先頭のコロンは付けません。

SCPI コマンドモード

```
:SOURce:GPRF:GENerator:SEQuence:CANCel
```

Native コマンドモード

```
SOUR:GPRF:GEN:SEQ:CANC
```

4.3　ノーマルモード

4.3.1　基本設定

VSG の基本設定をするには，次のコマンドを送信してノーマルモードにします。

```
:SOURce:GPRF:GENerator:MODE NORMAL
```

VSG の基本設定項目は次のとおりです。

- 信号を出力するポートの選択
- 信号出力のオン/オフ
- 周波数
- レベル
- 変調のオン/オフ
- スペクトル反転
- AWGN 付加のオン/オフ
- AWGN のレベル
- 波形パターンの再生

変調波形データ以外の設定には次のコマンドを使用します。変調波形データのコマンドは「4.3.2 変調」を参照してください。

- 出力制御
  出力のオン/オフ
  `:SOURce:GPRF:GENerator:STATe`
  ポートの選択
  `:ROUTe:PORT:CONNect:DIRection`
  `PORT`（Native コマンド）
- 周波数
  `:SOURce:GPRF:GENerator:RFSettings:FREQuency`
- レベル
  設定
  `:SOURce:GPRF:GENerator:RFSettings:LEVel`
  出力範囲確認
  `:SOURce:GPRF:GENerator:RFSettings:LEVel:SETTing`
- 変調
  変調のオン/オフ
  `:SOURce:GPRF:GENerator:BBMode`
  スペクトル反転
  `:SOURce:GPRF:GENerator:RFSettings:DM:POLarity`
  AWGN 付加のオン/オフ
  `:SOURce:GPRF:GENerator:ARB:NOISe:STATe`
  AWGN のレベル
  `:SOURce:GPRF:GENerator:ARB:NOISe:CN`
  波形パターンの再生
  `:SOURce:GPRF:GENerator:ARB:WAVeform:RESTart`

例として，周波数 2000 MHz，レベル–10 dBm の連続信号（CW）をポート 1 に出力する手順を示します。

1. 動作モードをノーマルモードに設定します。

   ```
   :SOURce:GPRF:GENerator:MODE NORMAL
   ```

2. 周波数を 2000 MHz に設定します。

   ```
   :SOURce:GPRF:GENerator:RFSettings:FREQuency 2GHZ
   ```

3. レベルを–10 dBm に設定します。

   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel -10
   ```

4. AWGN のレベルを 30 dB に設定します。

   ```
   :SOURce:GPRF:GENerator:ARB:NOISe:CN 30
   ```

5. 変調を無し（CW）に設定します。

   ```
   :SOURce:GPRF:GENerator:BBMode CW
   ```

6. ポート 1 を出力に設定します。

   ```
   :ROUTe:PORT:CONNect:DIRection PORT1, PORT1
   ```

7. ポートに信号を出力します。

   ```
   :SOURce:GPRF:GENerator:STATe ON
   ```

8. 出力レベルがレベル保証確度の範囲内にあるか問い合わせます。

   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel:SETTing?
   ```

9. AWGN を付加します。

   ```
   :SOURce:GPRF:GENerator:ARB:NOISe:STATe ON
   ```

注：

パネルのコネクタに出力される信号のレベルは，設定した値にケーブル損失を加算した値です。

ケーブル損失のコマンドの説明は「3.5.3 ケーブルの損失補正」を参照してください。

4.3.2　変調

MU887002A は，IQ データが記載された波形ファイルを使用して信号を変調します。波形ファイルは MU887002A 内部に保存されていて，使用する場合は波形メモリに読み出す必要があります。波形メモリに波形ファイルを読み込む操作を「ロードする」と呼びます。

注:

波形ファイルは，TRX 1/2 で共通のファイル保存媒体に保存されています。そのため TRX 1 でファイル保存媒体の波形ファイルを削除すると，TRX 2 の波形ファイルも削除されます。

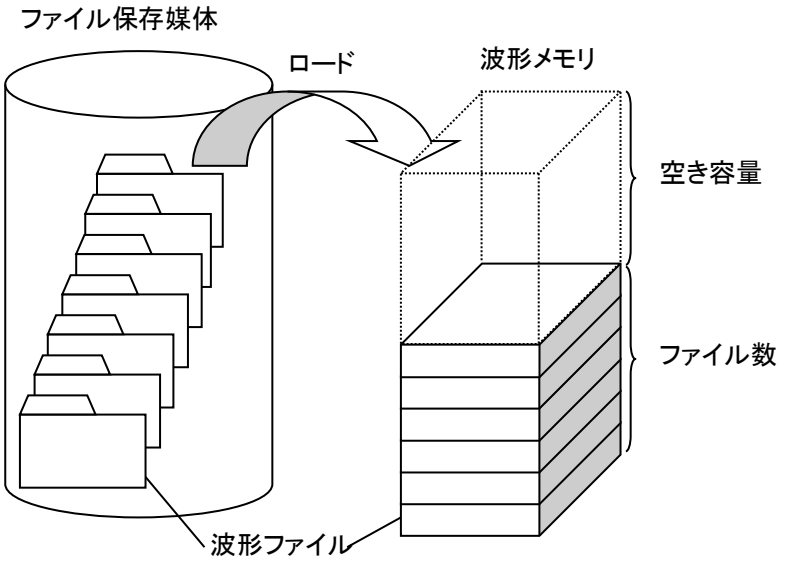


図4.3.2-1　波形ファイルのロード

波形ファイルを使用して変調するには，次の設定をします。

- 波形メモリに読み出した波形ファイルから，変調に使用する波形ファイルを指定します。
- IQ データを反転するかどうか設定します。
- 変調の設定を ARB（任意波形）にします。

注:

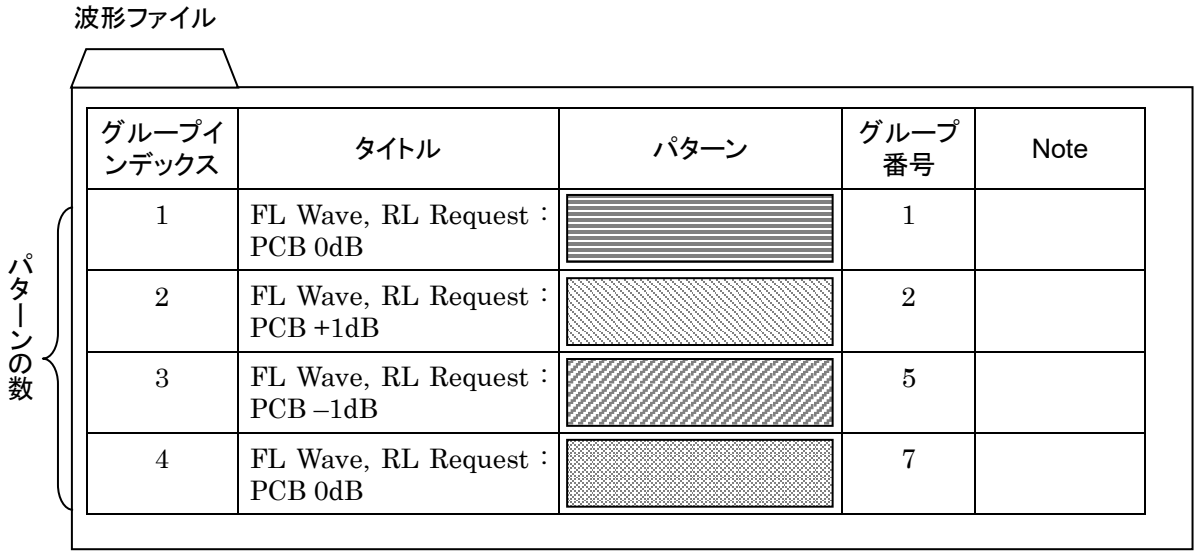
変調に使用する波形ファイルが指定されていない場合，変調の設定を ARB にしてもパネルのテストポートから信号が出力されません。

波形ファイルの構成

波形ファイルの拡張子は xml です。波形ファイルには 1 つ以上の波形パターンが含まれます。波形パターンは，パターン（IQ データ），パターンの名称（タイトル），グループ番号，Note から構成されます。ファイルに保存されている波形パターンの順番をグループインデックスと呼びます。これとは別にグループ番号が波形パターンに割り振られています。使用するパターンはグループ番号で指定します。

注:

グループ番号とセルラ用アプリケーションソフトウェアでのパターン番号は同じです。



グループインデックス	タイトル	パターン	グループ番号	Note
1	FL Wave, RL Request：PCB 0dB		1	
2	FL Wave, RL Request：PCB +1dB		2	
3	FL Wave, RL Request：PCB –1dB		5	
4	FL Wave, RL Request：PCB 0dB		7	

図4.3.2-2　波形ファイルの構成

ギャップ

波形ファイルによって変調する場合，パターンの再生間隔（ギャップ）を設定できます。

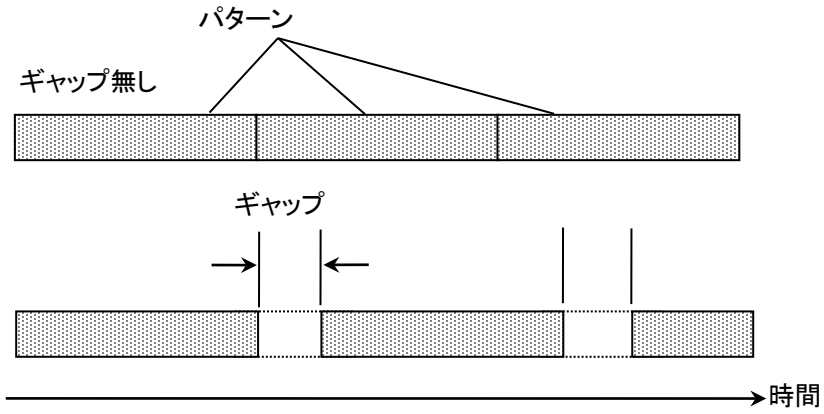


図4.3.2-3　パターンのギャップ

波形ファイル，メモリの操作には，次のコマンドを使用します。
波形ファイルの読み込み中は，パネルの状態表示ランプ 2 が緑色で点滅します。
ランプの説明は，「付録 D ランプの状態表示」を参照してください。

- 波形ファイルの読み込み
  `:SOURce:GPRF:GENerator:ARB:FILE:LOAD`
- 波形ファイルの読み込み中止
  `:SOURce:GPRF:GENerator:ARB:FILE:LOAD:CANCel`
- 波形ファイルの読み込み状態の問い合わせ
  `:SOURce:GPRF:GENerator:ARB:FILE:LOAD:STATus`
- ファイル保存媒体にある波形ファイルのグループ情報数問い合わせ
  `:SOURce:GPRF:GENerator:ARB:FILE:PATTern:COUNt`
- ファイル保存媒体にある波形ファイルのグループ番号問い合わせ
  `:SOURce:GPRF:GENerator:ARB:FILE:PATTern`

- ファイル保存媒体にある波形ファイルのグループ情報のパターン名問い合わせ
  :SOURce:GPRF:GENerator:ARB:FILE:PATTern:NAME
- 波形データの削除
  :SOURce:GPRF:GENerator:ARB:WAVeform:DELete
  :SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL
- ゼロパターン生成
  :SOURce:GPRF:GENerator:ARB:WAVeform:GENerate:ZVSPattern
- 読み込まれている波形ファイル数の問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:COUNt
- メモリの最適化
  :SOURce:GPRF:GENerator:ARB:WAVeform:DEFRag
- 再生する波形データを選択
  :SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect
- サブセクションの選択方法設定
  :SOURce:GPRF:GENerator:ARB:WAVeform:SSWitch
- ファイルバージョン問い合わせ
  :SOURce:GPRF:GENerator:ARB:FILE:VERSion
- メモリ空き容量問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:FREE
- ファイル数問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:COUNt
- ファイル名問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:NAME
- グループ番号問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:PATTern
- グループ情報数問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:COUNt
- ギャップ長の設定, 問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:GAP
- パターン名問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:NAME
- Note 文字列問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:NOTE
- サンプリングレート問い合わせ
  :SOURce:GPRF:GENerator:ARB:WAVeform:SCLock:RATE

例として，次の信号をポート 2 に出力する手順を示します。

周波数: 1800 MHz
レベル: –20 dBm
波形ファイル: MV887011A_WCDMA_0002
出力ポート: 2
スペクトル反転: 無し

1. 動作モードをノーマルモードに設定します。

```
:SOURce:GPRF:GENerator:MODE NORMAL
```

2. 変調を有り（任意波形）に設定します。

```
:SOURce:GPRF:GENerator:BBMode ARB
```

3. 周波数を 1800 MHz に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:FREQuency 1800MHZ
```

4. レベルを–20 dBm に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -20
```

5. 出力レベルがレベル保証確度の範囲内にあるか問い合わせます。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel:SETTing?
```

6. スペクトル反転を無しに設定します。

```
:SOURce:GPRF:GENerator:RFSettings:DM:POLarity NORMAL
```

7. ポート 2 を出力に設定します。

```
:ROUTe:PORT:CONNect:DIRection PORT1, PORT2
```

8. メモリのファイルをすべて削除します。

```
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL
```

9. ファイルをメモリに読み込みます。

```
:SOURce:GPRF:GENerator:ARB:FILE:LOAD
"MV887011A_WCDMA_0002"
```

10. ファイル読み込み状態を問い合わせます。

```
:SOURce:GPRF:GENerator:ARB:FILE:LOAD:STATus?
```

11. ファイルに保存されている複数のパターンからグループ番号 1 のパターンを指定します。

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect
"MV887011A_WCDMA_0002",1,1
```

12. ポートに信号を出力します。

```
:SOURce:GPRF:GENerator:STATe ON
```

13. 信号出力を停止します。

```
:SOURce:GPRF:GENerator:STATe OFF
```

4.3.3　TRX間の波形パターン同期

複数の TRX の波形パターンを同期して出力することができます。
同期方法には次の 2 種類があります。

・ 外部トリガのタイミングに同期
背面パネルのトリガコネクタ 1 にトリガを入力すると，そのタイミングに合わせてパターンを出力します。

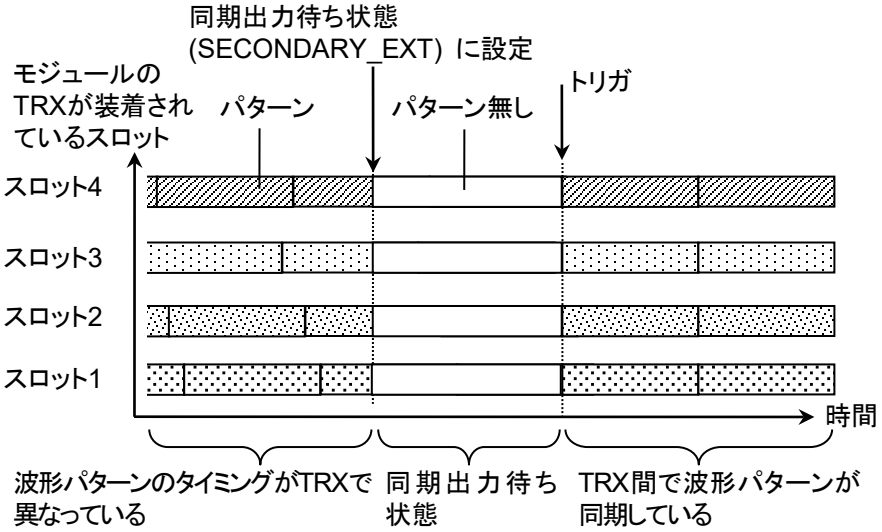


図4.3.3-1　外部トリガのタイミングに同期

・ スロット 1 の TRX の波形パターンに同期
スロット 1 の TRX にコマンドを送信すると，スロット 2～4 の TRX はそのタイミングに合わせてパターンを出力します。

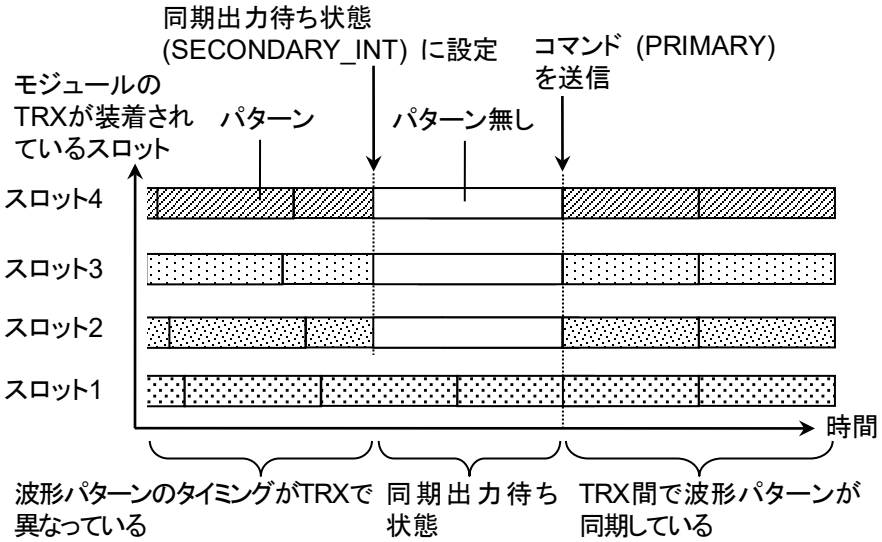


図4.3.3-2　スロット 1 の TRX の波形パターンに同期

TRX 間の波形パターン同期の設定には次のコマンドを使用します。

- 同期モードの設定
  `:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect`
- 同期出力待ち状態の確認
  `:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:STATe`
- 同期出力待ちの解除
  `:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:CANCel`

最初に TRX に対して，同期モードを設定します。

PRIMARY: ほかの TRX に対して，波形パターンのタイミングの基準となります。

SECONDARY_INT: 波形パターンのタイミングをスロット 1 の TRX に合わせます。

SECONDARY_EXT: 波形パターンのタイミングを外部トリガに合わせます。

TRX が装着されているスロットにより，設定できる同期モードに制限があります。

表4.3.3-1 設定できる同期モード

スロット / 同期モード	スロット 1	スロット 2	スロット 3	スロット 4
PRIMARY	○	×	×	×
SECONDARY_INT	×	○	○	○
SECONDARY_EXT	○	○	○	○

次の場合は，スロット 1 の TRX の波形パターンに同期して，波形パターンのタイミングを合わせることができません。

- スロット 1 に TRX が装着されていない。
- スロット 1 の同期モードが SECONDARY_EXT に設定されている。

SECONDARY_INT，または SECONDARY_EXT を設定すると，TRX は波形パターンの同期出力待ち状態になります。この状態を確認するには，次のコマンドを使用します。

`:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:STATe`

スロット 1 の TRX に PRIMARY を設定すると，SECONDARY_INT を設定した TRX はスロット 1 に同期して波形パターンを出力し，同期出力待ち状態が解除されます。

TRX に SECONDARY_EXT を設定すると，MT8870A 背面パネルのトリガコネクタ 1 に入力されたトリガのタイミングに合わせて波形パターンを出力して，同期出力待ち状態が解除されます。

波形パターンの同期出力待ち状態が解除された後にトリガコネクタ 1 にトリガを入力しても，パターン同期処理はされません。

同期出力待ち状態を途中で解除するには，次のどちらかのコマンドを使用します。
`:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:CANCel`
`*RST`

注:

`:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect` で設定した波形パターンは，
`:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect` で設定した波形パターンに変更されます。

TRX に同期モードを設定すると，MT8870A 背面パネルのトリガ設定が変わります。

表4.3.3-2　パターン同期時のトリガ設定

制御方法 / トリガ設定コマンド	PRIMARY	SECONDARY_INT	SECONDARY_EXT
:ROUTe:TRIGger:PORT	OUTPUT	INPUT	INPUT*
:ROUTe:TRIGger:OUTPut:SOURce	SGSYNC	(変更なし)	(変更なし)
:ROUTe:TRIGger:INPut:SOURce	(変更なし)	SLOT1	TRGIN1

＊:　SECONDARY_EXT を設定したスロットと，スロット 1 のトリガ設定が INPUT に変更されます。

2 つの MT8870A を次の図のように設定し，MT8870A-1 のスロット 1 の TRX に PRIMARY を設定すると，8 つの TRX の波形パターンが同期します。

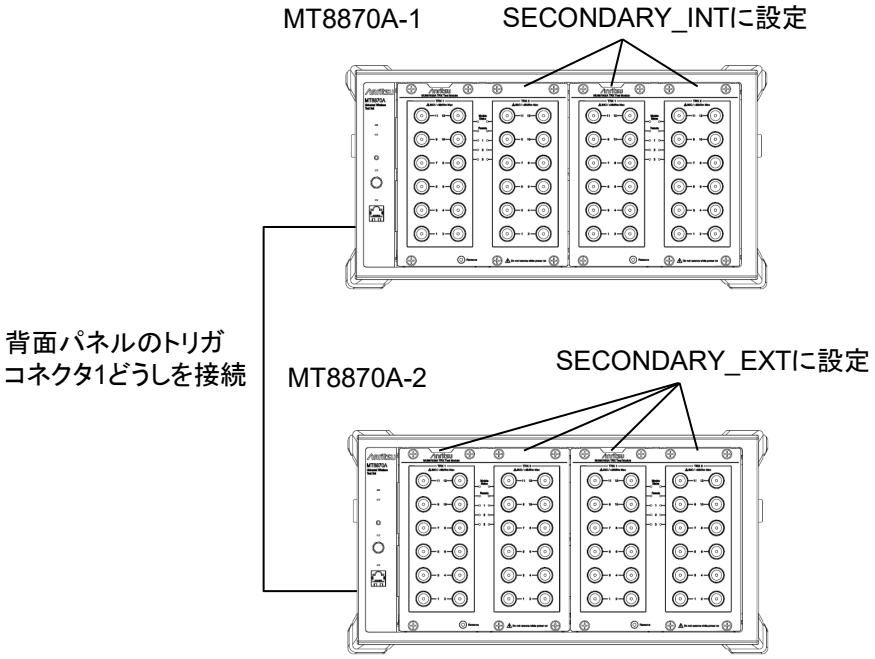


図4.3.3-3　8 つの TRX 間の波形パターン同期

図4.3.3-3 の設定で各 TRX のパターンを同期する手順を示します。

1. すべての TRX に対して，動作モードをノーマルモードに設定します。

   ```
   :SOURce:GPRF:GENerator:MODE NORMAL
   ```

2. すべての TRX に対して，周波数を 1800 MHz に設定します。

   ```
   :SOURce:GPRF:GENerator:RFSettings:FREQuency 1800MHZ
   ```

3. すべての TRX に対して，レベルを–20 dBm に設定します。

   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel -20
   ```

4. すべての TRX に対して，出力レベルがレベル保証確度の範囲内にあるか問い合わせます。

   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel:SETTing?
   ```

5. すべての TRX 対して，スペクトル反転を無しに設定します。

   ```
   :SOURce:GPRF:GENerator:RFSettings:DM:POLarity NORMAL
   ```

6. すべての TRX に対して，ポート 1 を出力に設定します。

   ```
   :ROUTe:PORT:CONNect:DIRection PORT1, PORT1
   ```

7. すべての TRX に対して，メモリのファイルをすべて削除します。

   ```
   :SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL
   ```

8. すべての TRX に対して，ファイルをメモリに読み込みます。

   ```
   :SOURce:GPRF:GENerator:ARB:FILE:LOAD
   "MV887012A_GSM_0002"
   ```

9. すべての TRX に対して，ファイル読み込み状態を問い合わせます。

   ```
   :SOURce:GPRF:GENerator:ARB:FILE:LOAD:STATus?
   ```

10. すべての TRX に対して，変調有り（ARB）を設定します。

    ```
    :SOURce:GPRF:GENerator:BBMode ARB
    ```

11. すべての TRX のポートに信号を出力します。

    ```
    :SOURce:GPRF:GENerator:STATe ON
    ```

12. MT8870A-2 のすべての TRX に次を設定します。
    同期モード:　外部トリガに同期（SECONDARY_EXT）
    波形パターン:　"MV887012A_GSM_0002"のパターン 1

    この時点で MT8870A-2 の TRX の波形パターン出力が停止します。

    ```
    :SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STAR
    t:PATTern:SELect
    SECONDARY_EXT,"MV887012A_GSM_0002",1,1
    ```

13. MT8870A-1 のスロット 2 からスロット 4 の TRX に次を設定します。
    同期モード:　スロット 1 のパターンに同期（SECONDARY_INT）
    波形パターン:　"MV887012A_GSM_0002"のパターン 1

この時点でスロット 2～4 の TRX の波形パターン出力が停止します。

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STAR
t:PATTern:SELect
SECONDARY_INT,"MV887012A_GSM_0002",1,1
```

14. MT8870A-2 のすべての TRX, MT8870A-1 のスロット 2 からスロット 4 の TRX に対して, 波形パターンの同期出力待ち状態になっているか問い合わせます。

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STAR
t:STATe?
```

15. MT8870A-1 のスロット 1 の TRX に次を設定します。
同期モード:　パターンのプライマリ（PRIMARY）
波形パターン:　"MV887012A_GSM_0002"のパターン 1

このコマンドを送信すると, すべての TRX 間の波形パターンが同期して出力されます。

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STAR
t:PATTern:SELect PRIMARY,"MV887012A_GSM_0002",1,1
```

4.4 シーケンスモード

4.4.1 動作内容

シーケンスモードは，シーケンステーブルに記載された順序に従って，周波数，レベル，変調波形を変えながら RF 信号を発生する機能です。

RF 信号を発生する時間は，ステップ数で設定します。1 ステップあたりの時間を変えることで，W-CDMA，GSM，EVDO などの規格に対応できます。

RF 信号を切り替えるタイミングは，時間，またはトリガから選択できます。
背面パネルのコネクタからの入力信号，シーケンステーブル内に記述したトリガ，出力信号のレベル，波形のグループ番号からトリガを選択できます。

周波数，レベルのシーケンスと変調波形のシーケンスが別々に設定できるため，変調波形を周波数，レベルと非同期に制御できます。

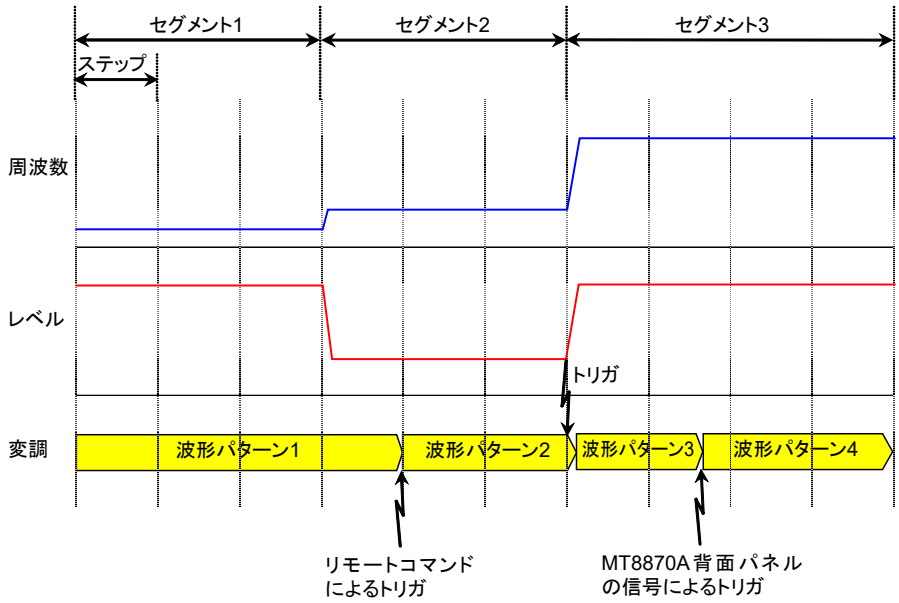


図4.4.1-1　シーケンスモードの実行例

⚠ 注意

RF 信号を切り替えてから信号が安定するまで，500 μs 程度時間がかかります。この時間が経過した後，被測定物の試験を開始してください。

4.4.2　シーケンステーブルの構成

シーケンスモードでは，最大 16 個のシーケンステーブルの記載内容に従って RF 信号を発生します。
シーケンステーブルは，共通パラメータ，波形リストテーブル，SG（Signal Generator）シーケンサテーブルで構成されます。

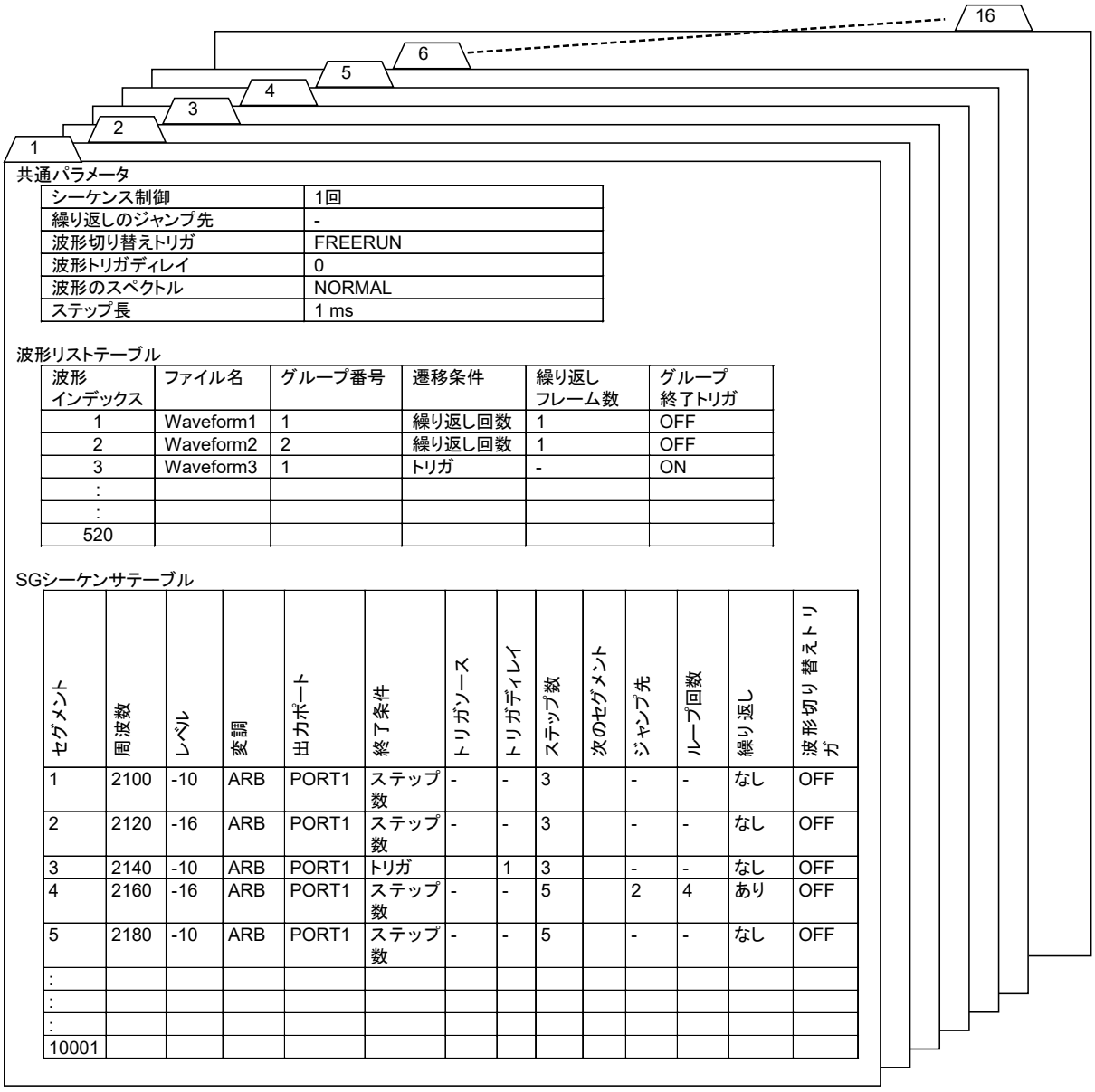


図4.4.2-1　シーケンステーブルの構成

共通パラメータ

共通パラメータには，SG シーケンサの制御方法を設定します。
共通パラメータの設定項目は次のとおりです。

- シーケンス制御
  シーケンスの処理方法
    Single: 1 回実行

Continuous:繰り返し実行

- 繰り返しのジャンプ先
  シーケンス制御が繰り返し実行（Continuous）の場合，2 回目以降の開始セグメント番号
- 波形切り替えトリガ
  波形リストテーブルで使用するトリガの選択
  外部トリガ：背面パネルのトリガコネクタ信号
  レベル：　信号のレベル
  SG：　SG シーケンサ処理で発生するトリガ
  マニュアル：コマンド送信によるトリガ
- 波形トリガディレイ
  トリガ発生から変調波形が変更されるまでの時間差（ms）
- 波形のスペクトル
  ベースバンド信号の IQ 反転のオン/オフ
- ステップ長
  SG シーケンサで設定するステップの長さ

指定した時間：	0.5~80 ms
GSM の Slot 時間：	15/26 ms
GSM の Frame 時間：	60/13 ms
WCDMA の Slot 時間：	10/15 ms
EVDO の Slot 時間：	80/48 ms
EVDO の Frame 時間：	80/3 ms

- 登録波形数
  波形リストテーブルに登録されている波形数（読み取りのみ）

共通パラメータの設定，問い合わせに使用するコマンドは次のとおりです。

- シーケンス制御
  `:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:REPetition`
- 繰り返しのジャンプ先
  `:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:GOTO`
- 波形切り替えトリガ
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce`
- 波形トリガディレイ
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:DELay`
- 波形のスペクトル
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:DM:POLarity`
- ステップ長
  `:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STEP:LENGth`
- 登録波形数（読み取りのみ）
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:PCOunt`

波形リストテーブル

波形リストテーブルの設定項目は次のとおりです。

- 波形インデックス
  波形パターンの格納場所を示す番号
- ファイル名
  波形のファイル名
- グループ番号
  波形ファイルに含まれる波形パターンのグループ番号
- 遷移条件
  次の波形パターンに切り替える条件の選択

無し:	切り替わり無し
繰り返し:	繰り返しフレーム数を実行
トリガ:	トリガが発生

- 繰り返しフレーム数
  波形パターンの繰り返し回数
- グループ終了トリガ
  同じグループ番号の波形ファイルが終了したときに発生するトリガ（Group End Trigger）のオン/オフ

波形リストテーブルの設定，問い合わせに使用するコマンドは次のとおりです。

- 波形パターンの選択
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:SELect`
- 波形パターンの削除
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:DELete`
- 遷移条件
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:ENDCondition`
- 波形パターンの繰り返し回数
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:IREPetition`
- グループ終了トリガ
  `:SOURce:GPRF:GENerator:SEQuence:WAVeform:GETRigger`

SG シーケンサテーブル

SG シーケンサテーブルの設定項目は次のとおりです。

- セグメント
  設定項目の番号
- 周波数
  出力周波数
- レベル
  出力レベル（dBm）
- 変調
  変調のオン/オフ
- 出力ポート
  RF 信号を出力するパネルのコネクタ番号（ポート 1～12）
- 終了条件
  実行中のセグメントを終了する条件
  ステップ： ステップ数に相当する時間が経過
  トリガ： トリガが発生
- トリガソース
  SG シーケンサで使用するトリガの選択
  外部トリガ：背面パネルのトリガコネクタ信号
  レベル： 信号のレベル
  波形グループ： 波形パターン切り替え時のトリガ
  波形マーカ： 波形マーカ 1~3
- トリガディレイ
  トリガ発生からセグメント処理が終了するまでの時間差（ms）
- ステップ数
  セグメントの継続時間
- 次のセグメント
  次に実行するセグメントの制御
  次のセグメント番号： 現在のセグメント番号より 1 大きいセグメント番号
  ループ： ジャンプ先で設定したセグメント番号
- ジャンプ先
  「次のセグメント」でループを設定したときに実行するセグメント番号
- ループ回数
  「次のセグメント」でループを設定したときにジャンプする回数
- 波形切り替えトリガ
  波形リストテーブルの実行を制御するトリガ発生のオン/オフ

SG シーケンサテーブルの内容と，設定コマンドを次の表に示します。

- 周波数
  :SOURce:GPRF:GENerator:SEQuence:RX:FREQuency
  :SOURce:GPRF:GENerator:SEQuence:RX:FREQuency:ALL
- レベル
  :SOURce:GPRF:GENerator:SEQuence:RX:LEVel
  :SOURce:GPRF:GENerator:SEQuence:RX:LEVel:ALL
- 変調
  :SOURce:GPRF:GENerator:SEQuence:RX:BBMode
  :SOURce:GPRF:GENerator:SEQuence:RX:BBMode:ALL
- 出力ポート
  :SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe
  :SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe:ALL
- 終了条件
  :SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition
  :SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition:ALL
- トリガソース
  :SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce
  :SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce:ALL
- トリガディレイ
  :SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay
  :SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay:ALL
- ステップ数
  :SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt
  :SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt:ALL
- 次のセグメント
  :SOURce:GPRF:GENerator:SEQuence:RX:NSLControl
  :SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:ALL
- ジャンプ先
  :SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment
  :SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment:ALL
- ループ回数
  :SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt
  :SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt:ALL
- 波形パターン切り替えトリガ
  :SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger
  :SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger:ALL

4.4.3 ファイル操作

SG シーケンサテーブルのデータ（SG シーケンサデータ）は次のコマンドでファイルに保存，または読み込みできます。

- SG シーケンサデータの保存
  :SOURce:GPRF:GENerator:SEQuence:STORe
- SG シーケンサデータの読み込み
  :SOURce:GPRF:GENerator:SEQuence:LOAD

次のコマンドで波形ファイルの情報を取得できます。

波形ファイルは次のコマンドを使用して，記憶媒体からメモリへロードします。

- 波形ファイルのロード
  :SOURce:GPRF:GENerator:ARB:FILE:LOAD
- 波形ファイルのロード中止
  :SOURce:GPRF:GENerator:ARB:FILE:LOAD:CANCel
- 波形ファイルのロード状態問い合わせ
  :SOURce:GPRF:GENerator:ARB:FILE:LOAD:STATus

波形ファイルと波形リストテーブルの関連付けには，次のコマンドを使用します。

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:SELect
```

メモリの空き容量が少ない場合は，次のコマンドで波形ファイルをメモリから削除します。

```
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL
```

4.4.4　設定項目のエラーチェック

シーケンステーブルの次の項目は，設定値がエラーになっていないか確認できます。

- 入力レベル
- 出力レベル
- ステップ数
- キャプチャメモリ量

次のコマンドで，シーケンステーブルにエラーがあるか問い合わせます。

```
:SOURce:GPRF:GENerator:SEQuence:RX:ERRor
```

レスポンスの値とエラー内容を次の表に示します。複数のエラーが発生している場合，そのエラーの値の合計値がレスポンスの値になります。

表4.4.4-1　シーケンス測定の測定条件

値	エラー内容
1	Loop のネストを発生させています。
2	Loop のジャンプ先が実行シーケンスの範囲外です。
4	トリガ待ちのセグメントで Loop を設定しています。
8	レベルが設定範囲外の値です。
16	使用できない波形パターンを設定しています。

シーケンステーブルにエラーがあると，シーケンスを実行できません。
ただし，エラーの値が 8 の場合は，
`:SOURce:GPRF:GENerator:SEQuence:EXECute`
の強制実行モードを指定して，シーケンスを実行することができます。
この場合，レベル設定値が上限値を超えるときは出力できる上限のレベルで，レベル設定値が下限値未満のときは出力できる下限のレベルで信号を出力します。

4.4.5 シーケンスの実行

シーケンスの実行制御に使用するコマンドは次のとおりです。

- 実行するシーケンスの数と順番設定
  :SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern
- シーケンスの開始
  :SOURce:GPRF:GENerator:SEQuence:EXECute
  :SOURce:GPRF:GENerator:SEQuence:RLISt
- シーケンスの状態問い合わせ
  :SOURce:GPRF:GENerator:SEQuence:STATus
- マニュアルトリガの発生
  :SOURce:GPRF:GENerator:SEQuence:WAVeform:MTEXecute
- シーケンスの停止
  :SOURce:GPRF:GENerator:SEQuence:CANCel

シーケンステーブルを編集したら，次の手順でシーケンスを実行します。

1. 動作モードをシーケンスモードに設定します。

   :SOURce:GPRF:GENerator:MODE SEQUENCE

2. 実行するシーケンステーブルと順序を次のコマンドで設定します。

   :SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern

3. シーケンスが終了した後にノーマルモードに変更する場合は，次のコマンドで設定します。

   :SOURce:GPRF:GENerator:SEQuence:REINitialization ON

4. 次のコマンドを送信して，シーケンスを開始します。

   :SOURce:GPRF:GENerator:SEQuence:EXECute

5. シーケンスが実行中かどうかは次のコマンドで確認できます。

   :SOURce:GPRF:GENerator:SEQuence:STATus

6. 波形切り替えトリガをManualに設定している場合は，次のコマンドを送信するとトリガが発生します。

   :SOURce:GPRF:GENerator:SEQuence:WAVeform:MTEXecute

7. シーケンスを中止する場合は，次のコマンドを送信します。

   :SOURce:GPRF:GENerator:SEQuence:CANCel

   :SOURce:GPRF:GENerator:MODE NORMAL
   で，ノーマルモードに設定してもシーケンスを中止できます。

4.4.6　TRX間の波形パターン同期

シーケンス実行時に，複数の TRX の波形パターンを同期して出力することができます。

波形パターンの同期方法は，ノーマルモードと同様に 2 種類あります。

- 外部トリガのタイミングに同期
- スロット 1 の TRX の波形パターンに同期

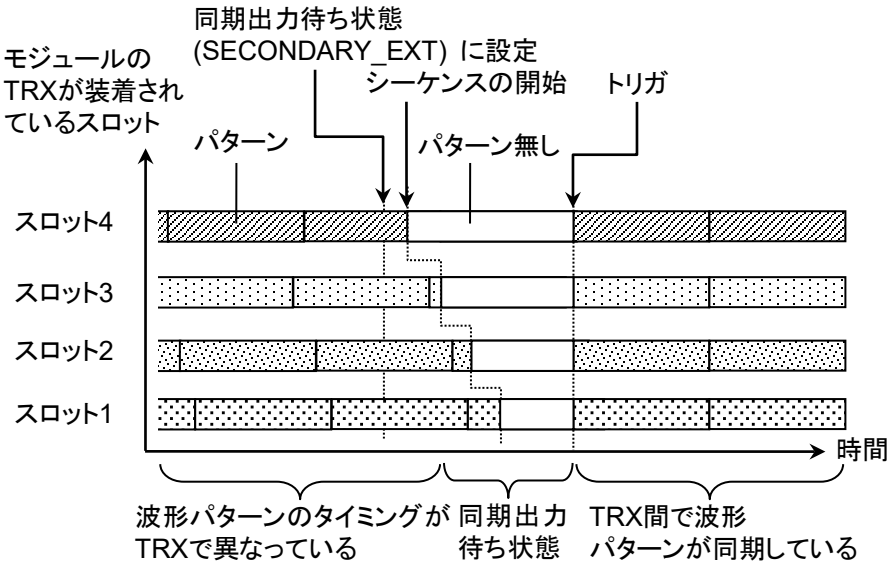


図4.4.6-1　外部トリガのタイミングに同期

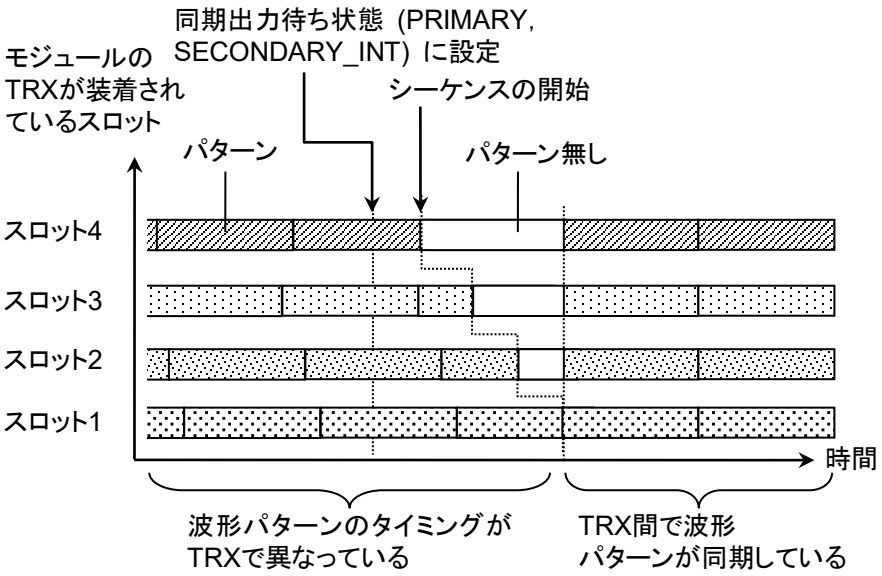


図4.4.6-2　スロット 1 の TRX の波形パターンに同期

TRX 間の波形パターン同期の設定には次のコマンドを使用します。

- 同期モードの設定
  :SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:SYNChronize
- 同期出力待ち状態の確認
  :SOURce:GPRF:GENerator:SEQuence:WAVeform:SYNChronize:STARt:STATe
- 同期出力待ちの解除
  :SOURce:GPRF:GENerator:SEQuence:CANCel

最初に TRX に対して，同期モードを設定します。

NONE: ほかの TRX と波形パターンを同期しません。

PRIMARY: ほかの TRX に対して，波形パターンのタイミングの基準となります。

SECONDARY_INT: 波形パターンのタイミングをスロット 1 の TRX に合わせます。

SECONDARY_EXT: 波形パターンのタイミングを外部トリガに合わせます。

TRX が装着されているスロットにより，設定できる同期モードに制限があります。

表4.4.6-1 設定できる同期モード

スロット / 同期モード	スロット 1	スロット 2	スロット 3	スロット 4
NONE	○	○	○	○
PRIMARY	○	×	×	×
SECONDARY_INT	×	○	○	○
SECONDARY_EXT	○	○	○	○

次の場合は，スロット 1 の TRX を基準として，波形パターンのタイミングを合わせることができません。

- スロット 1 に TRX が装着されていない。
- スロット 1 の同期モードが NONE，または SECONDARY_EXT に設定されている。

SECONDARY_INT，または SECONDARY_EXT を設定すると，TRX は波形パターンの同期出力待ち状態になります。この状態を確認するには，次のコマンドを使用します。

:SOURce:GPRF:GENerator:SEQuence:WAVeform:SYNChronize:STARt:STATe

同期を開始するには，次のどちらかのコマンドを送信します。

:SOURce:GPRF:GENerator:SEQuence:EXECute
:SOURce:GPRF:GENerator:SEQuence:RLISt

SECONDARY_INTを設定したTRXは，スロット1のTRXと同期して波形パターンを出力します。
SECONDARY_EXTを設定したTRXは，トリガコネクタ1に入力されたトリガのタイミングに合わせて波形パターンを出力します。波形パターンのタイミングを合わせるのは，1 回目に入力されたトリガだけです。2 回目以降のトリガに対して波形パターンは同期しません。

同期出力待ち状態を途中で解除するには，次のどちらかのコマンドを使用します。
`:SOURce:GPRF:GENerator:SEQuence:CANCel`
`*RST`

注:
`*RST`を使用すると，シーケンスモードからノーマルモードに変わります。

TRX に同期モードを設定すると，MT8870A 背面パネルのトリガ設定が変わります。

表4.4.6-2　パターン同期時のトリガ設定

制御方法 / トリガ設定コマンド	NONE	PRIMARY	SECONDARY_INT	SECONDARY_EXT
:ROUTe:TRIGger:PORT	(変更なし)	OUTPUT	INPUT	INPUT*
:ROUTe:TRIGger:OUTPut:SOURce	(変更なし)	SGSYNC	(変更なし)	(変更なし)
:ROUTe:TRIGger:INPut:SOURce	(変更なし)	(変更なし)	SLOT1	TRGIN1

＊:　SECONDARY_EXTを設定したスロットと，スロット1のトリガ設定がINPUTに変更されます。

MT8870A のスロット 1 の TRX に合わせて，すべての TRX の波形パターンを同期する手順を示します。あらかじめ，各 TRX のシーケンステーブルを編集しておきます。

1. すべての TRX に対して，動作モードをシーケンスモードに設定します。

   ```
   :SOURce:GPRF:GENerator:MODE SEQUENCE
   ```

2. すべての TRX に対して，実行するシーケンステーブルと順序を次のコマンドで設定します。

   ```
   :SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern
   ```

3. スロット 2 からスロット 4 の TRX に次を設定します。
   同期モード：スロット 1 のパターンに同期（SECONDARY_INT）

   ```
   :SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:
   SYNChronize SECONDARY_INT
   ```

4. スロット 2 からスロット 4 の TRX のシーケンスを実行します。この時点では，TRX に波形パターンが出力されません。

   ```
   :SOURce:GPRF:GENerator:SEQuence:EXECute
   ```

5. スロット 2 からスロット 4 の TRX に対して，波形パターンの同期出力待ち状態になっているか問い合わせます。

   ```
   :SOURce:GPRF:GENerator:SEQuence:WAVeform:SYNChronize
   :STARt:STATe?
   ```

6. スロット 1 の TRX に次を設定します。
   同期モード：パターンのプライマリ（PRIMARY）

   この時点では，TRX 間で波形パターンは同期していません。

   ```
   :SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:
   SYNChronize PRIMARY
   ```

7. スロット 1 の TRX のシーケンスを実行します。スロット 1 の TRX がシーケンスを開始したときに，波形パターンが TRX 間で同期します。

   ```
   :SOURce:GPRF:GENerator:SEQuence:EXECute
   ```

第5章　SCPI コマンドリファレンス

この章では，SCPI コマンドの詳細について説明します。

5.1 コマンド一覧 5-2
5.1.1 基本操作コマンド 5-2
5.1.2 VSGコマンド 5-15
5.2 コマンドの詳細 5-27
5.2.1 基本操作コマンド 5-28
5.2.2 VSGコマンド 5-151

5.1 コマンド一覧

5.1.1 基本操作コマンド

Calibration

機能	コマンド	クエリ	レスポンス
Band Calibration Error Count	-----	:CALCulate:CALibration:BAND:ERRor:COUNt?	<count>
BAND Calibration Error Last Time	-----	:CALCulate:CALibration:BAND:ERRor:LASTtime?	<year>,<month>,<day>,<hour>,<min>
BAND Calibration Last Time	-----	:CALCulate:CALibration:BAND:LASTtime?	<year>,<month>,<day>,<hour>,<min>
Result of Band Calibration	-----	:CALCulate:CALibration:BAND:RESult?	<Result>
Execute BAND Calibration	:CALCulate:CALibration:BAND:STARt	-----	-----
BAND Calibration Temperature	:CALCulate:CALibration:BAND:STARt:TEMPerature <temperature>	-----	-----
Elapsed Time after Full Calibration	-----	:CALCulate:CALibration:ELAPsed:TIME?	<time>
Full Calibration Error Count	-----	:CALCulate:CALibration:FULL:ERRor:COUNt?	<count>
FULL Calibration Error Last Time	-----	:CALCulate:CALibration:FULL:ERRor:LASTtime?	<year>,<month>,<day>,<hour>,<min>
FULL Calibration Last Time	-----	:CALCulate:CALibration:FULL:LASTtime?	<year>,<month>,<day>,<hour>,<min>
Result of Full Calibration	-----	:CALCulate:CALibration:FULL:RESult?	<Result>

Calibration (Cont'd)

機能	コマンド	クエリ	レスポンス
Execute FULL Calibration	:CALCulate:CALibration:FULL:STARt	-----	-----
Initial Calibration	:CALCulate:CALibration:INITial	-----	-----
Temperature at Full Calibration Execution	-----	:CALCulate:CALibration:TEMPerature?	<temp>

Communication

機能	コマンド	クエリ	レスポンス
GPIB Address	:SYSTem:COMMunicate:GPIB:ADDR <gp_address>	:SYSTem:COMMunicate:GPIB:ADDR?	<gp_address>
Delimiter	:SYSTem:COMMunicate:GPIB[:SELF]:DELimiter <code>	:SYSTem:COMMunicate:GPIB[:SELF]:DELimiter?	<code>
Terminator of Message	:SYSTem:COMMunicate:GPIB[:SELF]:TERMinator <code>	:SYSTem:COMMunicate:GPIB[:SELF]:TERMinator?	<code>
Domain Name	:SYSTem:COMMunicate:NET:DNS:DOMain <name>	:SYSTem:COMMunicate:NET:DNS:DOMain?	<name>
Host Name	:SYSTem:COMMunicate:NET:DNS:HOST <host>	:SYSTem:COMMunicate:NET:DNS:HOST?	<host>
MAC Address	-----	:SYSTem:COMMunicate:NET:HWADdress?	<macaddr>
IP Version	:SYSTem:COMMunicate:NET:IPVersion <ipver>	:SYSTem:COMMunicate:NET:IPVersion?	<ipver>
IPv4 Type	:SYSTem:COMMunicate:NET:IPVFour:ADDRess:TYPE <ipv4type>	:SYSTem:COMMunicate:NET:IPVFour:ADDRess:TYPE?	<ipv4type>
Get Current IPv4 Default Gateway	-----	:SYSTem:COMMunicate:NET:IPVFour:CURRent:GIP?	<gateway>

Communication (Cont'd)

機能	コマンド	クエリ	レスポンス
Get Current IPv4 Address	-----	:SYSTem:COMMunicate:NET:IPVFour:CURRent:IPADdress?	<num>,<ipaddr>
Get Current IPv4 Subnet Mask	-----	:SYSTem:COMMunicate:NET:IPVFour:CURRent:SMASk?	<mask>
IPv4 DNS Server Auto	:SYSTem:COMMunicate:NET:IPVFour:DNS:AUTO:ENABle <switch>	:SYSTem:COMMunicate:NET:IPVFour:DNS:AUTO:ENABle?	<switch>
IPv4 DNS Primary Address	:SYSTem:COMMunicate:NET:IPVFour:DNS:PRIMary <ip4addr>	:SYSTem:COMMunicate:NET:IPVFour:DNS:PRIMary?	<ip4addr>
IPv4 DNS Secondary Address	:SYSTem:COMMunicate:NET:IPVFour:DNS:SECondary <ip4addr>	:SYSTem:COMMunicate:NET:IPVFour:DNS:SECondary?	<ip4addr>
IPv4 Default Gateway	:SYSTem:COMMunicate:NET:IPVFour:STATic:GIP <gateway>	:SYSTem:COMMunicate:NET:IPVFour:STATic:GIP?	<gateway>
IPv4 Address	:SYSTem:COMMunicate:NET:IPVFour:STATic:IPADdress <ipv4_addr1>,<ipv4_addr2>,<ipv4_addr3>,<ipv4_addr4>	:SYSTem:COMMunicate:NET:IPVFour:STATic:IPADdress?	<ipv4_addr1>,<ipv4_addr2>,<ipv4_addr3>,<ipv4_addr4>
IPv4 Subnet Mask	:SYSTem:COMMunicate:NET:IPVFour:STATic:SMASk <mask>	:SYSTem:COMMunicate:NET:IPVFour:STATic:SMASk?	<mask>
IPv6 Type	:SYSTem:COMMunicate:NET:IPVSix:ADDRess:TYPE <ipv6type>	:SYSTem:COMMunicate:NET:IPVSix:ADDRess:TYPE?	<ipv6type>
Get IPv6 Default Router Address	-----	:SYSTem:COMMunicate:NET:IPVSix:CURRent:DEFRouter?	<ip6addr>
Get Current IPv6 Address and Prefix	-----	:SYSTem:COMMunicate:NET:IPVSix:CURRent:IPADdress?	<num>,<ipaddr>
Get IPv6 Link Local Address	-----	:SYSTem:COMMunicate:NET:IPVSix:CURRent:LOCaladdress?	<ip6addr>
IPv6 DNS Server Auto	:SYSTem:COMMunicate:NET:IPVSix:DNS:AUTO:ENABle <switch>	:SYSTem:COMMunicate:NET:IPVSix:DNS:AUTO:ENABle?	<switch>

Communication (Cont'd)

機能	コマンド	クエリ	レスポンス
IPv6 DNS Primary Address	:SYSTem:COMMunicate:NET:IPVSix:DNS:PRIMary <ip6addr>	:SYSTem:COMMunicate:NET:IPVSix:DNS:PRIMary?	<ip6addr>
IPv6 DNS Secondary Address	:SYSTem:COMMunicate:NET:IPVSix:DNS:SECondary <ip6addr>	:SYSTem:COMMunicate:NET:IPVSix:DNS:SECondary?	<ip6addr>
IPv6 Default Router	:SYSTem:COMMunicate:NET:IPVSix:STATic:DEFRouter<ip6addr>	:SYSTem:COMMunicate:NET:IPVSix:STATic:DEFRouter?	<ip6addr>
IPv6 Address	:SYSTem:COMMunicate:NET:IPVSix:STATic:IPADdress <ipv6_addr1>,<ipv6_addr2>,<ipv6_addr3>,<ipv6_addr4>	:SYSTem:COMMunicate:NET:IPVSix:STATic:IPADdress?	<ipv6_addr1>,<ipv6_addr2>,<ipv6_addr3>,<ipv6_addr4>
Network Restart	:SYSTem:COMMunicate:NET:RESTart	-----	-----
Raw Socket Port Number	:SYSTem:COMMunicate:NET:PORT:SETTing <port>	:SYSTem:COMMunicate:NET:PORT:SETTing?	<port>

FTP

機能	コマンド	クエリ	レスポンス
FTP User List	-----	:SYSTem:COMMunicate:NET:FTP:USER:CATalog?	<name>
Change FTP User Name	:SYSTem:COMMunicate:NET:FTP:USER:CHANge:NAME <before>,<after>	-----	-----
Change FTP User Password	:SYSTem:COMMunicate:NET:FTP:USER:CHANge:PASS <name>,<before>,<after>	-----	-----

Information

機能	コマンド	クエリ	レスポンス
Log Clear	:MMEMory:LOG:CLEar	-----	-----
Log Reading	-----	:MMEMory:LOG:LOAD? [<size>]	<log>
Log Size Reading	-----	:MMEMory:LOG:SIZE?	<log>
Module Model Number.	-----	:MAINtenance:SYSTem:INFormation:DEVice:MODel?	<model>
Main Frame Model Number.	-----	:MAINtenance:SYSTem:INFormation:MAINframe:MODel?	<model>
License Information	-----	:SYSTem:BASE:LICense:INFormation? <model>	<kind>,<days>
Module Information	-----	:SYSTem:INFormation? <index>	-----
Module Serial No.	-----	:SYSTem:INFormation:DEVice:ID?	<serial>
Module Power ON Count	-----	:SYSTem:INFormation:POWeron:COUNt?	<num>
Main Frame Information	-----	:SYSTem:INFormation:MAINframe? <index>	-----
Main Frame Serial No.	-----	:SYSTem:INFormation:MAINframe:DEVice:ID?	<serial>
Chassis Power ON Count	-----	:SYSTem:INFormation:MAINframe:POWeron:COUNt?	<num>
Main Frame Running Time Count Read	-----	:SYSTem:INFormation:MAINframe:RTIMe?	<time>
Firmware Version	-----	:SYSTem:INFormation:MAINframe:PACKage:VERSion?	<ver>
Type Name of Software	-----	:SYSTem:INFormation:MAINframe:SOFTware?	<n>,<model>
Option Number	-----	:SYSTem:INFormation:MAINframe:SOFTware:OPTion? <model>	<n>,<num>

Information (Cont'd)

機能	コマンド	クエリ	レスポンス
Type Name of Waveform	-----	:SYSTem:INFormation:MAINframe:WAVeform?	<n>,<model>
Main Frame Temperature	-----	:SYSTem:INFormation:MAINframe:TEMPerature?	<temp>
TRX 1/2	-----	:SYSTem:INFormation:MAINframe:TRX?	<trx>
Count of Module Insertion	-----	:SYSTem:INFormation:PLUG:COUNt?	<num>
Module Running Time Count Read	-----	:SYSTem:INFormation:RTIMe?	<time>
Module Slot Number	-----	:SYSTem:INFormation:SLOT?	<slot>
Module Temperature	-----	:SYSTem:INFormation:TEMPerature?	<temp>
Package Version	-----	:SYSTem:VERSion?	<ver>
Drive Size Reading	-----	:MMEMory:SIZE? [<target>]	<total>,<blank>

Operation Status Register

機能	コマンド	クエリ	レスポンス
Operation Status Condition Register Query	-----	:STATus:OPERation:CONDition?	<oscr>
Operation Status Enable Register	:STATus:OPERation:ENABle <oser>	:STATus:OPERation:ENABle?	<oser>
Generator Operation Status Register Query	-----	:STATus:OPERation:GENerator[:EVENt]?	<gosr>
Generator Operation Status Condition Register Query	-----	:STATus:OPERation:GENerator:CONDition?	<goscr>
Generator Operation Status Enable Register	:STATus:OPERation:GENerator:ENABle <goser>	:STATus:OPERation:GENerator:ENABle?	<goser>
Generator Operation Status Negative Transition Filter	:STATus:OPERation:GENerator:NTRansition <gosr_ntf>	:STATus:OPERation:GENerator:NTRansition?	<gosr_ntf>
Generator Operation Status Positive Transition Filter	:STATus:OPERation:GENerator:PTRansition <gosr_ptf>	:STATus:OPERation:GENerator:PTRansition?	<gosr_ptf>
Measurement Operation Status Register Query	-----	:STATus:OPERation:MEASure[:EVENt]?	<mosr>
Measurement Operation Status Condition Register Query	-----	:STATus:OPERation:MEASure:CONDition?	<moscr>
Measurement Operation Status Enable Register	:STATus:OPERation:MEASure:ENABle <moser>	:STATus:OPERation:MEASure:ENABle?	<moser>
Measurement Operation Status Negative Transition Filter	:STATus:OPERation:MEASure:NTRansition <mosr_ntr>	:STATus:OPERation:MEASure:NTRansition?	<mosr_ntr>
Measurement Operation Status Positive Transition Filter	:STATus:OPERation:MEASure:PTRansition <mosr_ptr>	:STATus:OPERation:MEASure:PTRansition?	<mosr_ptf>
Operation Status Negative Transition Filter	:STATus:OPERation:NTRansition <osr_ntf>	:STATus:OPERation:NTRansition?	<osr_ntf>

Operation Status Register (Cont'd)

機能	コマンド	クエリ	レスポンス
Operation Status Positive Transition Filter	:STATus:OPERation:PTRansition <osr_ptf>	:STATus:OPERation:PTRansition?	<osr_ptf>
Operation Status Register Query	-----	:STATus:OPERation[:EVENt]?	<osr>
Preset Status	:STATus:PRESet	-----	-----

Questionable Register

機能	コマンド	クエリ	レスポンス
Questionable Status Condition Register Query	-----	:STATus:QUEStionable:CONDition?	<qscr>
Questionable Status Enable Register	:STATus:QUEStionable:ENABle <qser>	:STATus:QUEStionable:ENABle?	<qser>
Generator Questionable Status Condition Register Query	-----	:STATus:QUEStionable:GENerator:CONDition?	<gqscr>
Generator Questionable Status Enable Register	:STATus:QUEStionable:GENerator:ENABle <gqser>	:STATus:QUEStionable:GENerator:ENABle?	<gqser>
Generator Questionable Status Negative Transition Filter	:STATus:QUEStionable:GENerator:NTRansition <gqsr_ntf>	:STATus:QUEStionable:GENerator:NTRansition?	<gqsr_ntf>
Generator Questionable Status Positive Transition Filter	:STATus:QUEStionable:GENerator:PTRansition <gqsr_ptf>	:STATus:QUEStionable:GENerator:PTRansition?	<gqsr_ptf>
Generator Questionable Status Register Query	-----	:STATus:QUEStionable:GENerator[:EVENt]?	<gqsr>
Measurement Questionable Status Condition Register Query	-----	:STATus:QUEStionable:MEASure:CONDition?	<mqscr>
Measurement Questionable Status Enable Register	:STATus:QUEStionable:MEASure:ENABle <mqser>	:STATus:QUEStionable:MEASure:ENABle?	<mqser>
Measurement Questionable Status Negative Transition Filter	:STATus:QUEStionable:MEASure:NTRansition <mqsr_ntr>	:STATus:QUEStionable:MEASure:NTRansition?	<mqsr_ntf>

Questionable Register (Cont'd)

機能	コマンド	クエリ	レスポンス
Measurement Questionable Status Positive Transition Filter	:STATus:QUEStionable:MEASure :PTRansition <mqsr_ptr>	:STATus:QUEStionable:MEASure :PTRansition?	<mqsr_ptf>
Measurement Questionable Status Register Query	-----	:STATus:QUEStionable:MEASure [:EVENt]?	<mqsr>
Questionable Status Negative Transition Filter	:STATus:QUEStionable:NTRansition <qsr_ntf>	:STATus:QUEStionable:NTRansition?	<qsr_ntf>
Questionable Status Positive Transition Filter	:STATus:QUEStionable:PTRansition <qsr_ptf>	:STATus:QUEStionable:PTRansition?	<qsr_ptf>
Questionable Status Register Query	-----	:STATus:QUEStionable[:EVENt]?	<qsr>

Reference Clock

機能	コマンド	クエリ	レスポンス
Adjust Reference Clock Default	-----	:SYSTem:BASE:REFerence:FREQuency:ADJust:DEFault?	<val>
Frequency Reference	:SYSTem:BASE:REFerence:FREQuency <source>	:SYSTem:BASE:REFerence:FREQuency?	<source>
Frequency Reference	:SYSTem:BASE:REFerence:FREQuency:CONFigure <source>	:SYSTem:BASE:REFerence:FREQuency:CONFigure?	<source>
Adjust Reference Clock	:SYSTem:BASE:REFerence:FREQuency:ADJust <val>	:SYSTem:BASE:REFerence:FREQuency:ADJust?	<val>
Reference Signal Query	-----	:SYSTem:BASE:REFerence:FREQuency:SOURce?	<source>
Reference Signal Query	-----	:SYSTem:BASE:REFerence:FREQuency:CONFigure:SOURce?	<source>

System

機能	コマンド	クエリ	レスポンス
Set Date	:SYSTem:DATE <year>,<month>,<day>,<hour>,<minute>,<second>	:SYSTem:DATE?	<year>,<month>,<day>,<hour>,<minute>,<second>
File List	-----	:MMEMory:CATalog? <path>	<num>,<filelist>
Delete File	:MMEMory:DELete <filepath>	-----	-----
User Area Delete	:MMEMory:DELete:USERarea	-----	-----
License Key Check	-----	:SYSTem:BASE:LICense:CHECk? <filename>	<result>
License Key Install	:SYSTem:BASE:LICense:INSTall	-----	-----
Firmware Update	:SYSTem:BASE:UPDate:FIRMware <target>[,<option>]	:SYSTem:BASE:UPDate:FIRMware?	<result>
Check Firmware Update Package	-----	:SYSTem:BASE:UPDate:FIRMware:CHECk? <filename>	<result>
Beep	:SYSTem:BEEPer <sw>	:SYSTem:BEEPer?	<sw>
Go to Local	:SYSTem:COMMunicate:GTL	-----	-----
System Error	-----	:SYSTem:ERRor?	<code>,<msg>
System All Errors	-----	:SYSTem:ERRor:ALL?	<code>,<msg>
System All Error Codes	-----	:SYSTem:ERRor:CODE:ALL?	<code>
System Error Code	-----	:SYSTem:ERRor:CODE[:NEXT]?	<code>
System Error Count	-----	:SYSTem:ERRor:COUNt?	<num>
Language Selection of Remote Command	:SYSTem:LANGuage <mode>	:SYSTem:LANGuage?	<mode>
System Last Error	-----	:SYSTem:LAST:ERRor?	<msg>
Reboot	:SYSTem:REBoot	-----	-----
System Test	-----	:SYSTem:SELF:TEST?	<status>

System (Cont'd)

機能	コマンド	クエリ	レスポンス
System Test Error Last Time	-----	:SYSTem:SELF:TEST:ERRor:LAST time?	<year>,<month>,<day>,<hour>,<min>
System Test Error Count	-----	:SYSTem:SELF:TEST:ERRor:COUNt?	<count>
Shutdown	:SYSTem:SHUTdown	-----	-----

Interface Setting

機能	コマンド	クエリ	レスポンス
Set Direction of RF Connector	:ROUTe:PORT:CONNect:DIRection <input>,<output>[,<output>,…,<output>]	:ROUTe:PORT:CONNect:DIRection?	<input>,<output>[,<output>,…,<output>]
Set In/Out of Trigger Connector	:ROUTe:TRIGger:PORT <trigger>	:ROUTe:TRIGger:PORT?	<trigger>
Select Source of Trigger Input	:ROUTe:TRIGger:INPut:SOURce <source>	:ROUTe:TRIGger:INPut:SOURce?	<source>
Select Source of Trigger Output	:ROUTe:TRIGger:OUTPut:SOURce <source>	:ROUTe:TRIGger:OUTPut:SOURce?	<source>

Loss Correction

機能	コマンド	クエリ	レスポンス
List Count in Loss Table	-----	:CALCulate:EXTLoss:TABLe:COUNt? <index>	<num>
Delete Loss Value from Cable Loss List	:CALCulate:EXTLoss:TABLe:DELete [<a>[,<b>]]	-----	-----
Loss Table	:CALCulate:EXTLoss:TABLe:SETTing <index>	:CALCulate:EXTLoss:TABLe:SETTing?	<index>
Register Loss Value into Cable Loss List	:CALCulate:EXTLoss:TABLe:VALue <frequency>,<Port1_2/out>,<Port1_2/in>,<Port3_4/out>[,<Port3_4/in>]	:CALCulate:EXTLoss:TABLe:VALue? <n>	<frequency>,<Port1_2/out>,<Port1_/in>,<Port3_4/out>,<Port3_4/in>
Register Loss Value into All Cable Loss List	:CALCulate:EXTLoss:TABLe:VALue:ALL <frequency>,<Port1out>,<Port1in>,<Port2out>,<Port2in>,<Port3out>,<Port3in>,<Port4out>,<Port4in>[,<Port5out>,<Port5in>,<Port6out>,<Port6in>,<Port7out>,<Port7in>,<Port8out>,<Port8in>,<Port9out>,<Port9in>,<Port10out>,<Port10in>,<Port11out>,<Port11in>,<Port12out>,<Port12in>]	:CALCulate:EXTLoss:TABLe:VALue:ALL? <n>	<frequency>,<Port1out>,<Port1in>,<Port2out>,<Port2in>,<Port3out>,<Port3in>,<Port4out>,<Port4in>[,<Port5out>,<Port5in>,<Port6out>,<Port6in>,<Port7out>,<Port7in>,<Port8out>,<Port8in>,<Port9out>,<Port9in>,<Port10out>,<Port10in>,<Port11out>,<Port11in>,<Port12out>,<Port12in>]
External Loss On/Off	[:ROUTe]:EXTLoss:TABLe:SWITch <sw>	[:ROUTe]:EXTLoss:TABLe:SWITch?	<sw>

5.1.2 VSGコマンド

AWGN

機能	コマンド	クエリ	レスポンス
AWGN C/N Ratio	:SOURce:GPRF:GENerator:ARB:NOISe:CN <cn_ratio>	:SOURce:GPRF:GENerator:ARB:NOISe:CN?	<cn_ratio>
AWGN Switch	:SOURce:GPRF:GENerator:ARB:NOISe:STATe <on_off>	:SOURce:GPRF:GENerator:ARB:NOISe:STATe?	<on_off>

BaseBand

機能	コマンド	クエリ	レスポンス
Spectrum Reverse	:SOURce:GPRF:GENerator:RFSettings:DM:POLarity <mode>	:SOURce:GPRF:GENerator:RFSettings:DM:POLarity?	<mode>

BaseBand ARB

機能	コマンド	クエリ	レスポンス
Waveform File Loading	:SOURce:GPRF:GENerator:ARB:FILE:LOAD <file_name>	:SOURce:GPRF:GENerator:ARB:FILE:LOAD? <file_name>	<status>
Waveform File Loading Cancel	:SOURce:GPRF:GENerator:ARB:FILE:LOAD:CANCel	-----	-----
Waveform Load Status	-----	:SOURce:GPRF:GENerator:ARB:FILE:LOAD:STATus?	<status_res>
Waveform Group Information	-----	:SOURce:GPRF:GENerator:ARB:FILE:PATTern? <file_name>,<group_index>	<group_number>
Waveform Group Count	-----	:SOURce:GPRF:GENerator:ARB:FILE:PATTern:COUNt? <file_name>	<count>
Waveform Pattern name	-----	:SOURce:GPRF:GENerator:ARB:FILE:PATTern:NAME? <file_name>,<group_number>	<pattern_name>

Baseband ARB (Cont'd)

機能	コマンド	クエリ	レスポンス
Waveform Version	-----	:SOURce:GPRF:GENerator:ARB:FILE:VERSion? <file_name>	<version>
ARB Memory Loaded files count	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:COUNt?	<count>
ARB Memory Defrag	:SOURce:GPRF:GENerator:ARB:WAVeform:DEFRag	-----	-----
Deleting Waveform File in ARB Memory	:SOURce:GPRF:GENerator:ARB:WAVeform:DELete <file_name>	-----	-----
Deleting All Waveform File in ARB Memory	:SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL	-----	-----
ARB Memory Free Space	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:FREE? <target>	<blank>,<consecutive_blank>,<total>
Waveform Generate for Zero Vector Signal Pattern	:SOURce:GPRF:GENerator:ARB:WAVeform:GENerate:ZVSPattern <file_name>,<sampling_rate>,<sample_num>	-----	-----
ARB Memory Loaded file name	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:NAME?<numeric_value>	<file_name>
ARB Memory Group Information	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern? <file_name>,<group_index>	<group_number>
ARB Memory Group Count	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:COUNt? <file_name>	<count>

Baseband ARB (Cont'd)

機能	コマンド	クエリ	レスポンス
ARB Memory Gap Length	:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:GAP <file_name>,<group_number>,<gap_length>	:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:GAP? <file_name>,<group_number>	<gap_length>
ARB Memory Pattern name	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:NAME? <file_name>,<group_number>	<pattern_name>
ARB Memory Pattern Note	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:NOTE? <file_name>,<group_number>	<note>
ARB Memory Pattern Selection	:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect <file_name>[,<group_number>[,<length>]]	:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect?	<file_name>,<group_number>,<length>
ARB Restart	:SOURce:GPRF:GENerator:ARB:WAVeform:RESTart	-----	-----
ARB Sampling Rate	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:SCLock:RATE?	<sampling_rate>
ARB Subsection Switch	:SOURce:GPRF:GENerator:ARB:WAVeform:SSWitch <switch>[,<subsection_number>]	:SOURce:GPRF:GENerator:ARB:WAVeform:SSWitch?	<switch>,<subsection_number>

Baseband ARB (Cont'd)

機能	コマンド	クエリ	レスポンス
Waveform Synchronization Select Pattern	:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect <mode>,<filename>[,<group_number>[,<length>]]	:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect?	<filename>,<group_number>,<length>
Waveform Synchronization State	-----	:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:STATe?	<status>
Waveform Synchronization Cancel	:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:CANCel	-----	-----

Common

機能	コマンド	クエリ	レスポンス
Modulation	:SOURce:GPRF:GENerator:BBMode <mode>	:SOURce:GPRF:GENerator:BBMode?	<mode>
VSG Mode	:SOURce:GPRF:GENerator:MODE <switch>	:SOURce:GPRF:GENerator:MODE?	<switch>
RF Output	:SOURce:GPRF:GENerator:STATe <on_off>	:SOURce:GPRF:GENerator:STATe?	<on_off_res>

Frequency

機能	コマンド	クエリ	レスポンス
Frequency	:SOURce:GPRF:GENerator:RFSettings:FREQuency <freq>	:SOURce:GPRF:GENerator:RFSettings:FREQuency?	<freq>

Level

機能	コマンド	クエリ	レスポンス
Level	:SOURce:GPRF:GENerator:RFSettings:LEVel <level>	:SOURce:GPRF:GENerator:RFSettings:LEVel?	<level>
Level Setting	-----	:SOURce:GPRF:GENerator:RFSettings:LEVel:SETTing?	<status>

Sequence

機能	コマンド	クエリ	レスポンス
Sequence Execution Cancel	:SOURce:GPRF:GENerator:SEQuence:CANCel	-----	-----
Sequence Table Combination Selection	:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern <num>,<table_n>	:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern?	<num>,<table_n>
Sequence Synchronization	:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:SYNChronize <mode>	-----	-----
Sequence Execution	:SOURce:GPRF:GENerator:SEQuence:EXECute [<mode>]	-----	-----
Load Sequence Parameter	:SOURce:GPRF:GENerator:SEQuence:LOAD <list_table_no>,<file_name>	-----	-----
Sequence Reinitialization	:SOURce:GPRF:GENerator:SEQuence:REINitialization <on_off>	:SOURce:GPRF:GENerator:SEQuence:REINitialization?	<on_off>
Sequence Re-Execution	:SOURce:GPRF:GENerator:SEQuence:RLISt	-----	-----
Base Band Mode of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:BBMode <list_table_no>,<segment_no>,<mode>	:SOURce:GPRF:GENerator:SEQuence:RX:BBMode? <list_table_no>,<segment_no>	<mode>
Base Band Mode Group of All Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:BBMode:ALL <list_table_no>,<mode_n>	:SOURce:GPRF:GENerator:SEQuence:RX:BBMode:ALL? <list_table_no>	<mode_n>
End Condition of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition <list_table_no>,<segment_no>,<condition>	:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition? <list_table_no>,<segment_no>	<condition>

Sequence (Cont'd)

機能	コマンド	クエリ	レスポンス
End Condition Group of All Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition:ALL <list_table_no>,<condition_n>	:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition:ALL? <list_table_no>	<condition_n>
Sequence Error Check	-----	:SOURce:GPRF:GENerator:SEQuence:RX:ERRor? <list_table_no>	<error_count>,<error_x_res>
Frequency of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency <list_table_no>,<segment_no>,<freq>	:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency? <list_table_no>,<segment_no>	<freq>
Frequency Group of All Segments in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency:ALL <list_table_no>,<freq_n>	:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency:ALL? <list_table_no>	<freq_n>
Sequence Start Index for continuous mode	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:GOTO <list_table_no>,<index>	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:GOTO? <list_table_no>	<index>
Sequence Repetition Mode	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:REPetition <list_table_no>,<mode>	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:REPetition? <list_table_no>	<mode>
Sequence Start and Stop Segment	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:SSTop <list_table_no>,<start_seg>,<stop_seg>	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:SSTop? <list_table_no>	<start_seg>,<stop_seg>
Sequence Start Segment	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STARt <list_table_no>,<start_seg>	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STARt? <list_table_no>	<start_seg>

Sequence (Cont'd)

機能	コマンド	クエリ	レスポンス
Sequence Step Length	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STEP:LENGth <list_table_no>,<mode>,<time>	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STEP:LENGth? <list_table_no>	<mode>,<time>
Sequence Stop Segment	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STOP <list_table_no>,<stop_seg>	:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STOP? <list_table_no>	<stop_seg>
Level of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:LEVel <list_table_no>,<segment_no>,<level>	:SOURce:GPRF:GENerator:SEQuence:RX:LEVel? <list_table_no>,<segment_no>	<level>
Level Group of All Segments in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:LEVel:ALL <list_table_no>,<level_n>	:SOURce:GPRF:GENerator:SEQuence:RX:LEVel:ALL? <list_table_no>	<level_n>
Loop Control of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl <list_table_no>,<segment_no>,<control>	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl? <list_table_no>,<segment_no>	<control>
Loop Control Group of All Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:ALL <list_table_no>,<control_n>	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:ALL? <list_table_no>	<control_n>
Loop Count of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt <list_table_no>,<segment_no>,<count>	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt? <list_table_no>,<segment_no>	<count>

Sequence (Cont'd)

機能	コマンド	クエリ	レスポンス
Loop Count Group of All Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt:ALL <list_table_no>,<count_n>	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt:ALL? <list_table_no>	<count_n>
Segment Number of Segment in Sequence for Loop Control	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment <list_table_no>,<segment_no>,<jump_segment >	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment? <list_table_no>,<segment_no>	<jump_segment>
Loop Jump Segment Group of All Segment in Sequence for Loop Control	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment:ALL <list_table_no>,<jump_segment_n>	:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment:ALL? <list_table_no>	<jump_segment_n>
Output Port of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe <list_table_no>,<segment_no>,<port>	:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe? <list_table_no>,<segment_no>	<port>
Output Port Group of All Segments in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe:ALL <list_table_no>,<port_n>	:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe:ALL? <list_table_no>	<port_n>
Steps of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt <list_table_no>,<segment_no>,<count>	:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt? <list_table_no>,<segment_no>	<count>
Steps Group of All Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt:ALL <list_table_no>,<count_n>	:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt:ALL? <list_table_no>	<count_n>

Sequence (Cont'd)

機能	コマンド	クエリ	レスポンス
Trigger Delay of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay <list_table_no>,<segment_no>,<delay>	:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay? <list_table_no>,<segment_no>	<delay>
Trigger Delay Group of All Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay:ALL <list_table_no>,<delay_n>	:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay:ALL? <list_table_no>	<delay_n>
Trigger Source of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce <list_table_no>,<segment_no>,<trigger>	:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce? <list_table_no>,<segment_no>	<trigger>
Trigger Source Group of All Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce:ALL <list_table_no>,<trigger_n>	:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce:ALL? <list_table_no>	<trigger_n>
Waveform Change Trigger of Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger <list_table_no>,<segment_no>,<on_off>	:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger? <list_table_no>,<segment_no>	<on_off_res>
Waveform Change Trigger Group of All Segment in Sequence	:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger:ALL <list_table_no>,<on_off_n>	:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger:ALL? <list_table_no>	<on_off_n_res>
Sequence Status	-----	:SOURce:GPRF:GENerator:SEQuence:STATus?	<status>
Save Sequence Parameter	:SOURce:GPRF:GENerator:SEQuence:STORe <list_table_no>,<file_name>	-----	-----

Sequence (Cont'd)

機能	コマンド	クエリ	レスポンス
Pattern End Condition of Sequence Waveform	:SOURce:GPRF:GENerator:SEQuence:WAVeform:ENDCondition <list_table_no>,<index>,<condition>	:SOURce:GPRF:GENerator:SEQuence:WAVeform:ENDCondition? <list_table_no>,<index>	<condition>
Sequence Waveform Spectrum Reverse	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:DM:POLarity <list_table_no>,<mode>	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:DM:POLarity? <list_table_no>	<mode>
Number of Registered Waveform	-----	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:PCOunt? <list_table_no>	<count_res>
Trigger Delay of Sequence Waveform Pattern	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:DELay <list_table_no>,<delay>	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:DELay? <list_table_no>	<delay>
Trigger Source for Sequence Waveform Pattern	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce <list_table_no>,<trigger>	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce? <list_table_no>	<trigger>
Group End Trigger of Sequence Waveform Pattern	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GETRigger <list_table_no>,<index>,<on_off>	:SOURce:GPRF:GENerator:SEQuence:WAVeform:GETRigger? <list_table_no>,<index>	<on_off_res>
Repetition Count of Sequence Waveform Pattern	:SOURce:GPRF:GENerator:SEQuence:WAVeform:IREPetition <list_table_no>,<index>,<count>	:SOURce:GPRF:GENerator:SEQuence:WAVeform:IREPetition? <list_table_no>,<index>	<count>
Manual Trigger for Sequence Waveform	:SOURce:GPRF:GENerator:SEQuence:WAVeform:MTEXecute	-----	-----

Sequence (Cont'd)

機能	コマンド	クエリ	レスポンス
Sequence Waveform Pattern Delete	:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:DELete <list_table_no>,<index>	-----	-----
Sequence Waveform Pattern Configuration	:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:SELect <list_table_no>,<index>,<file_name>[,<group_number>]	:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:SELect? <list_table_no>,<index>	<file_name>,<group_number>
Sequence Waveform Synchronization State	-----	:SOURce:GPRF:GENerator:SEQuence:WAVeform:SYNChronize:STARt:STATe?	<status>

5.2 コマンドの詳細

この節では SCPI コマンドを，アルファベット順に説明しています。

■コマンド表の見方

EX ..... コマンド名(ヘッダー)

Example ..... コマンドの機能名

機能 ..... コマンドの機能

コマンド ..... プログラミングコマンドの構文

クエリ ..... クエリの構文

レスポンス ..... レスポンスの構文

パラメータ ..... パラメータの定義

詳細 ..... コマンドの制約事項など

使用例 ..... コマンドの使用例

関連コマンド ..... 関連するコマンドの紹介

■サフィックスコード一覧

サフィックスコード	単位	サフィックスコード	単位
DBM	dBm	MHZ	MHz
DBUVEMF	dBμV (emf)	MS	ms
DBUVTERM	dBμV (Term)	MZ	MHz
GZ	GHz	NS	ns
GHZ	GHz	S	s
HZ	Hz	US	μs
KZ	kHz	Z	Hz
KHZ	kHz		

5.2.1　基本操作コマンド

:CALCulate:CALibration:BAND:ERRor:COUNt?

Band Calibration Error Count

機能

バンド校正を失敗した回数を問い合わせます。

クエリ

```
:CALCulate:CALibration:BAND:ERRor:COUNt?
```

レスポンス

<count>

パラメータ

<count>		バンド校正を失敗した回数
	範囲	0～

使用例

バンド校正を失敗した回数を問い合わせます。

:CALC:CAL:BAND:ERR:COUN?

> 0

:CALCulate:CALibration:BAND:ERRor:LASTtime?

BAND Calibration Error Last Time

機能

最後にバンド校正を失敗した日時を問い合わせます。

クエリ

```
:CALCulate:CALibration:BAND:ERRor:LASTtime?
```

レスポンス

<year>,<month>,<day>,<hour>,<min>

パラメータ

<year>	年 (2001～)
<month>	月 (1～12)
<day>	日 (1～31)
<hour>	時 (0～23)
<min>	分 (0～59)

使用例

最後にバンド校正を失敗した時刻を問い合わせます。

:CALC:CAL:BAND:ERR:LAST?

> 2011,09,13,21,45

備考

バンド校正を 1 度も失敗していない場合, レスポンスは"***"になります。

:CALCulate:CALibration:BAND:LASTtime?

BAND Calibration Last Time

機能

最後にバンド校正を実行した日時を問い合わせます。

クエリ

```
:CALCulate:CALibration:BAND:LASTtime?
```

レスポンス

<year>,<month>,<day>,<hour>,<min>

パラメータ

<year>	年 (2001～)
<month>	月 (1～12)
<day>	日 (1～31)
<hour>	時 (0～23)
<min>	分 (0～59)

使用例

最後にバンド校正を実行したときの時刻を問い合わせます。

:CALC:CAL:BAND:LAST?

> 2011,09,13,11,45

備考

バンド校正を 1 度も実行していない場合, レスポンスは"***"になります。

:CALCulate:CALibration:BAND:RESult?

Result of Band Calibration

機能

バンド校正の結果を問い合わせます。

クエリ

```
:CALCulate:CALibration:BAND:RESult?
```

レスポンス

<Result>

パラメータ

<Result>	バンド校正の結果
PASS	Calibration 成功
FAIL	Calibration 失敗
UNEXECUTED	Calibration 未実行

使用例

バンド校正の結果を問い合わせます。

:CALC:CAL:BAND:RES?

> PASS

:CALCulate:CALibration:BAND:STARt

Execute BAND Calibration

機能

バンド校正（Band Calibration）を実行します。

コマンド

```
:CALCulate:CALibration:BAND:STARt
```

使用例

Band Calibration を実行します。
:CALC:CAL:BAND:STAR

:CALCulate:CALibration:BAND:STARt:TEMPerature

BAND Calibration Temperature

機能

内部温度が，前回バンド校正を実行した時点より指定値以上変化している場合，バンド校正を実行します。
内部温度変化が指定値未満の場合は，バンド校正を実行しません。

コマンド

```
:CALCulate:CALibration:BAND:STARt:TEMPerature <temperature>
```

パラメータ

<temperature>	温度
範囲	1.0～50.0 ℃
分解能	0.1 ℃

使用例

内部温度が，前回バンド校正を実行した時点より 2℃以上変化している場合，バンド校正を実行します。
:CALC:CAL:BAND:STAR:TEMP 2.0

:CALCulate:CALibration:ELAPsed:TIME?

Elapsed Time after Full Calibration

機能

最後に全バンド校正を実行してからの経過時間を問い合わせます。

クエリ

```
:CALCulate:CALibration:ELAPsed:TIME?
```

レスポンス

<time>

パラメータ

<time>		経過時間　秒（seconds）単位の時間

使用例

最後に全バンド校正を実行してからの経過時間を問い合わせます。

:CALC:CAL:ELAP:TIME?

> 1280

備考

全バンド校正を 1 度も実行していない場合，レスポンスは"***"になります。

:CALCulate:CALibration:FULL:ERRor:COUNt?

Full Calibration Error Count

機能

全バンド校正を失敗した回数を問い合わせます。

クエリ

```
:CALCulate:CALibration:FULL:ERRor:COUNt?
```

レスポンス

<count>

パラメータ

<count>		全バンド校正を失敗した回数
	範囲	0 以上

使用例

全バンド校正を失敗した回数を問い合わせます。

:CALC:CAL:FULL:ERR:COUN?

> 0

:CALCulate:CALibration:FULL:ERRor:LASTtime?

FULL Calibration Error Last Time

機能

最後に全バンド校正を失敗した日時を問い合わせます。

クエリ

```
:CALCulate:CALibration:FULL:ERRor:LASTtime?
```

レスポンス

<year>,<month>,<day>,<hour>,<min>

パラメータ

<year>	年 (2001～)
<month>	月 (1～12)
<day>	日 (1～31)
<hour>	時 (0～23)
<min>	分 (0～59)

使用例

最後に全バンド校正を失敗したときの時刻を問い合わせます。

:CALC:CAL:FULL:ERR:LAST?

> 2012,03,10,11,45

備考

全バンド校正を 1 度も失敗していない場合, レスポンスは"***"になります。

:CALCulate:CALibration:FULL:LASTtime?

FULL Calibration Last Time

機能

最後に全バンド校正を実行した日時を問い合わせます。

クエリ

```
:CALCulate:CALibration:FULL:LASTtime?
```

レスポンス

<year>,<month>,<day>,<hour>,<min>

パラメータ

<year>	年 (2001～)
<month>	月 (1～12)
<day>	日 (1～31)
<hour>	時 (0～23)
<min>	分 (0～59)

使用例

最後に全バンド校正を実行したときの時刻を問い合わせます。

:CALC:CAL:FULL:LAST?

> 2012,03,13,11,45

備考

全バンド校正を 1 度も実行していない場合, レスポンスは"***"になります。

:CALCulate:CALibration:FULL:RESult?

Result of Full Calibration

機能

全バンド校正の結果を問い合わせます。

クエリ

```
:CALCulate:CALibration:FULL:RESult?
```

レスポンス

<Result>

パラメータ

<Result>	全バンド校正の結果
PASS	Calibration 成功
FAIL	Calibration 失敗
UNEXECUTED	Calibration 未実行

使用例

全バンド校正の結果を問い合わせます。
:CALC:CAL:FULL:RESult?
> PASS

:CALCulate:CALibration:FULL:STARt

Execute FULL Calibration

機能

全バンド校正を実行します。

コマンド

```
:CALCulate:CALibration:FULL:STARt
```

詳細

MU887002A-007 搭載時，校正の実行時間は MU887002A-007 非搭載時よりも長くなります。

使用例

全バンド校正を実行します。
:CALC:CAL:FULL:STAR

:CALCulate:CALibration:INITial

Initial Calibration

機能

全バンド校正 / バンド校正の値を初期化します。

コマンド

```
:CALCulate:CALibration:INITial
```

使用例

全バンド校正 / バンド校正の値を初期化します。

:CALC:CAL:INIT

備考

全バンド校正 / バンド校正の最後に実施した日時，実行結果も初期化します。

最後に失敗した日時，失敗した回数は初期化されません。

:CALCulate:CALibration:TEMPerature?

Temperature at Full Calibration Execution

機能

電源を入れてから最後に全バンド校正を実行したときのモジュールの温度を問い合わせます。

クエリ

```
:CALCulate:CALibration:TEMPerature?
```

レスポンス

<temp>

パラメータ

<temp>	温度（℃）

使用例

電源を入れてから最後に全バンド校正を実行したときのモジュールの温度を問い合わせます。

:CALC:CAL:TEMP?

> 35.31

備考

電源を入れてから全バンド校正を 1 度も実行していない場合，レスポンスは"***"になります。

:CALCulate:EXTLoss:TABLe:COUNt?

List Count in Loss Table

機能

損失補正テーブルに設定されているリストの数を問い合わせます。

クエリ

```
:CALCulate:EXTLoss:TABLe:COUNt? <index>
```

レスポンス

<num>

パラメータ

<index>		補正対象とするテーブルのインデックス番号
	範囲	1～16
	分解能	1
<num>		レベル周波数補正テーブルに登録されているリストの数
	範囲	0～1000

パラメータ

使用例

:CALCulate:EXTLoss:TABLe:COUNt? 1
> 3

:CALCulate:EXTLoss:TABLe:DELete

Delete Loss Value from Cable Loss List

機能

損失補正テーブルのリストに設定された補正値を消去します。
削除範囲は，登録されている損失補正テーブルのリスト No.に相当し，削除したい範囲を指定します。
削除範囲を指定しなかった場合は，損失補正テーブルのリストの値を全削除します。

コマンド

```
:CALCulate:EXTLoss:TABLe:DELete [<a>[,<b>]]
```

パラメータ

<a>	削除範囲指定
範囲	1～1000
分解能	1
<b>	削除範囲指定
範囲	a～1000
分解能	1

使用例

:CALCulate:EXTLoss:TABLe:DELete 1,1000

:CALCulate:EXTLoss:TABLe:SETTing

Loss Table

機能

損失補正テーブルのインデックスを設定，または問い合わせます。

コマンド

```
:CALCulate:EXTLoss:TABLe:SETTing <index>
```

クエリ

```
:CALCulate:EXTLoss:TABLe:SETTing?
```

レスポンス

<index>

パラメータ

<index>	補正テーブルのインデックス
範囲	1～16
分解能	1
初期値	1

使用例

:CALCulate:EXTLoss:TABLe:SETTing 1
:CALCulate:EXTLoss:TABLe:SETTing?
> 1

:CALCulate:EXTLoss:TABLe:VALue

Register Loss Value into Cable Loss List

機能

損失補正テーブルのリストの Port 1～Port 4 に値を設定，または問い合わせします。
損失補正テーブルのリスト内に同一周波数がある場合は上書きされます。
[Port3_4/in]の設定を省略した場合，[Port3_4/out] の値を損失補正テーブルのリストに設定します。

損失補正テーブルのリストは，任意の周波数ポイントの損失値を入力し，設定ポイント間の周波数について線形補間により自動的に補正値を決定します。
Port1 と Port2 の補正値は同じ値が設定されます。Port3 と Port4 の補正値も同じ値が設定されます。

コマンド

```
:CALCulate:EXTLoss:TABLe:VALue
<frequency>,<Port1_2/out>,<Port1_2/in>,<Port3_4/out>[,<Port3_4/in>]
```

クエリ

```
:CALCulate:EXTLoss:TABLe:VALue? <n>
```

レスポンス

<frequency>,<Port1_2/out>,<Port1_2/in>,<Port3_4/out>,<Port3_4/in>

パラメータ

<frequency>	設定する周波数ポイント
範囲	10 MHz～7300 MHz
分解能	1 Hz
サフィックスコード	HZ, Z, KHZ, KZ, MHZ, MZ, GHZ, GZ 省略した場合は HZ として扱われます。
<Port1_2/out>	Port1/2 の DL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
<Port1_2/in>	Port1/2 の UL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
<Port3_4/out>	Port3/4 の DL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
<Port3_4/in>	Port3/4 の UL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
<n>	問い合わせたい補正テーブルポイント

範囲　　1～1000
分解能　　1

詳細

複数の Port から信号出力する場合，Port 間の損失補正テーブル値の差が 12 dB を超えると信号は設定した出力レベルよりも低く出力されます。この場合，ランプ 2 が橙色に点灯します。
ただし，8 dB を超えた場合でも，出力レベルの誤差が大きくなる可能性があります。

使用例

損失補正テーブルのリストに次の表の値を設定します。

No	周波数	Port 1 (dB)		Port 2 (dB)		Port 3 (dB)		Port 4 (dB)	
		out	in	out	in	out	in	out	in
1	1000000000	10.00	11.00	10.00	11.00	12.00	13.00	12.00	13.00

:CALCulate:EXTLoss:TABLe:VALue 1GHz,10,11,12,13
:CALCulate:EXTLoss:TABLe:VALue? 1
> 1000000000,10.00,11.00,12.00,13.00

備考

損失補正テーブルのリストは，各測定ソフトウェア共通の補正値として保持されます。
周波数に対して昇順となるように，損失補正テーブルのリストは自動で並び替わります。

:CALCulate:EXTLoss:TABLe:VALue:ALL

Register Loss Value into All Cable Loss List

機能

損失補正テーブルのリストに値を設定，または問い合わせします。
損失補正テーブルのリスト内に同一周波数がある場合は上書きされます。

損失補正テーブルのリストは，任意の周波数ポイントの損失値を入力し，設定ポイント間の周波数について線形補間により自動的に補正値を決定します。
Port 1～12 で別々の補正値を設定できます。

コマンド

```
:CALCulate:EXTLoss:TABLe:VALue:ALL
<frequency>,<Port1out>,<Port1in>,<Port2out>,<Port2in>,<Port3out>,<Port3in
>,<Port4out>,<Port4in>[,<Port5out>,<Port5in>,<Port6out>,<Port6in>,<Port7o
ut>,<Port7in>,<Port8out>,<Port8in>,<Port9out>,<Port9in>,<Port10out>,<Port
10in>,<Port11out>,<Port11in>,<Port12out>,<Port12in>]
```

クエリ

```
:CALCulate:EXTLoss:TABLe:VALue:ALL? <n>
```

レスポンス

<frequency>,<Port1out>,<Port1in>,<Port2out>,<Port2in>,<Port3out>,<Port3in>,<Port4out>,<Port4in>[,<Port5out>,<Port5in>,<Port6out>,<Port6in>,<Port7out>,<Port7in>,<Port8out>,<Port8in>,<Port9out>,<Port9in>,<Port10out>,<Port10in>,<Port11out>,<Port11in>,<Port12out>,<Port12in>]

パラメータ

<frequency>	設定する周波数ポイント
範囲	10 MHz～7300 MHz
分解能	1 Hz
サフィックスコード	HZ, Z, KHZ, KZ, MHZ, MZ, GHZ, GZ 省略した場合は HZ として扱われます。
<Port1out>	Port 1 の DL の損失値
…	…
<Port12out>	Port 12 の DL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB,　省略した場合は DB として扱われます。
<Port1in>	Port 1 の UL の損失値
…	…
<Port12in>	Port 12 の UL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB　省略した場合は DB として扱われます。
<n>	問い合わせたい補正テーブルポイント
範囲	1～1000
分解能	1

詳細

Port 1～4 の設定は必須です。損失値を設定しない Port は，0 dB とします。
Port 5～12 は，Port 5～12 すべて設定しない場合のみ省略できます。損失値を設定する場合は，すべてに設定します。損失値を設定しない Port は，0 dB とします。
複数の Port から信号出力する場合，Port 間の損失補正テーブル値の差が 12 dB を超えると信号は設定した出力レベルよりも低く出力されます。この場合，ランプ 2 が橙色に点灯します。
ただし，8 dB を超えた場合でも，出力レベルの誤差が大きくなる可能性があります。

使用例

損失補正テーブルのリストに次の表の値を設定します。

No	周波数	Port 1 (dB)		Port 2 (dB)		Port 3 (dB)		Port 4 (dB)	
		out	in	out	in	out	in	out	in
1	1000000000	10.00	11.00	12.00	13.00	14.00	15.00	16.00	17.00
		Port 5 (dB)		Port 6 (dB)		Port 7 (dB)		Port 8 (dB)	
		out	in	out	in	out	in	out	in
		18.00	19.00	20.00	21.00	22.00	23.00	24.00	25.00
		Port 9 (dB)		Port 10 (dB)		Port 11 (dB)		Port 12 (dB)	
		out	in	out	in	out	in	out	in
		26.00	27.00	28.00	29.00	30.00	31.00	32.00	33.00

:CALCulate:EXTLoss:TABLe:VALue:ALL 1GHZ,10,11,12,13,14,15,16,17,18,19,20,
21,22,23,24,25,26,27,28,29,30,31,32,33
:CALCulate:EXTLoss:TABLe:VALue:ALL? 1
> 1000000000,10.00,11.00,12.00,13.00,14.00,15.00,16.00,17.00,18.00,19.00,20.00,
21.00,22.00,23.00,24.00,25.00,26.00,27.00,28.00,29.00,30.00,31.00,32.00,33.00

備考

損失補正テーブルのリストは，各測定ソフトウェア共通の補正値として保持されます。
周波数に対して昇順となるように，損失補正テーブルのリストは自動で並び替わります。

:MAINtenance:SYSTem:INFormation:DEVice:MODel?

Module Model Number

機能

モジュールのモデル番号を問い合わせます。

クエリ

```
:MAINtenance:SYSTem:INFormation:DEVice:MODel?
```

レスポンス

<model>

パラメータ

<model>	モデル番号

使用例

```
:MAINtenance:SYSTem:INFormation:DEVice:MODel?
> MU887002A
```

:MAINtenance:SYSTem:INFormation:MAINframe:MODel?

Main Frame Model Number.

機能

MT8870A のモデル番号を問い合わせます。

クエリ

```
:MAINtenance:SYSTem:INFormation:MAINframe:MODel?
```

レスポンス

<model>

パラメータ

<model>	モデル番号

使用例

```
:MAINtenance:SYSTem:INFormation:MAINframe:MODel?
> MT8870A
```

:MMEMory:CATalog?

File List

機能

指定したパスに保存されているファイルの一覧を問い合わせます。

クエリ

```
:MMEMory:CATalog? <path>
```

レスポンス

<num>,<filelist>

パラメータ

<path>	パス
<num>	ファイル数
<filelist>	ファイル一覧

パラメータ

詳細

レスポンスが 10485760 文字以上のリストになる場合は, レスポンスの最後に"…"が付きます。
パスには"*.h"のようにファイル拡張子を指定できます。

使用例

:MMEM:CAT? "CELLULAR/parameter"
> 3,setup.000,setup.001,setup.002

:MMEM:CAT? "CELLULAR/parameter/*001"
> 1,setup.001

:MMEMory:DELete

Delete File

機能

指定したファイルを削除します。

コマンド

```
:MMEMory:DELete <filepath>
```

パラメータ

<filepath>　　ファイル名（パス情報を含みます）

詳細

ファイル名はパスを含めた形式で指定します。

使用例

```
:MMEM:CAT? "CELLULAR/parameter"
> 3,setup.000,setup.001,setup.002
:MMEM:DEL "CELLULAR/parameter/setup.002"
:MMEM:CAT? "CELLULAR/parameter"
> 2,setup.000,setup.001
```

:MMEMory:DELete:USERarea

User Area delete

機能

ユーザエリアを削除します。

コマンド

```
:MMEMory:DELete:USERarea
```

詳細

FTP を使用してアクセスできるモジュール内部の記憶媒体上のデータを削除します。
波形ファイルなど /user 以下にあるファイルがすべて削除されます。

使用例

```
:MMEMory:DELete:USERarea
```

:MMEMory:LOG:CLEar

Log Clear

機能

ログを消去します。

コマンド

```
:MMEMory:LOG:CLEar
```

詳細

モジュールのログファイルを消去します。

使用例

:MMEMory:LOG:SIZE?
> 2147483647
:MMEMory:LOG:CLEar

:MMEMory:LOG:LOAD?

Log Reading

機能

ログ情報を問い合わせます。

クエリ

```
:MMEMory:LOG:LOAD? [<line>]
```

レスポンス

#XY<Y byte binary data><terminator>

X および Y はアスキー文字で，数値 X は数値 Y の桁数です。Y が 12500 の場合，X は 5 になります。
バイナリデータ<Y byte binary data>は，ログファイル内の文字列です。

パラメータ

<line>	最新のログ取得行数
範囲	1～999999
<terminator>	ターミネータ

詳細

パラメータでログ取得行数を指定した場合は，その行数分のログを取得します。ログ取得サイズを省略した場合は，すべてのログを取得します。
ログファイル内の改行コードは，タブに置き換えて表示されます。

使用例

:MMEMory:LOG:LOAD?
> CELLULAR Measurement Software License expired.¥n

:MMEM:LOAD:LOG? 2
> CELLULAR Measurement Software License expired.¥n
> R[*IDN?]¥n

:MMEMory:LOG:SIZE?

Log Size Reading

機能

ログサイズを問い合わせます。

クエリ

```
:MMEMory:LOG:SIZE?
```

レスポンス

<size>

パラメータ

<size>	ログのサイズ

詳細

レスポンスは，ログファイルのバイトサイズです。

使用例

:MMEMory:LOG:SIZE?
> 48

:MMEMory:SIZE?

Drive Size Reading

機能

記憶媒体の空き容量を問い合わせます。

クエリ

```
:MMEMory:SIZE? [<target>]
```

レスポンス

<total>,<blank>

パラメータ

<target>	空き容量を確認するエリア
LOG	ログを記録するエリア
USER	波形ファイル，インストーラファイルなどに使用するエリア
初期値	USER
<total>	指定エリアの総容量（バイト単位）
<blank>	指定エリアの空き容量（バイト単位）

使用例

:MMEMory:SIZE? LOG
> 2097152,734002

[:ROUTe]:EXTLoss:TABLe:SWITch

External Loss On/Off

機能

損失補正テーブルのリストの全ポートの DL/UL の損失データを使用して，レベル補正するかを設定，または問い合わせます。

コマンド

```
[:ROUTe]:EXTLoss:TABLe:SWITch <sw>
```

クエリ

```
[:ROUTe]:EXTLoss:TABLe:SWITch?
```

レスポンス

<sw>

パラメータ

<sw>	全ポートの DL/UL の損失値の有効/無効
COMMON	補正値を使用します。
ON	補正値を使用します。
OFF	補正値を使用しません。
初期値	OFF

詳細

複数の Port から信号出力する場合，Port 間の損失補正テーブル値の差が 12 dB を超えると信号は設定した出力レベルよりも低く出力されます。この場合，ランプ 2 が橙色に点灯します。
ただし，8 dB を超えた場合でも，出力レベルの誤差が大きくなる可能性があります。

使用例

[:ROUTe]:EXTLoss:TABLe:SWITch ON
[:ROUTe]:EXTLoss:TABLe:SWITch?
> ON

備考

損失補正テーブルのリストに設定されていない周波数の補正値には，前後の周波数の補正値を用いて線形補間で算出した値を使用します。
最小周波数未満の周波数の場合には最小周波数の補正値を，最大周波数より大きい周波数の場合には最大周波数の補正値を使用します。

:ROUTe:PORT:CONNect:DIRection

Set Direction of RF Connector

機能

RF 信号を入力するテストポートと出力するテストポートを設定，または問い合わせます。

コマンド

```
:ROUTe:PORT:CONNect:DIRection <input>,<output>[,<output>,…,<output>]
```

クエリ

```
:ROUTe:PORT:CONNect:DIRection?
```

レスポンス

<input>,<output>[,<output>,…,<output>]

パラメータ

<input>	テストポート番号
範囲	OFF, Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
OFF	RF 信号を入力するポートなし。
初期値	PORT1
<output>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
初期値	PORT1

詳細

Test Port 1～Test Port 12 は入力と出力の両方に設定できます。

MU887002A-007 搭載時，以下の制限事項があります。設定コマンドは有効ですが実行エラーとなります。

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では，入力または出力のどちらかにしか設定できません。

使用例

RF 信号を入力するテストポートを Test Port 1，RF 信号を出力するテストポートを Test Port 2 に設定します。

```
:ROUTe:PORT:CONNect:DIRection PORT1,PORT2
:ROUTe:PORT:CONNect:DIRection?
> PORT1,PORT2
```

RF 信号入力テストポートなし，RF 信号出力テストポートを Test Port 4，Test Port 6，Test Port 10 に設定します。

```
:ROUT:PORT:CONN:DIR OFF,PORT10,PORT4,PORT6
:ROUT:PORT:CONN:DIR?
> OFF,PORT4,PORT6,PORT10
```

:ROUTe:TRIGger:INPut:SOURce

Select Source of Trigger Input

機能

外部トリガとして使用する信号を設定，または問い合わせます。

コマンド

```
:ROUTe:TRIGger:INPut:SOURce <source>
```

クエリ

```
:ROUTe:TRIGger:INPut:SOURce?
```

レスポンス

<source>

パラメータ

<source>	外部トリガとして使用する信号
TRGIN	制御対象のモジュールの TRX が挿入されているスロット用トリガコネクタに入力されるトリガ
TRGIN1	トリガコネクタ 1 に入力されるトリガ
SLOT1	スロット 1 のモジュールの TRX から出力されるトリガ信号
初期値	TRGIN

詳細

TRGIN を選択する際は，Trigger コネクタの入出力設定を入力にしてください。
次のコマンドによって外部トリガとして使用する信号が変更されることがあります。

:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect
:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:SYNChronize

使用例

:ROUTe:TRIGger:INPut:SOURce SLOT1
:ROUTe:TRIGger:INPut:SOURce?
> SLOT1

関連コマンド

:ROUTe:TRIGger:PORT

:ROUTe:TRIGger:OUTPut:SOURce

Select Source of Trigger Output

機能

MT8870A 背面パネルのトリガコネクタに出力する信号を設定，または問い合わせます。

コマンド

```
:ROUTe:TRIGger:OUTPut:SOURce <source>
```

クエリ

```
:ROUTe:TRIGger:OUTPut:SOURce?
```

レスポンス

<source>

パラメータ

<source>	トリガコネクタに出力する信号
WFM1	SG 波形マーカ 1
WFM2	SG 波形マーカ 2
WFM3	SG 波形マーカ 3
SGSYNC	SG 同期開始信号
NONE	出力信号なし
初期値	NONE

詳細

Trigger コネクタの入出力設定を出力にした際に，本コマンドで選択した信号が Trigger コネクタに出力されます。

次のコマンドによってトリガコネクタに出力する信号が変更されることがあります。

:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect

:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:SYNChronize

使用例

:ROUTe:TRIGger:OUTPut:SOURce WFM1

:ROUTe:TRIGger:OUTPut:SOURce?

> WFM1

関連コマンド

:ROUTe:TRIGger:PORT

:ROUTe:TRIGger:PORT

Set In/Out of Trigger Connector

機能

MT8870A 背面パネルのトリガコネクタの入出力を設定，または問い合わせます。

コマンド

```
:ROUTe:TRIGger:PORT <trigger>
```

クエリ

```
:ROUTe:TRIGger:PORT?
```

レスポンス

<trigger>

パラメータ

<trigger>	トリガコネクタの入出力設定
INPUT	入力
OUTPUT	出力
初期値	INPUT

詳細

モジュールの TRX ごとにトリガコネクタの入出力を設定できます。

次のコマンドによってトリガコネクタの入出力が変更されることがあります。

:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect

:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:SYNChronize

使用例

:ROUTe:TRIGger:PORT INPUT

:ROUTe:TRIGger:PORT?

> INPUT

:STATus:OPERation:CONDition?

Operation Status Condition Register Query

機能

実行状態コンディションレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:OPERation:CONDition?
```

レスポンス

<oscr>
レジスタのビット割り当ては，:STATus:OPERation[:EVENt]の説明を参照してください。

パラメータ

<oscr>	実行状態コンディションレジスタ
範囲	0～65535

詳細

実行状態コンディションレジスタのbit0，1～15に対応する値$2^0 = 1$，$2^1 = 2$～$2^{15} = 32768$の中から，発生したイベントのビットの値の総和がレスポンスとなります。

使用例

実行状態コンディションレジスタの内容を問い合わせます。

:STAT:OPER:COND?
>0

:STATus:OPERation:ENABle

Operation Status Enable Register

機能

実行状態イネーブルレジスタの値を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:ENABle <oser>
```

クエリ

```
:STATus:OPERation:ENABle?
```

レスポンス

<oser>

レジスタのビット割り当ては，:STATus:OPERation[:EVENt]の説明を参照してください。

パラメータ

<oser>		実行状態イネーブルレジスタ
	範囲	0～65535
	分解能	1
	初期値	0

詳細

ステータスバイトレジスタに検出結果を出力するビットを設定します。

使用例

実行状態イベントレジスタのビット 4 だけを読みとれるようにします。
このとき，実行状態イネーブルレジスタには 2^4=16 を設定します。

```
:STAT:OPER:ENAB 16
```

実行状態イネーブルレジスタの値を問い合わせます。

```
:STAT:OPER:ENAB?
>16
```

:STATus:OPERation[:EVENt]?

Operation Status Register Query

機能

実行状態レジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:OPERation[:EVENt]?
```

レスポンス

<osr>　実行状態イベントレジスタ

値=bit0 + bit1 + ... + bit15

bit0 = 2^0 = 1	未使用
bit1 = 2^1 = 2	未使用
bit2 = 2^2 = 4	未使用
bit3 = 2^3 = 8	未使用
bit4 = 2^4 = 16	未使用
bit5 = 2^5 = 32	未使用
bit6 = 2^6 = 64	未使用
bit7 = 2^7 = 128	未使用
bit8 = 2^8 = 256	測定
bit9 = 2^9 = 512	信号発生器
bit10 = 2^{10} = 1024	未使用
bit11 = 2^{11} = 2048	未使用
bit12 = 2^{12} = 4096	未使用
bit13 = 2^{13} = 8192	未使用
bit14 = 2^{14} = 16384	未使用
bit15 = 2^{15} = 32768	未使用

パラメータ

<osr>	実行状態レジスタ
範囲	0～65535

詳細

実行状態レジスタの bit0，1～15 に対応する値 2^0 = 1，2^1 = 2～2^{15}= 32768 の中から，発生したイベントのビットの値の総和がレスポンスとなります。

使用例

実行状態レジスタの内容を問い合わせます。

```
:STAT:OPER?
>512
```

:STATus:OPERation:GENerator:CONDition?

Generator Operation Status Condition Register Query

機能

信号発生器実行状態コンディションレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:OPERation:GENerator:CONDition?
```

レスポンス

<goscr>
レジスタのビット割り当ては，:STATus:OPERation:GENerator[:EVENt]の説明を参照してください。

パラメータ

<goscr>	信号発生器実行状態コンディションレジスタ
範囲	0～65535

詳細

:STATus:OPERation:CONDition と同様です。

:STATus:OPERation:GENerator:ENABle

Generator Operation Status Enable Register

機能

信号発生器実行状態レジスタの値を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:GENerator:ENABle <goser>
```

クエリ

```
:STATus:OPERation:GENerator:ENABle?
```

レスポンス

<goser>
レジスタのビット割り当ては，:STATus:OPERation:GENerator[:EVENt]の説明を参照してください。

パラメータ

<goser>	信号発生器実行状態レジスタ
範囲	0～65535
分解能	1
初期値	0

詳細

:STATus:OPERation:ENABle と同様です。

:STATus:OPERation:GENerator[:EVENt]?

Generator Operation Status Register Query

機能

信号発生器実行状態レジスタの値を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:OPERation:GENerator[:EVENt]?
```

レスポンス

<gosr>
値=bit0 + bit1 + ... + bit15

bit0 = 2^0 = 1	ファイルロード中
bit1 = 2^1 = 2	デフラグ実行中
bit2 = 2^2 = 4	波形パターンの同期出力準備中
bit3 = 2^3 = 8	未使用
bit4 = 2^4 = 16	未使用
bit5 = 2^5 = 32	未使用
bit6 = 2^6 = 64	未使用
bit7 = 2^7 = 128	未使用
bit8 = 2^8 = 256	未使用
bit9 = 2^9 = 512	未使用
bit10 = 2^{10} = 1024	未使用
bit11 = 2^{11} = 2048	未使用
bit12 = 2^{12} = 4096	未使用
bit13 = 2^{13} = 8192	未使用
bit14 = 2^{14} = 16384	未使用
bit15 = 2^{15} = 32768	未使用

パラメータ

<gosr>	信号発生器実行状態レジスタ
範囲	0～65535

詳細

信号発生器状態レジスタの bit0，1～15 に対応する値 2^0 = 1，2^1 = 2～2^{15}= 32768 の中から，発生したイベントのビットの値の総和がレスポンスとなります。

使用例

信号発生器実行状態レジスタの内容を問い合わせます。

```
:STAT:OPER:GEN?
> 1
```

:STATus:OPERation:GENerator:NTRansition

Generator Operation Status Negative Transition Filter

機能

信号発生器実行状態レジスタの遷移フィルタ（負方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:GENerator:NTRansition <gosr_ntf>
```

クエリ

```
:STATus:OPERation:GENerator:NTRansition?
```

レスポンス

<gosr_ntf>

レジスタのビット割り当ては，:STATus:OPERation:GENerator[:EVENt]の説明を参照してください。

パラメータ

<gosr_ntf>		信号発生器実行状態レジスタの遷移フィルタ（負方向変化）
	範囲	0～65535
	分解能	1
	初期値	0

詳細

:STATus:OPERation:NTRansition と同様です。

:STATus:OPERation:GENerator:PTRansition

Generator Operation Status Positive Transition Filter

機能

信号発生器実行状態レジスタの遷移フィルタ（正方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:GENerator:PTRansition <gosr_ptf>
```

クエリ

```
:STATus:OPERation:GENerator:PTRansition?
```

レスポンス

<gosr_ptf>

レジスタのビット割り当ては，:STATus:OPERation:GENerator[:EVENt]の説明を参照してください。

パラメータ

<gosr_ptf>	信号発生器実行状態レジスタの遷移フィルタ（正方向変化）	
範囲	0～65535	
分解能	1	
初期値	65535	

詳細

:STATus:OPERation:PTRansition と同様です。

:STATus:OPERation:MEASure:CONDition?

Measurement Operation Status Condition Register Query

機能

測定実行状態コンディションレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:OPERation:MEASure:CONDition?
```

レスポンス

<moscr>
レジスタのビット割り当ては，:STATus:OPERation:MEASure[:EVENt]の説明を参照してください。

パラメータ

<moscr>	測定実行状態コンディションレジスタ
範囲	0～65535

詳細

:STATus:OPERation:CONDition と同様です。

使用例

測定実行状態コンディションレジスタの内容を問い合わせます。

```
:STAT:OPER:MEAS:COND?
> 1
```

:STATus:OPERation:MEASure:ENABle

Measurement Operation Status Enable Register

機能

測定実行状態イネーブルレジスタの値を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:MEASure:ENABle <moser>
```

クエリ

```
:STATus:OPERation:MEASure:ENABle?
```

レスポンス

<moser>

レジスタのビット割り当ては，:STATus:OPERation:MEASure[:EVENt]の説明を参照してください。

パラメータ

<moser>	測定実行状態イネーブルレジスタ
範囲	0～65535
分解能	1
初期値	0

詳細

:STATus:OPERation:ENABle と同様です。

:STATus:OPERation:MEASure[:EVENt]?

Measurement Operation Status Register Query

機能

測定実行状態レジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:OPERation:MEASure[:EVENt]?
```

レスポンス

<mosr>
値=bit0 + bit1 + ... + bit15

bit0 = 2^0 = 1	測定中
bit1 = 2^1 = 2	未使用
bit2 = 2^2 = 4	未使用
bit3 = 2^3 = 8	未使用
bit4 = 2^4 = 16	未使用
bit5 = 2^5 = 32	未使用
bit6 = 2^6 = 64	未使用
bit7 = 2^7 = 128	未使用
bit8 = 2^8 = 256	未使用
bit9 = 2^9 = 512	未使用
bit10 = 2^{10} = 1024	未使用
bit11 = 2^{11} = 2048	未使用
bit12 = 2^{12} = 4096	未使用
bit13 = 2^{13} = 8192	未使用
bit14 = 2^{14} = 16384	未使用
bit15 = 2^{15} = 32768	未使用

ビットの割り当てはアプリケーションによって変わります。
アプリケーションソフトウェアの取扱説明書を参照してください。

パラメータ

<mosr>	測定実行状態レジスタ
範囲	0～65535

詳細

測定実行状態レジスタの bit0, 1～15 に対応する値 2^0 = 1, 2^1 = 2～2^{15} = 32768 の中から, 発生したイベントのビットの値の総和がレスポンスとなります。

使用例

測定実行状態レジスタの内容を問い合わせます。

```
:STAT:OPER:MEAS?
> 1
```

:STATus:OPERation:MEASure:NTRansition

Measurement Operation Status Negative Transition Filter

機能

測定実行状態レジスタの遷移フィルタ（負方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:MEASure:NTRansition <mosr_ntr>
```

クエリ

```
:STATus:OPERation:MEASure:NTRansition?
```

レスポンス

<mosr_ntr>

レジスタのビット割り当ては，:STATus:OPERation:MEASure[:EVENt]の説明を参照してください。

パラメータ

<mosr_ntr>	測定実行状態レジスタの遷移フィルタ（負方向変化）
範囲	0～65535
分解能	1
初期値	0

詳細

:STATus:OPERation:NTRansition と同様です。

:STATus:OPERation:MEASure:PTRansition

Measurement Operation Status Positive Transition Filter

機能

測定実行状態レジスタの遷移フィルタ（正方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:MEASure:PTRansition <mosr_ptr>
```

クエリ

```
:STATus:OPERation:MEASure:PTRansition?
```

レスポンス

<mosr_ptr>

レジスタのビット割り当ては，:STATus:OPERation:MEASure[:EVENt]の説明を参照してください。

パラメータ

<mosr_ptr>	測定実行状態レジスタの遷移フィルタ（正方向変化）
範囲	0～65535
分解能	1
初期値	65535

詳細

:STATus:OPERation:PTRansition と同様です。

:STATus:OPERation:NTRansition

Operation Status Negative Transition Filter

機能

実行状態レジスタの遷移フィルタ（負方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:NTRansition <osr_ntf>
```

クエリ

```
:STATus:OPERation:NTRansition?
```

レスポンス

<osr_ntf>
レジスタのビット割り当ては，:STATus:OPERation[:EVENt]の説明を参照してください。

パラメータ

<osr_ntf>	実行状態レジスタの遷移フィルタ（負方向変化）
範囲	0～65535
分解能	1
初期値	0

詳細

実行状態コンディションレジスタの値が 1→0 に変化したことを検出するビットを 1 に設定します。

使用例

実行状態コンディションレジスタのビット 8 の値が 1 から 0 への変化を実行状態イベントレジスタに出力します。
このとき，遷移フィルタ（負方向変化）には 2^8=256 を設定します。
:STAT:OPER:NTR 256
実行状態レジスタの遷移フィルタ（負方向変化）の値を問い合わせます。
:STAT:OPER:NTR?
>256

:STATus:OPERation:PTRansition

Operation Status Positive Transition Filter

機能

実行状態レジスタの遷移フィルタ（正方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:OPERation:PTRansition <osr_ptf>
```

クエリ

```
:STATus:OPERation:PTRansition?
```

レスポンス

<osr_ptf>

レジスタのビット割り当ては，:STATus:OPERation[:EVENt]の説明を参照してください。

パラメータ

<osr_ptf>	実行状態レジスタの遷移フィルタ（正方向変化）
範囲	0～65535
分解能	1
初期値	65535

詳細

実行状態コンディションレジスタの値が 0→1 に変化したことを検出するビットを 1 に設定します。

使用例

実行状態コンディションレジスタのビット 9 の値が 0 から 1 に変化したことを，実行状態イベントレジスタに出力します。

このとき，遷移フィルタ（正方向変化）には 2^9=512 を設定します。

:STAT:OPER:PTR 512

実行状態レジスタの遷移フィルタ（正方向変化）の値を問い合わせます。

:STAT:OPER:PTR?

>512

:STATus:PRESet

Preset Status

機能

各種イネーブルレジスタ，各種レジスタの遷移フィルタ（正方向変化，負方向変化）を初期化します。

コマンド

```
:STATus:PRESet
```

詳細

実行状態レジスタ，クエスチョナブルステータスレジスタ，測定実行状態レジスタ，測定クエスチョナブルステータスレジスタ，信号発生器実行状態レジスタ，信号発生器クエスチョナブルステータスレジスタの，各種イネーブルレジスタ，遷移フィルタ（正方向変化，負方向変化）を初期化します。
各種イネーブルレジスタの設定値は 0 になります。
各種遷移フィルタ（正方向変化）の設定値は 65535 になります。
各種遷移フィルタ（負方向変化）の設定値は 0 になります。
各種イベントステータスレジスタ，各種コンディションレジスタは初期化しません。
ステータスバイトレジスタ，標準イベントステータスレジスタは初期化しません。

使用例

:STAT:PRES

:STATus:QUEStionable:CONDition?

Questionable Status Condition Register Query

機能

クエスチョナブルステータスコンディションレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:QUEStionable:CONDition?
```

レスポンス

<qscr>

レジスタのビット割り当ては，:STATus:QUEStionable[:EVENt]の説明を参照してください。

パラメータ

<qscr>	クエスチョナブルステータスコンディションレジスタ
範囲	0～65535

詳細

:STATus:OPERation:CONDition と同様です。

:STATus:QUEStionable:ENABle

Questionable Status Enable Register

機能

クエスチョナブルステータスイネーブルレジスタの値を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:ENABle <qser>
```

クエリ

```
:STATus:QUEStionable:ENABle?
```

レスポンス

<qser>

レジスタのビット割り当ては，:STATus:QUEStionable[:EVENt]の説明を参照してください。

パラメータ

<qser>	クエスチョナブルステータスイネーブルレジスタ
範囲	0～65535
分解能	1
初期値	0

詳細

ステータスバイトレジスタに検出結果を出力するビットを設定します。

:STATus:QUEStionable[:EVENt]?

Questionable Status Register Query

機能

クエスチョナブルステータスレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:QUEStionable[:EVENt]?
```

レスポンス

<qsr>

値=bit0 + bit1 + ... + bit15

bit0 = 2^0 = 1	未使用
bit1 = 2^1 = 2	未使用
bit2 = 2^2 = 4	未使用
bit3 = 2^3 = 8	未使用
bit4 = 2^4 = 16	未使用
bit5 = 2^5 = 32	未使用
bit6 = 2^6 = 64	未使用
bit7 = 2^7 = 128	未使用
bit8 = 2^8 = 256	未使用
bit9 = 2^9 = 512	測定
bit10 = 2^{10} = 1024	信号発生器
bit11 = 2^{11} = 2048	未使用
bit12 = 2^{12} = 4096	未使用
bit13 = 2^{13} = 8192	未使用
bit14 = 2^{14} = 16384	未使用
bit15 = 2^{15} = 32768	未使用

パラメータ

<qsr>	クエスチョナブルステータスレジスタ
範囲	0～65535

詳細

クエスチョナブルステータスレジスタの bit0, 1～15 に対応する値 2^0 = 1, 2^1 = 2～2^{15} = 32768 の中から，発生したイベントのビットの値の総和がレスポンスとなります。

使用例

クエスチョナブルステータスレジスタの内容を問い合わせます。

```
:STAT:QUES?
> 512
```

:STATus:QUEStionable:GENerator:CONDition?

Generator Questionable Status Condition Register Query

機能

信号発生器クエスチョナブルステータスコンディションレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:QUEStionable:GENerator:CONDition?
```

レスポンス

<gqscr>
レジスタのビット割り当ては，:STATus:QUEStionable:GENerator[:EVENt]の説明を参照してください。

パラメータ

<gqscr>	信号発生器クエスチョナブルステータスコンディションレジスタ
範囲	0～65535

詳細

:STATus:OPERation:CONDition と同様です。

:STATus:QUEStionable:GENerator:ENABle

Generator Questionable Status Enable Register

機能

信号発生器クエスチョナブルステータスイネーブルレジスタの値を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:GENerator:ENABle <gqser>
```

クエリ

```
:STATus:QUEStionable:GENerator:ENABle?
```

レスポンス

<gqser>

レジスタのビット割り当ては，:STATus:QUEStionable:GENerator[:EVENt]の説明を参照してください。

パラメータ

<gqser>	信号発生器クエスチョナブルステータスイネーブルレジスタ
範囲	0～65535
分解能	1
初期値	0

詳細

:STATus:OPERation:ENABle と同様です。

:STATus:QUEStionable:GENerator[:EVENt]?

Generator Questionable Status Register Query

機能

信号発生器クエスチョナブルステータスレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:QUEStionable:GENerator[:EVENt]?
```

レスポンス

<gqsr>　値=bit0 + bit1 + ... + bit15

bit0 = 2^0 = 1	ロードしたファイルにエラーがあります。
bit1 = 2^1 = 2	未使用
bit2 = 2^2 = 4	未使用
bit3 = 2^3 = 8	未使用
bit4 = 2^4 = 16	未使用
bit5 = 2^5 = 32	未使用
bit6 = 2^6 = 64	未使用
bit7 = 2^7 = 128	未使用
bit8 = 2^8 = 256	未使用
bit9 = 2^9 = 512	未使用
bit10 = 2^{10} = 1024	未使用
bit11 = 2^{11} = 2048	未使用
bit12 = 2^{12} = 4096	未使用
bit13 = 2^{13} = 8192	未使用
bit14 = 2^{14} = 16384	未使用
bit15 = 2^{15} = 32768	未使用

パラメータ

<gqsr>	信号発生器クエスチョナブルステータスレジスタ
範囲	0～65535

詳細

信号発生器クエスチョナブルステータスレジスタの bit0，1～15 に対応する値 2^0 = 1，2^1 = 2～2^{15} = 32768 の中から，発生したイベントのビットの値の総和がレスポンスとなります。

使用例

信号発生器クエスチョナブルステータスレジスタの内容を問い合わせます。

```
:STAT:QUES:GEN?
> 1
```

:STATus:QUEStionable:GENerator:NTRansition

Generator Questionable Status Negative Transition Filter

機能

信号発生器クエスチョナブルステータスレジスタの遷移フィルタ（負方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:GENerator:NTRansition <gqsr_ntf>
```

クエリ

```
:STATus:QUEStionable:GENerator:NTRansition?
```

レスポンス

<gqsr_ntf>
レジスタのビット割り当ては，:STATus:QUEStionable:GENerator[:EVENt]の説明を参照してください。

パラメータ

<gqsr_ntf>	信号発生器クエスチョナブルステータスレジスタの遷移フィルタ（負方向変化）
範囲	0～65535
分解能	1
初期値	0

詳細

:STATus:OPERation:NTRansition と同様です。

:STATus:QUEStionable:GENerator:PTRansition

Generator Questionable Status Positive Transition Filter

機能

信号発生器クエスチョナブルステータスレジスタの遷移フィルタ（正方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:GENerator:PTRansition <gqsr_ptf>
```

クエリ

```
:STATus:QUEStionable:GENerator:PTRansition?
```

レスポンス

<gqsr_ptf>
レジスタのビット割り当ては，:STATus:QUEStionable:GENerator[:EVENt]の説明を参照してください。

パラメータ

<gqsr_ptf>	信号発生器クエスチョナブルステータスレジスタの遷移フィルタ（正方向変化）
範囲	0～65535
分解能	1
初期値	65535

詳細

:STATus:OPERation:PTRansition と同様です。

:STATus:QUEStionable:MEASure:CONDition?

Measurement Questionable Status Condition Register Query

機能

測定クエスチョナブルステータスコンディションレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:QUEStionable:MEASure:CONDition?
```

レスポンス

<mqscr>
レジスタのビット割り当ては，:STATus:QUEStionable:MEASure[:EVENt]の説明を参照してください。

パラメータ

<mqscr>	測定クエスチョナブルステータスコンディションレジスタ
範囲	0～65535

詳細

:STATus:OPERation:CONDition と同様です。

:STATus:QUEStionable:MEASure:ENABle

Measurement Questionable Status Enable Register

機能

測定クエスチョナブルステータスイネーブルレジスタの値を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:MEASure:ENABle <mqser>
```

クエリ

```
:STATus:QUEStionable:MEASure:ENABle?
```

レスポンス

<mqser>

レジスタのビット割り当ては，:STATus:QUEStionable:MEASure[:EVENt]の説明を参照してください。

パラメータ

<mqser>	測定クエスチョナブルステータスイネーブルレジスタ
範囲	0～65535
分解能	1
初期値	0

詳細

:STATus:OPERation:ENABleと同様です。

:STATus:QUEStionable:MEASure[:EVENt]?

Measurement Questionable Status Register Query

機能

測定クエスチョナブルステータスレジスタの内容を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
:STATus:QUEStionable:MEASure[:EVENt]?
```

レスポンス

<mqsr>
値=bit0 + bit1 + ... + bit15

bit0 = 2^0 = 1	レベルオーバー
bit1 = 2^1 = 2	レベルアンダー
bit2 = 2^2 = 4	タイムアウト
bit3 = 2^3 = 8	未使用
bit4 = 2^4 = 16	未使用
bit5 = 2^5 = 32	未使用
bit6 = 2^6 = 64	未使用
bit7 = 2^7 = 128	未使用
bit8 = 2^8 = 256	未使用
bit9 = 2^9 = 512	未使用
bit10 = 2^{10} = 1024	未使用
bit11 = 2^{11} = 2048	未使用
bit12 = 2^{12} = 4096	未使用
bit13 = 2^{13} = 8192	未使用
bit14 = 2^{14} = 16384	未使用
bit15 = 2^{15} = 32768	未使用

ビットの割り当ては，アプリケーションによって変わります。
アプリケーションソフトウェアの取扱説明書を参照してください。

パラメータ

<mqsr>	測定クエスチョナブルステータスレジスタ
範囲	0～65535

詳細

測定クエスチョナブルステータスレジスタの bit0，1～15 に対応する値 2^0 = 1，2^1 = 2～2^{15}= 32768 の中から，発生したイベントのビットの値の総和がレスポンスとなります。

使用例

測定クエスチョナブルステータスレジスタの内容を問い合わせます。

```
:STAT:QUES:MEAS?
> 1
```

:STATus:QUEStionable:MEASure:NTRansition

Measurement Questionable Status Negative Transition Filter

機能

測定クエスチョナブルステータスレジスタの遷移フィルタ（負方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:MEASure:NTRansition <mqsr_ntr>
```

クエリ

```
:STATus:QUEStionable:MEASure:NTRansition?
```

レスポンス

<mqsr_ntr>

レジスタのビット割り当ては，:STATus:QUEStionable:MEASure[:EVENt]の説明を参照してください。

パラメータ

<mqsr_ntr>	測定クエスチョナブルステータスレジスタの遷移フィルタ（負方向変化）
範囲	0～65535
分解能	1
初期値	0

詳細

:STATus:OPERation:NTRansition と同様です。

:STATus:QUEStionable:MEASure:PTRansition

Measurement Questionable Status Positive Transition Filter

機能

測定クエスチョナブルステータスレジスタの遷移フィルタ（正方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:MEASure:PTRansition <mqsr_ptr>
```

クエリ

```
:STATus:QUEStionable:MEASure:PTRansition?
```

レスポンス

<mqsr_ptr>

レジスタのビット割り当ては，:STATus:QUEStionable:MEASure[:EVENt]の説明を参照してください。

パラメータ

<mqsr_ptr>	測定クエスチョナブルステータスレジスタの遷移フィルタ（正方向変化）
範囲	0～65535
分解能	1
初期値	65535

詳細

:STATus:OPERation:PTRansition と同様です。

:STATus:QUEStionable:NTRansition

Questionable Status Negative Transition Filter

機能

クエスチョナブルステータスレジスタの遷移フィルタ（負方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:NTRansition <qsr_ntf>
```

クエリ

```
:STATus:QUEStionable:NTRansition?
```

レスポンス

<qsr_ntf>

レジスタのビット割り当ては，:STATus:QUEStionable[:EVENt]の説明を参照してください。

パラメータ

<qsr_ntf>		クエスチョナブルステータスレジスタの遷移フィルタ（負方向変化）
	範囲	0～65535
	分解能	1
	初期値	0

詳細

:STATus:OPERation:NTRansition と同様です。

:STATus:QUEStionable:PTRansition

Questionable Status Positive Transition Filter

機能

クエスチョナブルステータスレジスタの遷移フィルタ（正方向変化）を設定，または問い合わせます。

コマンド

```
:STATus:QUEStionable:PTRansition <qsr_ptf>
```

クエリ

```
:STATus:QUEStionable:PTRansition?
```

レスポンス

<qsr_ptf>

レジスタのビット割り当ては，:STATus:QUEStionable[:EVENt]の説明を参照してください。

パラメータ

<qsr_ptf>	クエスチョナブルステータスレジスタの遷移フィルタ（正方向変化）	
範囲	0～65535	
分解能	1	
初期値	65535	

詳細

:STATus:OPERation:PTRansition と同様です。

:SYSTem:BASE:LICense:CHECk?

License Key Check

機能

ライセンスのチェックを行い，正常であればインストール準備を行います。

クエリ

```
:SYSTem:BASE:LICense:CHECk? <filename>
```

レスポンス

<result>

パラメータ

<filename>	ライセンスファイル名
<result>	ライセンスファイルのチェック結果
–1	同じ本体の別モジュールで実行したためインストールできない。
0	正常
1	指定したファイルが見つからない。
2	ファイルが違う。
3	不正なファイル　または　別の MT8870 用のファイル
4	すでにライセンスが存在する。
5	ライセンス期間外
6	書き込めるライセンス数の上限に達している。

詳細

事前に FTP によりソフトウェアライセンスファイルを指定フォルダに転送する必要があります。転送先フォルダは，"/install/license/" です。
:SYSTem:BASE:LICense:INSTall コマンドによりインストールする前に，再起動したり電源を切ったりするとインストール準備は無効になります。

使用例

```
:SYSTem:BASE:LICense:CHECk? "Licenses.xml"
>0
:SYSTem:BASE:LICense:INSTall
```

:SYSTem:BASE:LICense:INFormation?

License Information

機能

ライセンスの期限情報を問い合わせます。

クエリ

```
:SYSTem:BASE:LICense:INFormation? <model>
```

レスポンス

<kind>,<days>

パラメータ

<model>	ライセンス形名
<kind>	ライセンスの種類
0	Normal ライセンス
1	期限付きライセンス
2	トライアルライセンス
<days>	残り日数
1 以上	ライセンスの残り日数 1 の場合は, ライセンス最終日です。
0	ライセンスの期限切れ
–1	期限無し (Normal ライセンスの場合)

詳細

ライセンス形名には, ライセンスの期限を問い合わせる製品の形名 (オプションを含む) を指定します。
例: MX887011A,MX887012A

使用例

:SYST:BASE:LIC:INF? "MX887012A"
> 0,-1

:SYSTem:BASE:LICense:INSTall

License Key Install

機能

ライセンスをインストールします。

コマンド

```
:SYSTem:BASE:LICense:INSTall
```

詳細

事前に:SYSTem:BASE:LICense:CHECk?コマンドでインストール準備をする必要があります。
本コマンド実行後は，システム再起動により全モジュールのリモートの接続が切断されます。

使用例

:SYSTem:BASE:LICense:CHECk? "Licenses.xml"
> 0
:SYSTem:BASE:LICense:INSTall

備考

ハードウェアオプションライセンスは，モジュールに保持されます。オプションの違いにより，同一 MT8870A 内の特定モジュールだけアプリケーションが起動しない場合があります。
ソフトウェアライセンス，ソフトウェアオプションライセンス，波形ライセンスは，MT8870A に保持されます。モジュールを他の MT8870A に移動した場合，ライセンスの違いによりアプリケーションが起動しない場合があります。

:SYSTem:BASE:REFerence:FREQuency

Frequency Reference

機能

周波数基準信号源を設定，または問い合わせます。

コマンド

```
:SYSTem:BASE:REFerence:FREQuency <source>
```

クエリ

```
:SYSTem:BASE:REFerence:FREQuency?
```

レスポンス

<source>

パラメータ

<source>		内部基準信号/外部基準信号
	INT	MT8870A の内部基準信号
	AUTO	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
	初期値	AUTO

詳細

MT8870A に装着されている全モジュールに基準信号源を設定します。モジュールごとに基準信号源を設定できません。

使用例

```
:SYST:BASE:REF:FREQ INT
:SYST:BASE:REF:FREQ?
> INT
```

関連コマンド

:SYSTem:BASE:REFerence:FREQuency:CONFigure

備考

*RST コマンドで初期化されません。

:SYSTem:BASE:REFerence:FREQuency:ADJust

Adjust Reference Clock

機能

基準周波数補正値を設定，または問い合わせます。

コマンド

```
:SYSTem:BASE:REFerence:FREQuency:ADJust <val>
```

クエリ

```
:SYSTem:BASE:REFerence:FREQuency:ADJust?
```

レスポンス

<val>

パラメータ

<val>	調整値
範囲	0～1023
分解能	1
初期値	512

詳細

MT8870A に装着されている全モジュールで共通の設定です。

使用例

:SYST:BASE:REF:FREQ:ADJ 512
:SYST:BASE:REF:FREQ:ADJ?
> 512

備考

*RST コマンドで初期化されません。

:SYSTem:BASE:REFerence:FREQuency:ADJust:DEFault?

Adjust Reference Clock Default

機能

工場出荷時の基準周波数補正値を問い合わせます。

クエリ

```
:SYSTem:BASE:REFerence:FREQuency:ADJust:DEFault?
```

レスポンス

<val>

パラメータ

<val>		工場出荷時の基準周波数補正値
	範囲	0～1023
	分解能	1
	初期値	512

詳細

MT8870A に装着されている全モジュールで共通の値です。

使用例

```
:SYST:BASE:REF:FREQ:ADJ:DEF?
> 512
```

:SYSTem:BASE:REFerence:FREQuency:CONFigure

Frequency Reference

機能

周波数基準信号源を設定，または問い合わせます。

コマンド

```
:SYSTem:BASE:REFerence:FREQuency:CONFigure <source>
```

クエリ

```
:SYSTem:BASE:REFerence:FREQuency:CONFigure?
```

レスポンス

<source_res>

パラメータ

<source>	内部基準信号／外部基準信号
10MHZINT	MT8870A の内部基準信号
10MHZEXT	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
13MHZEXT	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
AUTO	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
<source_res>	内部基準信号／外部基準信号
10MHZINT	MT8870A の内部基準信号
AUTO	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
初期値	AUTO

詳細

MT8870A に装着されている全モジュールに基準信号源を設定します。モジュールごとに基準信号源を設定できません。

使用例

```
:SYSTem:BASE:REFerence:FREQuency:CONFigure 10MHZINT
:SYSTem:BASE:REFerence:FREQuency:CONFigure?
> 10MHZINT
```

関連コマンド

:SYSTem:BASE:REFerence:FREQuency

備考

*RST コマンドで初期化されません。

:SYSTem:BASE:REFerence:FREQuency:CONFigure:SOURce?

Reference Signal Query

機能

周波数基準信号源の状態を問い合わせます。

クエリ

```
:SYSTem:BASE:REFerence:FREQuency:CONFigure:SOURce?
```

レスポンス

<source>

パラメータ

<source>	基準信号の状態
INT	内部基準信号源でロックしています。
VALI	外部基準信号源でロックしています。
INVALID	外部基準信号源でアンロック状態です

詳細

すべてのスロットで共通です。

使用例

:SYSTem:BASE:REFerence:FREQuency:CONFigure:SOURce?
> INT

関連コマンド

:SYSTem:BASE:REFerence:FREQuency:SOURce

:SYSTem:BASE:REFerence:FREQuency:SOURce?

Reference Signal Query

機能

周波数基準信号源の状態を問い合わせます。

クエリ

```
:SYSTem:BASE:REFerence:FREQuency:SOURce?
```

レスポンス

<source>

パラメータ

<source>	基準信号の状態
INT	内部基準信号源でロックしています。
EXT	外部基準信号源でロックしています。
EXTU	外部基準信号源でアンロック状態です。

詳細

すべてのスロットで共通です。

関連コマンド

:SYSTem:BASE:REFerence:FREQuency:CONFigure:SOURce

使用例

```
:SYST:BASE:REF:FREQ:SOUR?
> EXT
```

:SYSTem:BASE:UPDate:FIRMware

Firmware Update

機能

ファームウェアパッケージのアップデートを実行，またはアップデート結果を問い合わせます。

Command

```
:SYSTem:BASE:UPDate:FIRMware <target>[,<option>]
```

クエリ

```
:SYSTem:BASE:UPDate:FIRMware?
```

レスポンス

<result>

パラメータ

<target>	アップデート先モジュール
SINGLE	単体モジュール
ALL	MT8870A に装着されている全モジュール
<option>	オプション
DIFF	バージョンが違う場合だけアップデートします。
FORCE	強制的にアップデートします。
<result>	最後にインストールしたファームウェアの更新結果
0	正常
1	更新失敗
2	異常発生

詳細

事前に:SYSTem:BASE:UPDate:FIRMware:CHECk?コマンドで更新準備をする必要があります。
本コマンドを実行すると，システムが再起動します。再起動中は全モジュールがリモート制御できなくなります。
MT8870A 内の全モジュールをアップデートするか，コマンド送信先の単体モジュールをアップデートするかを選択してください。
:SYSTem:BASE:UPDate:FIRMware?で最後にインストールしたファームウェアパッケージのアップデート結果を取得できます。

使用例

```
:SYSTem:BASE:UPDate:FIRMware:CHECk? "package.inst"
> 0
:SYSTem:BASE:UPDate:FIRMware ALL,DIFF
```

:SYSTem:BASE:UPDate:FIRMware:CHECk?

Check Firmware Update Package

機能

ファームウェアパッケージのチェックを行い，正常であれば更新準備を行います。

クエリ

```
:SYSTem:BASE:UPDate:FIRMware:CHECk? <filename>
```

レスポンス

<result>

パラメータ

<filename>	ファームウェアパッケージファイル名
<result>	ファームウェアパッケージファイルのチェック結果
0	正常
1	指定したファイルが見つからない
2	ファイルが違う
3	ファイルが壊れているまたは本器用のインストーラではない
4	ファイルは正常だがインストールできないバージョンである
5	ファイルは正常だが，TRX 2 では更新準備を行うことができない。

詳細

事前に FTP によりファームウェアパッケージファイルを指定フォルダに転送する必要があります。転送先フォルダは，"/install/package/" です。

:SYSTem:BASE:UPDate:FIRMware コマンドによりファームウェアをアップデートする前に，再起動したり電源を切ったりした場合は，更新準備は無効になります。

MU887002Aは，V3.00.00以降のファームウェアパッケージでなければインストールできません。
MU887002A-007 搭載時は，V4.00.00 以降のファームウェアパッケージでなければインストールできません。

使用例

```
:SYSTem:BASE:UPDate:FIRMware:CHECk? "package.inst"
> 0
:SYSTem:BASE:UPDate:FIRMware ALL,DIFF
```

:SYSTem:BEEPer

Beep

機能

ブザーのオン/オフを設定，または問い合わせます。

コマンド

```
:SYSTem:BEEPer <sw>
```

クエリ

```
:SYSTem:BEEPer?
```

レスポンス

<sw>

パラメータ

<sw>	ブザーのオン/オフ
ON	ブザーを鳴らします。
OFF	ブザーを鳴らしません。

詳細

モジュールごとに設定できます。
オンに設定すると，次の場合にブザーが鳴ります。
「ピ」は短い音，「ピー」は長い音を表します。

- 電源投入時の自己診断で異常がある場合：「ピー」
- 電源投入時のソフトウェア起動時：「ピ，ピ，ピ」
- 電源切断時，または:SYSTem:REBoot，SYSTem:SHUTdown コマンド送信時：「ピ，ピ」
- 送信したコマンドにエラーがある場合：「ピ」

使用例

```
:SYST:BEEP ON
:SYST:BEEP?
> ON
```

:SYSTem:COMMunicate:GPIB:ADDR

GPIB Address

機能

GPIB アドレスを設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:GPIB:ADDR <gp_address>
```

クエリ

```
:SYSTem:COMMunicate:GPIB:ADDR?
```

レスポンス

<gp_address>

パラメータ

<gp_address>	GPIB アドレス	
範囲	1～30	
初期値	スロット 1	1
	スロット 2	1
	スロット 3	1
	スロット 4	1

詳細

MU887002A では設定はできますが，GPIB を使用できません。
モジュールの TRX ごとに GPIB アドレス設定できます。
GPIB アドレスは，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの GPIB アドレスが適用されます。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，GPIB アドレスは初期値に変更されます。

使用例

```
:SYST:COMM:GPIB:ADDR 8
:SYST:COMM:NET:REST

:SYST:COMM:GPIB:ADDR?
> 8
```

:SYSTem:COMMunicate:GPIB:ADDR:ALL

GPIB Address

機能

全モジュールの TRX の GPIB アドレスを設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:GPIB:ADDR:ALL
<address1>,<address2>,<address3>,<address4>
```

クエリ

```
:SYSTem:COMMunicate:GPIB:ADDR:ALL?
```

レスポンス

<address1,address2,address3,address4>

パラメータ

<address1>	スロット 1 の GPIB アドレス
範囲	1～30
初期値	1
<address2>	スロット 2 の GPIB アドレス
範囲	1～30
初期値	1
<address3>	スロット 3 の GPIB アドレス
範囲	1～30
初期値	1
<address4>	スロット 4 の GPIB アドレス
範囲	1～30
初期値	1

詳細

MU887002A では設定はできますが，GPIB を使用できません。
全モジュールの TRX の GPIB アドレスを設定できます。
GPIB アドレスは，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの GPIB アドレスが適用されます。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，GPIB アドレスは初期値に変更されます。

使用例

```
:SYST:COMM:GPIB:ADDR:ALL 1,2,3,4
:SYST:COMM:NET:REST

:SYST:COMM:GPIB:ADDR:ALL?
> 1,2,3,4
```

:SYSTem:COMMunicate:GPIB[:SELF]:DELimiter

Delimiter

機能

デリミタ（メッセージ末尾に付加される文字）を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:GPIB[:SELF]:DELimiter <code>
```

クエリ

```
:SYSTem:COMMunicate:GPIB[:SELF]:DELimiter?
```

レスポンス

<code>

パラメータ

<code>	終端コード
LF	LF (Line Feed)
CRLF	CR/LF (Carriage Return and Line Feed)
NONE	None(EOI のみ)
初期値	CRLF

詳細

モジュールごとに設定できます。

MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，終端コードは初期値に変更されます。

使用例

:SYST:COMM:GPIB:DEL CRLF
:SYST:COMM:GPIB:DEL?
>CRLF

備考

:SYSTem:COMMunicate:GPIB[:SELF]:TERMinator と同じ処理をします。

*RST，:SYSTem:REBoot，:SYSTem:COMMunicate:NET:RESTart コマンドで初期化されません。

イーサネット接続でパラメータを NONE に設定した場合，本器からのレスポンスメッセージはデリミタが CRLF に設定されています。

:SYSTem:COMMunicate:GPIB[:SELF]:TERMinator

Terminator

機能

ターミネータ（メッセージ末尾に付加される文字）を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:GPIB[:SELF]:TERMinator <code>
```

クエリ

```
:SYSTem:COMMunicate:GPIB[:SELF]:TERMinator?
```

レスポンス

<code>

パラメータ

<code>	終端コード
LF	LF (Line Feed)
CRLF	CR/LF (Carriage Return and Line Feed)
NONE	None (EOI のみ)
初期値	CRLF

詳細

モジュールごとに設定できます。

MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，終端コードは初期値に変更されます。

使用例

:SYST:COMM:GPIB:TERM CRLF
:SYST:COMM:GPIB:TERM?
> CRLF

備考

:SYSTem:COMMunicate:GPIB[:SELF]:DELimiter と同じ処理をします。

*RST，:SYSTem:REBoot，:SYSTem:COMMunicate:NET:RESTart コマンドで初期化されません。

イーサネット接続でパラメータを NONE に設定した場合，本器からのレスポンスメッセージはターミネータが CRLF に設定されています。

:SYSTem:COMMunicate:GTL

Go to Local

機能

リモート接続を切断します。

コマンド

```
:SYSTem:COMMunicate:GTL
```

詳細

モジュールパネルの Remote ランプが消灯します。

使用例

:SYST:COMM:GTL

:SYSTem:COMMunicate:NET:DNS:DOMain

Domain Name

機能

ドメイン名を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:DNS:DOMain <name>
```

クエリ

```
:SYSTem:COMMunicate:NET:DNS:DOMain?
```

レスポンス

<name>

パラメータ

<name>	ドメイン名	
	最大	63 バイト
	初期値	"DOMAIN"

設定時に以下の命名規則に適合していない場合はパラメータ・エラーとなります。

- 最初の文字は英数字であること。
- 途中の文字は英数字またはハイフンであること。
- 最後の文字は英数字であること。
- 文字を""で囲っていること。

詳細

MT8870A に装着されている全モジュールにドメイン名を設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，ドメイン名は初期値に変更されます。

使用例

```
:SYST:COMM:NET:DNS:DOM "Domain"
:SYST:COMM:NET:REST

:SYST:COMM:NET:DNS:DOM?
> Domain
```

:SYSTem:COMMunicate:NET:DNS:HOST

Host Name

機能

ホスト名を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:DNS:HOST <host>
```

クエリ

```
:SYSTem:COMMunicate:NET:DNS:HOST?
```

レスポンス

<host>

パラメータ

<host>　ホスト名

最小	1 バイト
最大	61 バイト
初期値	"HOST"

設定時に以下の命名規則に適合していない場合はパラメータ・エラーとなります。

- 最初の文字は英数字であること。
- 途中の文字は英数字またはハイフンであること。
- 最後の文字は英数字であること。
- 文字を""で囲っていること。

詳細

設定したホスト名は，スロット番号が後ろに追加され，HOSTNAME"-"スロット番号となります。
MT8870A に装着されている全モジュールにホスト名を設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，ホスト名は初期値に変更されます。

使用例

```
:SYST:COMM:NET:DNS:HOST "ANRITSU-PC"
:SYST:COMM:NET:REST

:SYST:COMM:NET:DNS:HOST?
> ANRITSU-PC-1
```

:SYSTem:COMMunicate:NET:FTP:USER:CATalog?

FTP User List

機能

すべての FTP ユーザ名を問い合わせます。

クエリ

```
:SYSTem:COMMunicate:NET:FTP:USER:CATalog?
```

レスポンス

<name>,<name>,.....

パラメータ

<name>	ユーザ名

使用例

:SYST:COMM:NET:FTP:USER:CAT?
> user1,user2,user3

:SYSTem:COMMunicate:NET:FTP:USER:CHANge:NAME

Change FTP User Name

機能

FTP ユーザ名を変更します。

コマンド

```
:SYSTem:COMMunicate:NET:FTP:USER:CHANge:NAME <before>,<after>
```

パラメータ

<before>	変更前のユーザ名
<after>	変更後のユーザ名

詳細

ユーザ名に使用できる文字と文字数は以下のとおりです。
文字数は 1～31 文字までです。""で囲って設定します。
先頭文字は英字[大・小]とドット（.）とアンダースコア（_）です。
2 文字目以降は英数字[大・小]とドット（.）とアンダースコア（_）です。
すべて数字は不可とします。

使用例

user1 を user2 に変更します。
:SYST:COMM:NET:FTP:USER:CHAN:NAME "user1","user2"

:SYSTem:COMMunicate:NET:FTP:USER:CHANge:PASS

Change FTP User Password

機能

FTP ユーザのパスワードを変更します。

コマンド

```
:SYSTem:COMMunicate:NET:FTP:USER:CHANge:PASS <name>,<before>,<after>
```

パラメータ

<name>	パスワードを変更するユーザ名
<before>	変更前のパスワード
<after>	変更後のパスワード

詳細

パスワードに使用できる文字と文字数は以下のとおりです。
文字数は 1～16 文字までです。""で囲って設定します。
使用可能文字は英数字[大・小]と記号（! #$%'()-=^~@[{;+]},.?_）です。

使用例

:SYST:COMM:NET:FTP:USER:CHAN:PASS "user1","abcde","fghij"

:SYSTem:COMMunicate:NET:HWADdress?

MAC Address

機能

MAC アドレスを問い合わせます。

クエリ

```
:SYSTem:COMMunicate:NET:HWADdress?
```

レスポンス

<macaddr>

パラメータ

<macaddr>	MAC アドレス

詳細

MU887002A の MAC アドレスを問い合わせます。

使用例

:SYST:COMM:NET:HWAD?
> 00-00-00-00-00-00

:SYSTem:COMMunicate:NET:IPVersion

IP Version

機能

IPv4, IPv6 のオン/オフを設定, または問い合わせます。
オンに設定すると, その IP Version でモジュールと通信できます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVersion <ipver>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVersion?
```

レスポンス

<ipver>

パラメータ

<ipver>	IP Version
V4	IPv4 のみオンにします。
V6	IPv6 のみオンにします。
V4V6	IPv4, IPv6 ともにオンにします。
初期値	V4

詳細

MT8870A に装着されているすべてのモジュールに IP バージョンのオン/オフを設定します。
本コマンドの設定を有効にするのは, :SYSTem:COMMunicate:NET:RESTart コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると, IP Version は初期値に変更されます。

使用例

:SYST:COMM:NET:IPV V4V6
:SYST:COMM:NET:REST

再起動後
:SYST:COMM:NET:IPV?
> V4V6

備考

*RST, :SYSTem:REBoot コマンドで初期化されません。

:SYSTem:COMMunicate:NET:IPVFour:ADDRess:TYPE

IPv4 Type

機能

IPv4 アドレスの種別を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVFour:ADDRess:TYPE <ipv4type>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:ADDRess:TYPE?
```

レスポンス

<ipv4type>

パラメータ

<ipv4type>	IPv4 アドレスの種別
STATIC	固定 IP です。
DHCP	DHCP サーバから取得して設定します。
初期値	STATIC

詳細

MT8870A の全スロットに IPv4 アドレスの種別を設定します。
STATIC を設定した場合は，次のコマンドでアドレス，サブネットマスク，デフォルトゲートウェイを設定します。

:SYSTem:COMMunicate:NET:IPVFour:STATic:IPADdress
:SYSTem:COMMunicate:NET:IPVFour:STATic:SMASk
:SYSTem:COMMunicate:NET:IPVFour:STATic:GIP

本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart コマンドを送信します。
IPv4 アドレスの種別は，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの IPv4 アドレスの種別が適用されます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，IPv4 アドレスの種別は初期値に変更されます。

使用例

:SYST:COMM:NET:IPVF:ADDR:TYPE DHCP
:SYST:COMM:NET:REST

再起動後
:SYST:COMM:NET:IPVF:ADDR:TYPE TYPE?
> DHCP

:SYSTem:COMMunicate:NET:IPVFour:CURRent:GIP?

Get Current IPv4 Default Gateway

機能

現在のデフォルト・ゲートウェイ・アドレスを問い合わせます。

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:CURRent:GIP?
```

レスポンス

<gateway>

パラメータ

<gateway>	デフォルト・ゲートウェイ・アドレス

詳細

通信先モジュールのデフォルト・ゲートウェイ・アドレスを取得します。
モジュールの IPv4 アドレスがオフに設定されている場合, レスポンスは 0 になります。

使用例

```
:SYST:COMM:NET:IPVF:CURR:GIP?
> 192.168.0.1
```

:SYSTem:COMMunicate:NET:IPVFour:CURRent:IPADdress?

Get Current IPv4 Address

機能

モジュールの IPv4 アドレスを問い合わせます。

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:CURRent:IPADdress?
```

レスポンス

<num>	設定された IP アドレスの個数
<ipaddr>	IP アドレス

パラメータ

<num>	設定された IP アドレスの個数
<ipaddr>	IP アドレス

詳細

モジュールの IPv4 アドレスを取得します。
モジュールの IPv4 アドレスがオフに設定されている場合, レスポンスは 0 になります。

使用例

```
:SYST:COMM:NET:IPVF:CURR:IPAD?
> 2,192.168.0.1,172.16.0.31
```

:SYSTem:COMMunicate:NET:IPVFour:CURRent:SMASk?

Get Current IPv4 Subnet Mask

機能

現在のサブネットマスクを問い合わせます。

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:CURRent:SMASk?
```

レスポンス

<mask>　　サブネットマスク

詳細

通信先モジュールのサブネットマスクを取得します。
モジュールの IPv4 アドレスがオフに設定されている場合，レスポンスは 0 になります。

使用例

:SYST:COMM:NET:IPVF:CURR:SMAS?
> 255.255.255.0

:SYSTem:COMMunicate:NET:IPVFour:DNS:AUTO:ENABle

IPv4 DNS Server Auto

機能

DNS サーバのアドレスを自動的に取得するかどうかを設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVFour:DNS:AUTO:ENABle <switch>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:DNS:AUTO:ENABle?
```

レスポンス

<switch>

パラメータ

<switch>	DNS サーバのアドレス自動取得オンオフ設定
ON	オンにします。
OFF	オフにします。

詳細

MT8870A に装着されている全モジュールに設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS サーバアドレスの自動取得は OFF に変更されます。

使用例

```
:SYST:COMM:NET:IPVF:DNS:AUTO:ENAB ON
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVF:DNS:AUTO:ENAB?
> ON
```

:SYSTem:COMMunicate:NET:IPVFour:DNS:PRIMary

IPv4 DNS Primary Address

機能

DNS のプライマリ・アドレス（IPv4）を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVFour:DNS:PRIMary <ip4addr>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:DNS:PRIMary?
```

レスポンス

<ip4addr>

パラメータ

<ip4addr>	DNS プライマリ・アドレス xxx.xxx.xxx.xxx

詳細

MT8870A に装着されている全モジュールに DNS プライマリ・アドレスを設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS プライマリ・アドレスは 0 に変更されます。

使用例

:SYST:COMM:NET:IPVF:DNS:PRIM 192.168.10.1
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVF:DNS:PRIM?
> 192.168.10.1

:SYSTem:COMMunicate:NET:IPVFour:DNS:SECondary

IPv4 DNS Secondary Address

機能

DNS のセカンダリ・アドレス（IPv4）を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVFour:DNS:SECondary <ip4addr>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:DNS:SECondary?
```

レスポンス

<ip4addr>

パラメータ

<ip4addr>	DNS セカンダリ・アドレス xxx.xxx.xxx.xxx

詳細

MT8870A に装着されている全モジュールに DNS セカンダリ・アドレスを設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS セカンダリ・アドレスは 0 に変更されます。

使用例

```
:SYST:COMM:NET:IPVF:DNS:SEC 192.168.10.2
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVF:DNS:SEC?
>192.168.10.2
```

:SYSTem:COMMunicate:NET:IPVFour:STATic:GIP

IPv4 Default Gateway

機能

固定 IPv4 設定のデフォルト・ゲートウェイを設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVFour:STATic:GIP <gateway>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:STATic:GIP?
```

レスポンス

<gateway>

パラメータ

<gateway>	デフォルト・ゲートウェイ・アドレス xxx.xxx.xxx.xxx	

詳細

MT8870A に装着されている全モジュールにデフォルトゲートウェイを設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart コマンドを送信します。
パラメータに"" (空文字) を指定すると，デフォルトゲートウェイの設定を初期化できます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，デフォルトゲートウェイは 0 に変更されます。

使用例

```
:SYST:COMM:NET:IPVF:STAT:GIP "192.168.0.1"
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVF:STAT:GIP?
> 192.168.0.1
```

:SYSTem:COMMunicate:NET:IPVFour:STATic:IPADdress

IPv4 Address

機能

モジュールの固定 IPv4 アドレスを設定/問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVFour:STATic:IPADdress
<ipv4_addr1>,<ipv4_addr2>,<ipv4_addr3>,<ipv4_addr4>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:STATic:IPADdress?
```

レスポンス

<ipv4_addr1>,<ipv4_addr2>,<ipv4_addr3>,<ipv4_addr4>

パラメータ

<ipv4_addr1>	スロット 1 の IPv4 アドレス xxx.xxx.xxx.xxx
<ipv4_addr2>	スロット 2 の IPv4 アドレス xxx.xxx.xxx.xxx
<ipv4_addr3>	スロット 3 の IPv4 アドレス xxx.xxx.xxx.xxx
<ipv4_addr4>	スロット 4 の IPv4 アドレス xxx.xxx.xxx.xxx

詳細

MT8870A の各スロットの IPv4 アドレスを設定します。モジュールが挿入されていないスロットにも IPv4 アドレスを設定できます。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart コマンドを送信します。
IPv4 アドレスは，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの IPv4 アドレスが適用されます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，このコマンドの設定はクリアされます。

使用例

```
:SYST:COMM:NET:IPVF:STAT:IPAD "192.168.1.1","192.168.1.2","192.168.1.3","192.168.1.4"
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVF:STAT:IPAD?
> 192.168.1.1,192.168.1.2,192.168.1.3,192.168.1.4
```

:SYSTem:COMMunicate:NET:IPVFour:STATic:SMASk

IPv4 Subnet Mask

機能

固定 IPv4 アドレスのサブネットマスクを設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVFour:STATic:SMASk <mask>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVFour:STATic:SMASk?
```

レスポンス

<mask>

パラメータ

<mask>	サブネットマスク

詳細

MT8870A に装着されている全モジュールにサブネットマスクを設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，サブネットマスクは 255.255.255.0 に変更されます。

使用例

```
:SYST:COMM:NET:IPVF:STAT:SMAS "255.255.255.0"
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVF:STAT:SMAS?
> 255.255.255.0
```

:SYSTem:COMMunicate:NET:IPVSix:ADDRess:TYPE

IPv6 Type

機能

IPv6 アドレスの種別を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVSix:ADDRess:TYPE <ipv6type>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:ADDRess:TYPE?
```

レスポンス

<ipv6type>

パラメータ

<ipv6type>	IPv6 アドレスの種別
STATIC	固定 IP です。
RA_EUI64	EUI-64 (RFC4862) に基づく RA による自動割り当てです。
RA_RANDOM	プライバシ拡張 (RFC4941) に基づく RA による自動割り当てです。
DHCPV6	DHCP サーバからアドレスを取得して設定します。
初期値	STATIC

詳細

MT8870A の全スロットに IPv6 アドレスの種別を設定します。
STATIC を設定した場合は，次のコマンドでアドレスを設定します。
:SYSTem:COMMunicate:NET:IPVSix:STATic:IPADdress
:SYSTem:COMMunicate:NET:IPVSix:STATic:DEFRouter

本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
IPv6 アドレスの種別は，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの IPv6 アドレスの種別が適用されます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，IPv6 アドレスの種別は初期値に変更されます。

使用例

```
:SYST:COMM:NET:IPVS:ADDR:TYPE RA_EUI64
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVS:ADDR:TYPE?
> RA_EUI64
```

:SYSTem:COMMunicate:NET:IPVSix:CURRent:DEFRouter?

Get IPv6 Default Router Address

機能

現在のデフォルト・ルータ・アドレスを問い合わせます。

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:CURRent:DEFRouter?
```

レスポンス

<ip6addr>

パラメータ

<ip6addr>	デフォルト・ルータ・アドレス xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

詳細

通信先モジュールの IPv6 アドレス/プリフィックス長を問い合わせます。
モジュールの IPv6 アドレスがオフに設定されている場合，レスポンスは 0 になります。

使用例

:SYST:COMM:NET:IPVS:CURR:DEFR?
> 2001:DB8::1

:SYSTem:COMMunicate:NET:IPVSix:CURRent:IPADdress?

Get Current IPv6 Address and Prefix

機能

現在の IPv6 アドレス/プリフィックス長を問い合わせます。

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:CURRent:IPADdress?
```

レスポンス

<num>,<ip6addr>,…

パラメータ

<num>	設定された IP アドレスの個数
<ip6addr>	IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx

詳細

通信先モジュールの IPv6 アドレス/プリフィックス長を問い合わせます。
モジュールの IPv6 アドレスがオフに設定されている場合，レスポンスは 0 になります。

使用例

:SYST:COMM:NET:IPVS:CURR:IPAD?
> 2,2001:DB8::1/64,2001:DB8::2/64

:SYSTem:COMMunicate:NET:IPVSix:CURRent:LOCaladdress?

Get IPv6 Link Local Address

機能

現在のリンク・ローカル・アドレス/プリフィックス長を問い合わせます。

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:CURRent:LOCaladdress?
```

レスポンス

<ip6addr>

パラメータ

<ip6addr>	リンク・ローカル・アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx

詳細

通信先モジュールのリンク・ローカル・アドレス/プリフィックス長を問い合わせます。
モジュールの IPv6 アドレスがオフに設定されている場合，レスポンスは 0 になります。

使用例

```
:SYST:COMM:NET:IPVS:CURR:LOC?
> 2001:DB8::1/64
```

:SYSTem:COMMunicate:NET:IPVSix:DNS:AUTO:ENABle

IPv6 DNS Server Auto

機能

DNS サーバのアドレスを自動的に取得するかどうかを設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVSix:DNS:AUTO:ENABle <switch>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:DNS:AUTO:ENABle?
```

レスポンス

<switch>

パラメータ

<switch>	DNS サーバのアドレス自動取得オンオフ設定
ON	オンにします。
OFF	オフにします。

詳細

MT8870A に装着されている全モジュールに設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS サーバアドレスの自動取得は OFF に変更されます。

使用例

```
:SYST:COMM:NET:IPVS:DNS:AUTO:ENAB ON
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVS:DNS:AUTO:ENAB?
> ON
```

:SYSTem:COMMunicate:NET:IPVSix:DNS:PRIMary

IPv6 DNS Primary Address

機能

DNS のプライマリ・アドレス（IPv6）を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVSix:DNS:PRIMary <ip6addr>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:DNS:PRIMary?
```

レスポンス

<ip6addr>

パラメータ

<ip6addr>	DNS プライマリ・アドレス xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

詳細

MT8870A に装着されている全モジュールに DNS プライマリ・アドレスを設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS プライマリ・アドレスは 0 に変更されます。

使用例

```
:SYST:COMM:NET:IPVS:DNS:PRIM "fe80::215:c5ff:febb:42cf"
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVS:DNS:PRIM?
> fe80::215:c5ff:febb:42cf
```

:SYSTem:COMMunicate:NET:IPVSix:DNS:SECondary

IPv6 DNS Secondary Address

機能

DNS セカンダリ・アドレス (IPv6) を設定, または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVSix:DNS:SECondary <ip6addr>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:DNS:SECondary?
```

レスポンス

<ip6addr>

パラメータ

<ip6addr>	DNS セカンダリ・アドレス xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

詳細

MT8870A に装着されている全モジュールに DNS セカンダリ・アドレスを設定します。
本コマンドの設定を有効にするには, :SYSTem:COMMunicate:NET:RESTart を送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると, DNS セカンダリ・アドレスは 0 に変更されます。

使用例

```
:SYST:COMM:NET:IPVSix:DNS:SEC "fe80::215:c5ff:febb:42ce"
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVSix:DNS:SEC?
> fe80::215:c5ff:febb:42ce
```

:SYSTem:COMMunicate:NET:IPVSix:STATic:DEFRouter

IPv6 Default Router

機能

固定 IPv6 設定のデフォルト・ルータ・アドレスを設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVSix:STATic:DEFRouter <ip6addr>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:STATic:DEFRouter?
```

レスポンス

<ip6addr>

パラメータ

<ip6addr>	デフォルト・ルータ・アドレス xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

詳細

MT8870A に装着されている全モジュールにデフォルト・ルータ・アドレスを設定します。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
パラメータに"" (空文字) を指定すると，デフォルト・ルータ・アドレスの設定を初期化できます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，デフォルト・ルータ・アドレスは 0 に変更されます。

使用例

```
:SYST:COMM:NET:IPVS:STAT:DEFR "2001:DB8::1"
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVS:STAT:DEFR?
> 2001:DB8::1
```

:SYSTem:COMMunicate:NET:IPVSix:STATic:IPADdress

IPv6 Address

機能

固定 IPv6 設定の IPv6 アドレス/プリフィックス長を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:IPVSix:STATic:IPADdress
<ipv6_addr1>,<ipv6_addr2>,<ipv6_addr3>,<ipv6_addr4>
```

クエリ

```
:SYSTem:COMMunicate:NET:IPVSix:STATic:IPADdress?
```

レスポンス

<ipv6_addr1>,<ipv6_addr21>,<ipv6_addr3>,<ipv6_addr4>

パラメータ

<ipv6_addr1>	スロット 1 の IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx
<ipv6_addr2>	スロット 2 の IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx
<ipv6_addr3>	スロット 3 の IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx
<ipv6_addr4>	スロット 4 の IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx

詳細

MT8870A のスロットの IPv6 アドレスを設定します。モジュールが挿入されていないスロットにも IPv6 アドレスを設定できます。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart を送信します。
IPv6 アドレスは，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの IPv6 アドレスが適用されます。

使用例

```
:SYST:COMM:NET:IPVS:STAT:IPAD
"2001:DB8::1/64","2001:DB8::2/64","2001:DB8::3/64","2001:DB8::4/64"
:SYST:COMM:NET:REST

:SYST:COMM:NET:IPVS:STAT:IPAD?
> 2001:DB8::1/64,2001:DB8::2/64,2001:DB8::3/64,2001:DB8::4/64
```

:SYSTem:COMMunicate:NET:PORT:SETTing

Raw Socket Port Number

機能

RawSocket 通信時のポート番号を設定，または問い合わせます。

コマンド

```
:SYSTem:COMMunicate:NET:PORT:SETTing <port>
```

クエリ

```
:SYSTem:COMMunicate:NET:PORT:SETTing?
```

レスポンス

<port>

パラメータ

<port>	ポート番号
範囲	5001～60000
初期値	56001

詳細

MT8870A と RawSocket 通信を行う場合の TCP/IP のポート番号を設定します。
ポート番号は IPv4 および IPv6 で共通となります。
本コマンドの設定を有効にするには，:SYSTem:COMMunicate:NET:RESTart コマンドを送信します。

使用例

```
:SYSTem:COMMunicate:NET:PORT:SETTing 56001
:SYSTem:COMMunicate:NET:RESTart
:SYSTem:COMMunicate:NET:PORT:SETTing?
> 56001
```

:SYSTem:COMMunicate:NET:RESTart

Network Restart

機能

ネットワークを再起動します。

コマンド

```
:SYSTem:COMMunicate:NET:RESTart
```

詳細

本コマンドを送信すると，システムが再起動します。再起動中は全モジュールのリモート制御ができません。
リモート制御を続ける場合は，本コマンドの送信後，しばらくしてからリモートを再接続してください。

使用例

:SYST:COMM:NET:REST

備考

*RST，および:SYSTem:REBoot では通信設定が更新されません。

:SYSTem:DATE

Set Date

機能

日時を設定，または問い合わせます。

コマンド

```
:SYSTem:DATE <year>,<month>,<day>,<hour>,<minute>,<second>
```

クエリ

```
:SYSTem:DATE?
```

レスポンス

<year>,<month>,<day>,<hour>,<minute>,<second>

パラメータ

<year>	年
範囲	2012～2027
分解能	1
<month>	月
範囲	1～12
分解能	1
<day>	日
範囲	1～31
分解能	1
<hour>	時
範囲	0～23
分解能	1
<minute>	分
範囲	0～59
分解能	1
<second>	秒
範囲	0～59
分解能	1

使用例

:SYSTem:DATE 2012,9,13,11,6,17
:SYSTem:DATE?
> 2012,9,13,11,6,17

備考

日時の設定に失敗した場合，レスポンスは"***"になります。

:SYSTem:ERRor?

System Error

機能

エラー・イベントキューに存在するエラーメッセージを問い合わせます。

クエリ

```
:SYSTem:ERRor?
```

レスポンス

<code>,<msg>

パラメータ

<code>	エラーコード
0	"No Error"
–113	"Undefined header"
–200	"Execution error"
–220	"Parameter error"
–310	"Option license is required to execute the command"
–370	"Main frame upgrade is required to execute the command"
<msg>	エラー・イベント番号に対応したイベントメッセージ

詳細

エラー・イベントキューが空の場合のレスポンスは，"0,"No Error""です。

使用例

```
:SYST:ERR?
> -220,"Parameter error"
```

:SYSTem:ERRor:ALL?

System All Errors

機能

エラー・イベントキューに存在するエラーメッセージを問い合わせます。

クエリ

```
:SYSTem:ERRor:ALL?
```

レスポンス

<code>,<msg>,<code>,<msg>.....

パラメータ

<code>	エラーコード
0	"No Error"
–113	"Undefined header"
–200	"Execution error"
–220	"Parameter error"
–310	"Option license is required to execute the command"
–370	"Main frame upgrade is required to execute the command"
<msg>	エラー・イベント番号に対応したイベントメッセージ

詳細

エラー・イベントキューが空の場合のレスポンスは，"0,"No Error""です。
実行後，エラー・イベントキューは空になります。

使用例

```
:SYST:ERR:ALL?
> -113,"Undefined header",-220,"Parameter error"
```

:SYSTem:ERRor:CODE:ALL?

System All Error Codes

機能

エラー・イベントキューに存在するすべてのエラーコードを問い合わせます。

クエリ

```
:SYSTem:ERRor:CODE:ALL?
```

レスポンス

<code>,<code>,....

パラメータ

<code>	エラーコード
0	"No Error"
–113	"Undefined header"
–200	"Execution error"
–220	"Parameter error"
–310	"Option license is required to execute the command"
–370	"Main frame upgrade is required to execute the command"

詳細

エラー・イベントキューが空の場合のレスポンスは，0 です。

実行後，エラー・イベントキューは空になります。

使用例

```
:SYST:ERR:CODE:ALL?
> -113,-220
```

:SYSTem:ERRor:CODE[:NEXT]?

System Error Code

機能

エラー・イベントキューに存在するエラーコードを問い合わせます。

クエリ

```
:SYSTem:ERRor:CODE[:NEXT]?
```

レスポンス

<code>

パラメータ

<code>	エラーコード
0	"No Error"
–113	"Undefined header"
–200	"Execution error"
–220	"Parameter error"
–310	"Option license is required to execute the command"
–370	"Main frame upgrade is required to execute the command"

詳細

エラー・イベントキューが空の場合のレスポンスは，0 です。

使用例

```
:SYST:ERR:CODE?
> -220
```

:SYSTem:ERRor:COUNt?

System Error Count

機能

エラー・イベントキューに存在するすべてのエラー・イベントの数を問い合わせます。

クエリ

```
:SYSTem:ERRor:COUNt?
```

レスポンス

<num>

パラメータ

<num>	エラー・イベント数

詳細

エラー・イベントキューは最大 4 個までキューイングします。
エラーが 5 個以上発生した場合は，最も古いエラーが削除されます。
:SYSTem:ERRor?，:SYSTem:ERRor:CODE?コマンドで，最も古いエラーから順に取得し，エラー・イベントキューは 1 つ削除されます。
:SYSTem:ERRor:ALL?，:SYSTem:ERRor:CODE:ALL?コマンドで，すべてのエラーを取得した後，エラー・イベントキューは空になります。

使用例

```
:SYST:ERR:COUN?
> 3
```

:SYSTem:INFormation?

Module Information

機能

モジュールの情報を問い合わせます。

クエリ

```
:SYSTem:INFormation? <index>
```

レスポンス

パラメータが 1 の場合
　model name　　品名（30 文字以内）
パラメータが 2 の場合
　model number　　形名（20 文字以内）
パラメータが 3 の場合
　BIOS version　　BIOS (IPL) version (15 文字以内)
パラメータが 4 の場合
　OS version　　OS version (255 文字以内)
パラメータが 5 の場合
　number of HW,name/rev,name/rev,...,name/rev
　　number of HW:　　ハードウェアの数
　　name:　　ハードウェア名称（30 文字以内, ""でくくられる）
　　rev:　　ハードウェアレビジョン（10 文字以内）
パラメータが 6 の場合
　number of FPGA,name/ver,name/ver,...,name/ver
　　number of FPGA:　　FPGA の数
　　name:　　FPGA 名称（30 文字以内, ""でくくられる）
　　ver:　　FPGA バージョン（"xx.xx.xx"で記述）
パラメータが 7 の場合
　number of HW option,num/name/sw,num/name/sw,...,num/name/sw
　　number of HW option:　ハードウェアオプションの数
　　num:　　オプション番号（0～999）
　　name:　　オプション名称（50 文字以内, ""でくくられる）
　　sw:　　オプションスイッチ
　　ON:　　オプション有り
　　OFF:　　オプション無し
パラメータが 8 の場合
　常に 0 です。
パラメータが 9 の場合
　常に 0 です。
パラメータが 10 の場合
　常に 0 です。
パラメータが 0, または, 省略した場合
　上記のすべてのレスポンスをコンマで連結した文字列

パラメータ

<index>　　取得情報のインデックス
　範囲　　0～10

使用例

:SYST:INF? 1
> TRX Test Module
:SYST:INF? 2
> MU887002A
:SYST:INF? 3
> 3.01.10.5.11
:SYST:INF? 4
> CentOS Linux release 7.4.1708 (Core) /Linux 3.10.0-693.el7.x86_64
:SYST:INF? 5
> 5,"Carrier Board"/00, "BB Board"/01, "RF Board"/01,"Divider1 Board"/01,"Divider2 Board"/01
:SYST:INF? 6
> 4,"BB FPGA"/00.00.92,"RF FPGA"/07.00.00,"Divider1 FPGA"/16.00.00,"Divider2 FPGA"/16.00.00
:SYST:INF? 7
> 1, 007/"7GHz Frequency Extension"/ON
:SYST:INF? 8
> 0
:SYST:INF? 9
> 0
:SYST:INF? 10
> 0

関連コマンド

:SYSTem:INFormation:MAINframe?

備考

インストールされている期限付きライセンスの期限を過ぎている場合は，本コマンドでライセンス情報を取得できません。

:SYSTem:INFormation:DEVice:ID?

Module Serial No.

機能

モジュールのシリアル番号を問い合わせます。

クエリ

```
:SYSTem:INFormation:DEVice:ID?
```

レスポンス

<serial>

パラメータ

<serial>	モジュールのシリアル番号 10 桁の数字です。 例 6200103056

使用例

:SYST:INF:DEV:ID?
> 6123456789

備考

MU887002A の場合, TRX1, TRX2 ともモジュールのシリアル番号は同じ値が返ります。

:SYSTem:INFormation:MAINframe?

Main Frame Information

機能

MT8870A の情報を問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe? <index>
```

レスポンス

パラメータが 1 の場合
　model name　　品名（30 文字以内）
パラメータが 2 の場合
　model number　　形名（20 文字以内）
パラメータが 3 の場合
　BIOS version　　常に 0 です。
パラメータが 4 の場合
　OS version　　常に 0 です。
パラメータが 5 の場合
　number of HW,name/rev,name/rev,...,name/rev
　　number of HW:　　ハードウェアの数
　　name:　　ハードウェア名称（30 文字以内, ""でくくられる）
　　rev:　　ハードウェアレビジョン（10 文字以内）
パラメータが 6 の場合
　number of FPGA,name/ver,name/ver,...,name/ver
　　number of FPGA:　　FPGA の数
　　name:　　FPGA 名称（30 文字以内, ""でくくられる）
　　ver:　　FPGA バージョン（10 文字以内）
パラメータが 7 の場合
　number of HW option,num/name/sw,num/name/sw,...,num/name/sw
　　number of HW option:　ハードウェアオプションの数
　　num:　　オプション番号（0～999）
　　name:　　オプション名称（50 文字以内, ""でくくられる）
　　sw:　　オプションスイッチ
　　　ON:　　オプション有り
　　　OFF:　　オプション無し
パラメータが 8 の場合
　number of Software
　license,model/serial/name/ver,model/serial/name/ver,...,model/serial/name/ver
　　model:　　ソフトウェア形名（30 文字以内）
　　serial:　　シリアル番号（10 文字）
　　name:　　ソフトウェアライセンス名称（50 文字以内, ""でくくられる）
　　ver:　　ソフトウェアバージョン（"xx.xx"で記述）
パラメータが 9 の場合
　number of SW option,num/name/sw,num/name/sw,...,num/name/sw
　　number of SW option:　ソフトウェアオプションの数
　　num:　　オプション形名（20 文字以内 MX887000A-001）

name: オプション名称（50 文字以内，""でくくられる）
sw: オプションスイッチ
ON: オプション有り
OFF: オプション無し

パラメータが 10 の場合
number of SG wave license,model/serial/name/maxver,model/serial/name/maxver,...,model/serial/name/maxver
number of SG wave license: SG 波形ライセンスの数
model: SG 波形ライセンス形名（30 文字以内）
serial: シリアル番号（10 文字）
name: SG 波形ライセンス名称（50 文字以内）
maxver: 使用可能な最大バージョン（0～999）

パラメータが 0，または省略した場合
上記のすべてのレスポンスをコンマで連結した文字列

パラメータ

<index> 取得情報のインデックス

使用例

```
:SYST:INF:MAIN? 1
> Universal Wireless Test Set
:SYST:INF:MAIN? 2
> MT8870A
:SYST:INF:MAIN? 3
> 1.0
:SYST:INF:MAIN? 4
> 2.6
:SYST:INF:MAIN? 5
> 2, "Main Board/01","Option Board/FE"
:SYST:INF:MAIN? 6
> 2, "High Speed Control FPGA/01","Wide Band FPGA/FF"
:SYST:INF:MAIN? 7
> 2,001/"Rubidium"/ON,009/"GPIB Interface"/ON
:SYST:INF:MAIN? 8
> 2,MX887010A/6123456789/"Cellular Measurement
Software"/01.01,MX887020A/6123456789/"Vector SG Software"/01.00
:SYST:INF:MAIN? 9
> 2,MX887000A-001/"Extended Digitizing Option"/ON,MX887000A-002/"Phase Noise
Measurement"/OFF
:SYST:INF:MAIN? 10
> 1,MX887070A/6123456789/Standard wave license/5
```

関連コマンド

:SYSTem:INFormation?

備考

インストールされている期限付きライセンスの期限を過ぎている場合は，本コマンドでライセンス情報を取得できません。

:SYSTem:INFormation:MAINframe:DEVice:ID?

Main Frame Serial No.

機能

MT8870A のシリアル番号を問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:DEVice:ID?
```

レスポンス

<serial>

パラメータ

<serial>	シリアル番号 10 桁の数字です。 例 6200112233

使用例

:SYST:INF:MAIN:DEV:ID?
> 6123456789

:SYSTem:INFormation:MAINframe:PACKage:VERSion?

Firmware Version

機能

ファームウェアバージョンを問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:PACKage:VERSion?
```

レスポンス

<ver>

パラメータ

<ver>	ファームウェアバージョン（"xx.xx.xx"で記述）

使用例

:SYSTem:INFormation:MAINframe:PACKage:VERSion?
> 1.00.00

:SYSTem:INFormation:MAINframe:POWeron:COUNt?

Chassis Power ON Count

機能

MT8870A で電源を投入した回数を問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:POWeron:COUNt?
```

レスポンス

<num>

パラメータ

<num>	MT8870A の電源投入回数

使用例

```
:SYSTem:INFormation:MAINframe:POWeron:COUNt?
> 1971
```

:SYSTem:INFormation:MAINframe:RTIMe?

Main Frame Running Time Count Read

機能

MT8870A の積算稼働時間を問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:RTIMe?
```

レスポンス

<time>

パラメータ

<time>	積算稼働時間（分単位）

使用例

:SYST:INF:MAIN:RTIM?
> 1234

:SYSTem:INFormation:MAINframe:SOFTware?

Type Name of Software

機能

ソフトウェア形名を問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:SOFTware?
```

レスポンス

<n>,<model>,…,<model>

パラメータ

<n>	ソフトウェア数
<model>	ソフトウェア形名

使用例

:SYST:INF:MAIN:SOFT?
> 3,MX887010A,MX887011A,MX887012A

備考

オーダリング情報に登録されている形名のみを取得できます。

:SYSTem:INFormation:MAINframe:SOFTware:OPTion?

Option Number

機能

ソフトウェア Option 番号を問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:SOFTware:OPTion? <model>
```

レスポンス

<n>,<num>,...,<num>

パラメータ

<model>	ソフトウェア形名
<n>	オプションの数
<num>	オプション番号

使用例

下記がインストールされている場合

- MX887011A
- オプション無し

```
:SYST:INF:MAIN:SOFT:OPT? MX887011A
> 0
```

備考

オーダリング情報に登録されている形名のみを取得できます。
無効なソフトウェア形名をパラメータに設定した場合はパラメータエラーとなります。

:SYSTem:INFormation:MAINframe:TEMPerature?

Main Frame Temperature

機能

MT8870A の温度を問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:TEMPerature?
```

レスポンス

<temp>　　MT8870A の温度（°C）
　分解能:　　0.01

使用例

:SYST:INF:MAIN:TEMP?
> 47.12

:SYSTem:INFormation:MAINframe:WAVeform?

Type Name of Waveform

機能

波形ライセンスを問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:WAVeform?
```

レスポンス

<n>,<model>,...,<model>

パラメータ

<n>　　波形ライセンス数
<model>　　波形ライセンス形名

使用例

:SYST:INF:MAIN:WAV?
> 2,MV887011A,MV887012A

備考

オーダリング情報に登録されている形名のみを取得できます。

:SYSTem:INFormation:MAINframe:TRX?

TRX 1/2

機能

TRX 1/2 を問い合わせます。

クエリ

```
:SYSTem:INFormation:MAINframe:TRX?
```

レスポンス

<trx>

パラメータ

<trx>	TRX 1/2
1	TRX 1
2	TRX 2

使用例

:SYST:INF:MAIN:TRX?
> 1

:SYSTem:INFormation:POWeron:COUNt?

Module Power ON Count

機能

モジュールで電源を投入した回数を問い合わせます。

クエリ

```
:SYSTem:INFormation:POWeron:COUNt?
```

レスポンス

<num>

パラメータ

<num>	モジュールの電源投入回数

使用例

:SYSTem:INFormation:POWeron:COUNt?
> 2525

:SYSTem:INFormation:PLUG:COUNt?

Count of Module Insertion

機能

モジュールの挿抜回数を問い合わせます。

クエリ

```
:SYSTem:INFormation:PLUG:COUNt?
```

レスポンス

<num>

パラメータ

<num>	モジュールの挿抜回数

詳細

注:

モジュールを別のスロットに移動した場合に，挿抜回数がカウントされます。
同じ MT8870A の同じスロットにモジュールを抜き差しした場合は，挿抜回数がカウントされません。

使用例

:SYSTem:INFormation:PLUG:COUNt?
> 40

:SYSTem:INFormation:RTIMe?

Module Running Time Count Read

機能

モジュールの積算稼働時間を問い合わせます。

クエリ

```
:SYSTem:INFormation:RTIMe?
```

レスポンス

<time>

パラメータ

<time>		積算稼働時間 (分単位)

使用例

```
:SYST:INF:RTIM?
> 1234
```

:SYSTem:INFormation:SLOT?

Module Slot Number

機能

モジュールが装着されているスロット番号を問い合わせます。

クエリ

```
:SYSTem:INFormation:SLOT?
```

レスポンス

<slot>

パラメータ

<slot>		スロット番号
	範囲	1～4

使用例

```
:SYST:INF:SLOT?
> 1
```

:SYSTem:INFormation:TEMPerature?

Module Temperature

機能

モジュールの温度を問い合わせます。

クエリ

```
:SYSTem:INFormation:TEMPerature?
```

レスポンス

<temp>

パラメータ

<temp>	モジュールの温度（℃）
分解能:	0.01

使用例

:SYST:INF:TEMP?
> 47.12

:SYSTem:LANGuage

Language Selection of Remote Command

機能

リモート制御コマンドの言語モードを切り替えます。

コマンド

```
:SYSTem:LANGuage <mode>
```

クエリ

```
:SYSTem:LANGuage?
```

レスポンス

<mode>

パラメータ

<mode>	言語モード
NAT	Native
SCPI	SCPI

使用例

:SYST:LANG NAT
:SYST:LANG?
> NAT

:SYSTem:LAST:ERRor?

System Last Error

機能

前回終了時の異常を問い合わせます。

クエリ

```
:SYSTem:LAST:ERRor?
```

レスポンス

<msg>

パラメータ

<msg>	異常内容
NO ERROR	異常なし
TemperatureAlarm	温度異常により強制シャットダウン

詳細

本コマンドを実行しても前回終了時の異常の情報はクリアされません。

使用例

:SYSTem:LAST:ERRor?
> NO ERROR

:SYSTem:REBoot

Reboot

機能

システムを再起動します。

コマンド

```
:SYSTem:REBoot
```

詳細

本コマンドを送信すると，システムが再起動します。再起動中は MT8870A に装着されている全モジュールがリモート制御できません。
本コマンドの送信後，MT8870A の Error ランプが消灯すると，リモート制御ができます。

使用例

:SYST:REB

関連コマンド

:SYSTem:SHUTdown

:SYSTem:SELF:TEST?

System Test

機能

モジュールの自己診断結果と実行した日時を問い合わせます。

クエリ

```
:SYSTem:SELF:TEST?
```

レスポンス

<status1>,<status2>,<status3>,...,<year>,<month>,<day>,<hour>,<min>
<status>の詳細は以下のようなフォーマットになります。

"Chassis EEPROM: xx","Module EEPROM: xx","RAM: xx","Correction ROM: xx","FPGA Configuration: xx"

xx には"Pass"または"Fail"が入ります。

パラメータ

<status>	最後実行したセルフテストの結果
<year>	年 (2001～)
<month>	月 (1～12)
<day>	日 (1～31)
<hour>	時 (0～23)
<min>	分 (0～59)

使用例

:SYST:SELF:TEST?
> "Chassis EEPROM: Pass","Module EEPROM:Pass","RAM: Pass","Correction ROM: Pass","FPGA Configuration:Pass",2012,4,5,12,34

:SYSTem:SELF:TEST:ERRor:COUNt?

System Test Error Count

機能

モジュールの自己診断を失敗した回数を問い合わせます。

クエリ

```
:SYSTem:SELF:TEST:ERRor:COUNt?
```

レスポンス

<count>

パラメータ

<count>	モジュールの自己診断を失敗した回数
Range	0 以上

使用例

```
:SYST:SELF:TEST:ERR:COUN?
> 0
```

:SYSTem:SELF:TEST:ERRor:LASTtime?

System Test Error Last Time

機能

最後にモジュールの自己診断を失敗した日時を問い合わせます。

クエリ

```
:SYSTem:SELF:TEST:ERRor:LASTtime?
```

レスポンス

<year>,<month>,<day>,<hour>,<min>

パラメータ

<year>	年 (2001～)
<month>	月 (1～12)
<day>	日 (1～31)
<hour>	時 (0～23)
<min>	分 (0～59)

使用例

```
:SYST:SELF:TEST:ERR:LAST?
> 2012,1,2,3,45
```

備考

モジュールの自己診断を 1 度も失敗していない場合, レスポンスは"***"になります。

:SYSTem:SHUTdown

Shutdown

機能

システムをシャットダウンします。

コマンド

```
:SYSTem:SHUTdown
```

詳細

本コマンドを送信すると，全モジュールのリモートの接続が切断されます。

使用例

:SYST:SHUT

関連コマンド

:SYSTem:REBoot

:SYSTem:VERSion?

Package Version

機能

パッケージバージョンを問い合わせます。

クエリ

```
:SYSTem:VERSion?
```

レスポンス

<ver>

パラメータ

<ver>	パッケージバージョン（"xx.xx.xx"で記述）

使用例

:SYST:VERS?
> 01.00.00

5.2.2 VSGコマンド

:SOURce:GPRF:GENerator:ARB:FILE:LOAD

Waveform File Loading

機能

記憶媒体に保存されている波形ファイルを波形メモリにロードします。

コマンド

```
:SOURce:GPRF:GENerator:ARB:FILE:LOAD <file_name>
```

クエリ

```
:SOURce:GPRF:GENerator:ARB:FILE:LOAD? <file_name>
```

レスポンス

<status>

パラメータ

<file_name>	波形ファイル名
<status>	波形ファイルの状態
0	すでに波形メモリにロード済み
1	波形メモリにロード可能
2	波形ファイルを使用するライセンスが必要
3	該当する波形ファイルなし
4	波形メモリの空き容量不足
5	波形メモリを最適化すればロード可能
6	波形情報ファイルのパラメータエラー
7	波形メモリへのロード中
8	波形メモリの最適化中
9	MU887002A で使用できない波形ファイルを設定

詳細

本コマンドを使用するには，「表 1.3.3-2 波形ファイル」に示す波形ファイルが 1 つ以上必要です。
<file_name>に指定するファイル名は，次の説明を参照してください。
『セルラアプリケーション用波形ファイル 取扱説明書』の「第 3 章 波形ファイルの詳細」
『GNSS & Broadcast 用波形ファイル 取扱説明書』の「第 3 章 波形ファイルの詳細」
『SRW アプリケーション用波形ファイル 取扱説明書』の「第 4 章 波形ファイル」

波形パターン再生中に本コマンドを使用して波形ファイルをロードすると，波形パターンの再生が停止します。波形ファイルのロードが終了すると，自動的に波形パターンの先頭から再生を再開します。

使用例

記憶媒体の波形ファイル"MV887011A_WCDMA_0002.xml"を波形メモリにロードします。
:SOUR:GPRF:GEN:ARB:FILE:LOAD "MV887011A_WCDMA_0002.xml"

記憶媒体の波形ファイル"MV887013A_LTEFDD_0004.xml"の状態を問い合わせます。
:SOUR:GPRF:GEN:ARB:FILE:LOAD? "MV887013A_LTEFDD_0004.xml"
> 0

関連コマンド

:SOURce:GPRF:GENerator:ARB:WAVeform:DEFRag

備考

動作モードが以下の状態の場合，実行エラーとなります。

・ シーケンスモード

波形ロード中は，ロードキャンセル，クエリ以外のコマンドは使用できません（実行エラー）。

指定したファイルが存在しない場合，パラメータエラーとなります。

波形ファイル名の拡張子（xml）は省略可能です。

:SOURce:GPRF:GENerator:ARB:FILE:LOAD:CANCel

Waveform File Loading Cancel

機能

波形メモリへのファイルロードを中断します。

コマンド

```
:SOURce:GPRF:GENerator:ARB:FILE:LOAD:CANCel
```

使用例

波形メモリへのファイルロードを中断します。

:SOUR:GPRF:GEN:ARB:FILE:LOAD:CANC

備考

動作モードが以下の状態の場合，実行エラーとなります。

- シーケンスモード

波形ロード中でない場合，何もしません。

:SOURce:GPRF:GENerator:ARB:FILE:LOAD:STATus?

Waveform Load Status

機能

波形ファイルのロード実行状態を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:FILE:LOAD:STATus?
```

レスポンス

<status_res>

パラメータ

<status_res>	波形ファイルのロード実行状態
0	ロード実行していない（ロード未実行，または実行済み）
1	ロード実行中

使用例

波形ファイルのロード実行状態を問い合わせます。

:SOUR:GPRF:GEN:ARB:FILE:LOAD:STAT?

> 1

:SOURce:GPRF:GENerator:ARB:FILE:PATTern?

Waveform Group Information

機能

記憶媒体に保存されている波形ファイルのグループ番号を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:FILE:PATTern? <file_name>,<group_index>
```

レスポンス

<group_number>

パラメータ

<file_name>	波形ファイル名
<group_index>	波形ファイルに保存されている波形パターンの順番
範囲	1 ～ 波形ファイルに保存されている波形パターン数
<group_number>	グループ番号

使用例

記憶媒体に保存されている波形ファイルの，2 番目の波形パターンのグループ番号を問い合わせます。

```
:SOUR:GPRF:GEN:ARB:FILE:PATT? "MV887070A_FMRDS_0001.xml",2
> 2
```

備考

波形ファイル名の拡張子（xml）は省略可能です。

指定したファイルが存在しない場合，パラメータエラーとなります。

:SOURce:GPRF:GENerator:ARB:FILE:PATTern:COUNt?

Waveform Group Count

機能

記憶媒体に保存されている波形ファイルの波形パターン数を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:FILE:PATTern:COUNt? <file_name>
```

レスポンス

<count>

パラメータ

<file_name>	波形ファイル名
<count>	波形パターン数

使用例

記憶媒体に保存されている波形ファイルに含まれる，波形パターン数を問い合わせます。

```
:SOUR:GPRF:GEN:ARB:FILE:PATT:COUN? "MV887013A_LTEFDD_0002.xml"
> 7
```

備考

波形ファイル名の拡張子（xml）は省略可能です。

:SOURce:GPRF:GENerator:ARB:FILE:PATTern:NAME?

Waveform Pattern name

機能

記憶媒体に保存されている波形ファイルのグループ番号を指定して，パターン名（タイトル）を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:FILE:PATTern:NAME? <file_name>,<group_number>
```

レスポンス

<pattern_name>

パラメータ

<file_name>	波形ファイル名
<group_number>	グループ番号
範囲	波形ファイルで定義されたグループ番号の範囲
<pattern_name>	パターン名（タイトル）

使用例

記憶媒体に保存されている波形ファイル"MV887111A_ISDBT_0001.xml"のグループ番号2に記載されているパターン名（タイトル）を問い合わせます。

```
:SOUR:GPRF:GEN:ARB:FILE:PATT:NAME? "MV887111A_ISDBT_0001.xml",2
> Mode:3, GI:1/8, Data:PN23fix, (A:1segPR,QPSK), (B:12seg,64QAM)
```

備考

波形ファイル名の拡張子（xml）は省略可能です。

指定したファイル，またはグループ番号が存在しない場合，パラメータエラーとなります。

:SOURce:GPRF:GENerator:ARB:FILE:VERSion?

Waveform Version

機能

記憶媒体に保存されている波形ファイルのバージョンを問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:FILE:VERSion? <file_name>
```

レスポンス

<version>

パラメータ

<file_name>	波形ファイル名
<version>	バージョン番号

使用例

記憶媒体に保存されている波形ファイル"MV887030A_b_2_100L.xml"のバージョンを問い合わせます。

:SOUR:GPRF:GEN:ARB:FILE:VERS? "MV887030A_b_2_100L.xml"

> 01.00.00

備考

波形ファイル名の拡張子（xml）は省略可能です。

指定したファイルが存在しない場合，パラメータエラーとなります。

以下の処理が実行されている場合，レスポンスは"–999"となります。

・波形ロード

・波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:NOISe:CN

AWGN C/N Ratio

機能

AWGN が On のときのキャリアと AWGN の出力レベル比（C/N）を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:ARB:NOISe:CN <cn_ratio>
```

クエリ

```
:SOURce:GPRF:GENerator:ARB:NOISe:CN?
```

レスポンス

<cn_ratio>

単位　　dB

パラメータ

<cn_ratio>	C/N 設定値
範囲	–12～40 dB
分解能	1 dB
サフィックスコード	DB，　省略した場合は DB として扱われます。
初期値	40 dB

使用例

キャリアと AWGN の出力レベル比（C/N）の設定を –10 dB にします。
:SOUR:GPRF:GEN:ARB:NOIS:CN -10

キャリアと AWGN の出力レベル比（C/N）の設定値を問い合わせます。
:SOUR:GPRF:GEN:ARB:NOIS:CN?
> -10

備考

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
・波形ロード
・波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:NOISe:STATe

AWGN Switch

機能

AWGN 出力の On / Off を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:ARB:NOISe:STATe <on_off>
```

クエリ

```
:SOURce:GPRF:GENerator:ARB:NOISe:STATe?
```

レスポンス

<on_off>

パラメータ

<on_off>	AWGN 出力の On / Off
0	Off
1	On
OFF	Off
ON	On
初期値	0

使用例

AWGN を出力します。
:SOUR:GPRF:GEN:ARB:NOIS:STAT ON

AWGN 出力の状態を問い合わせます。
:SOUR:GPRF:GEN:ARB:NOIS:STAT?
> 1

関連コマンド

:SOURce:GPRF:GENerator:ARB:NOISe:CN

備考

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:WAVeform:COUNt?

ARB Memory Loaded files count

機能

波形メモリにロードされている波形ファイルの数を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:COUNt?
```

レスポンス

<count>

パラメータ

<count>	波形メモリにロードされている波形ファイルの数

使用例

波形メモリにロードされている波形ファイルの数を問い合わせます。

:SOUR:GPRF:GEN:ARB:WAV:COUN?

> 2

:SOURce:GPRF:GENerator:ARB:WAVeform:DEFRag

ARB Memory Defrag

機能

波形メモリを最適化して，連続空き容量を増やします。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:DEFRag
```

使用例

波形メモリを最適化します。

:SOUR:GPRF:GEN:ARB:WAV:DEFR

備考

以下の状態の場合，実行エラーとなります。

- シーケンスモード
- 波形メモリデフラグ中のクエリ以外のコマンド
- 波形メモリにロードした波形ファイルが記憶媒体に存在しない場合

:SOURce:GPRF:GENerator:ARB:WAVeform:DELete

Deleting Waveform File in ARB Memory

機能

波形メモリから指定した波形ファイルを削除します。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete <file_name>
```

パラメータ

<file_name>　　波形ファイル名

詳細

ファイル名には，波形メモリにロードされている波形ファイルの名前を指定します。
波形メモリにロードされている波形ファイルの名前を調べるには，次のコマンドを使用します。
:SOURce:GPRF:GENerator:ARB:WAVeform:COUNt?
:SOURce:GPRF:GENerator:ARB:WAVeform:NAME?

使用例

波形メモリから波形ファイル"MV887100A_GPS_0002.xml"を削除します。
:SOUR:GPRF:GEN:ARB:WAV:DEL "MV887100A_GPS_0002.xml"

備考

動作モードが以下の状態の場合，実行エラーとなります。

- シーケンスモード

以下の処理が実行されている場合，実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

指定したファイルがロードされていない場合，実行エラーとなります。
波形ファイル名の拡張子（xml）は省略可能です。

:SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL

Deleting All Waveform Files in ARB Memory

機能

波形メモリからすべての波形ファイルを削除します。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL
```

使用例

波形メモリからすべての波形ファイルを削除します。

:SOUR:GPRF:GEN:ARB:WAV:DEL:ALL

備考

動作モードが以下の状態の場合，実行エラーとなります。

- シーケンスモード

以下の処理が実行されている場合，実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:WAVeform:FREE?

ARB Memory Free Space

機能

波形メモリの空き容量を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:FREE? <target>
```

レスポンス

<blank>,<consecutive_blank>,<total>

パラメータ

<target>	空き容量を読み出すメモリの種類
ALL	すべてのメモリの空き容量を読み出します。
GROUPINFO	グループ情報空間の空き容量を読み出します。
SECTIONINFO	セクション情報空間の空き容量を読み出します。
ARB	ARB メモリの空き容量を読み出します。
<blank>	空き容量 (バイト単位)
<consecutive_blank>	連続空き容量 (バイト単位)
<total>	波形メモリの総容量 (バイト単位)

使用例

波形メモリの空き容量を問い合わせます。

```
:SOUR:GPRF:GEN:ARB:WAV:FREE? GROUPINFO
> 1000,2000,3000
:SOUR:GPRF:GEN:ARB:WAV:FREE? SECTIONINFO
> 10000,20000,30000
:SOUR:GPRF:GEN:ARB:WAV:FREE? ARB
> 100,200,300
:SOUR:GPRF:GEN:ARB:WAV:FREE? ALL
> 1000,2000,3000,10000,20000,30000,100,200,300
```

備考

パラメータに ALL を設定した場合, レスポンスの順番は, グループ情報空間, セクション情報空間, ARB メモリとなります。

以下の処理が実行されている場合, 各項目のレスポンスは"–999"となります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:WAVeform:GENerate:ZVSPattern

Waveform Generate for Zero Vector Signal Pattern

機能

ゼロ波形の波形パターンを生成して波形メモリにロードします。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:GENerate:ZVSPattern
<file_name>,<sampling_rate>,<sample_num>
```

パラメータ

<file_name>	生成する波形パターンのファイル名（拡張子 xml）
<sampling_rate>	サンプリングレート
範囲	2～250 MHz
分解能	1 Hz
サフィックスコード	HZ, Z, KHZ, KZ, MHZ, MZ, GHZ, GZ 省略した場合は HZ として扱われます。
<sample_num>	サンプル数
範囲	512～100,000
分解能	1

使用例

ゼロ波形の波形パターンを生成して波形メモリにロードします。

:SOUR:GPRF:GEN:ARB:WAV:GEN:ZVSP "ZeroWave.xml",2MHZ,512

備考

動作モードが以下の状態の場合，実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

ファイル名の拡張子（xml）は省略可能です。
同名のファイルがすでに存在する場合，実行エラーとなります。

:SOURce:GPRF:GENerator:ARB:WAVeform:NAME?

ARB Memory Loaded file name

機能

波形メモリにロードされている波形ファイルの名称を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:NAME? <numeric_value>
```

レスポンス

<file_name>

パラメータ

<numeric_value>	波形ファイルに割り当てられた任意の番号
<file_name>	波形ファイル名

使用例

波形メモリにロードされている波形ファイルの名称を問い合わせます。
:SOUR:GPRF:GEN:ARB:WAV:NAME? 1
> MV887040A_BLE

備考

設定した番号が波形メモリにロードされている波形ファイル数よりも大きい場合，パラメータエラーとなります。
レスポンスの波形ファイル名は，拡張子（.xml）を含まない文字列です。

:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern?

ARB Memory Group Information

機能

波形メモリにロードされている波形ファイルのグループ番号を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern? <file_name>,<group_index>
```

レスポンス

<group_number>

パラメータ

<file_name>	波形ファイル名
<group_index>	波形ファイルに保存されている波形パターンの順番
範囲	1～波形ファイルに保存されている波形パターン数
<group_number>	グループ番号

使用例

波形メモリにロードされている波形ファイルの，2 番目の波形パターンのグループ番号を問い合わせます。

```
:SOUR:GPRF:GEN:ARB:WAV:PATT? "MV887016A_EVDO_0002",2
> 2
```

備考

ファイル名は拡張子（xml）を除いた文字列です。

指定したファイルがロードされていない場合，実行エラーとなります。

以下の処理が実行されている場合，レスポンスは"–999"となります。

・波形ロード

・波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:COUNt?

ARB Memory Group Count

機能

波形メモリにロードされている波形ファイルの波形パターン数を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:COUNt? <file_name>
```

レスポンス

<count>

パラメータ

<file_name>	波形ファイル名
<count>	波形パターン数

使用例

波形メモリにロードされている波形ファイルに含まれる，波形パターン数を問い合わせます。

```
:SOUR:GPRF:GEN:ARB:WAV:PATT:COUN? "MV887111A_ISDBT_0004"
> 5
```

備考

ファイル名は拡張子（xml）を除いた文字列です。

指定したファイルがロードされていない場合，実行エラーとなります。

以下の処理が実行されている場合，レスポンスは"–999"となります。

・波形ロード

・波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:GAP

ARB Memory Gap Length

機能

波形メモリにロードされている波形ファイルの波形パターンにギャップ長を設定します。
波形メモリにロードされている波形ファイルの波形パターンのギャップ長を問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:GAP
<file_name>,<group_nunber>,<gap_length>
```

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:GAP?
<file_name>,<group_nunber>
```

レスポンス

<gap_length>
　単位　　μs

パラメータ

<file_name>	波形ファイル名
<group_number>	グループ番号
範囲	波形ファイルで定義されたグループ番号の範囲
分解能	1
<gap_length>	ギャップ長（f:波形のサンプリング周波数)
範囲	0, 3～10000 μs (200≦f≦250 MHz)
	0, 30～10000 μs (20≦f＜200 MHz)
	0, 300～100000 μs (2≦f＜20 MHz)
	0, 3～1000 ms (200≦f＜2000 kHz)
	0, 30～10000 ms (20≦f＜200 kHz)
分解能	1 μs (2≦f≦200 MHz)
	1 ms (20≦f＜2000 kHz)
サフィックスコード	S, MS, US, NS　　省略した場合は μs として扱われます。
初期値	0

使用例

波形ファイル"MV887030A_b_11_1024L.xml"のグループ番号 1 の波形パターンに 10 μs のギャップ長を設定します。

```
:SOUR:GPRF:GEN:ARB:WAV:PATT:GAP "MV887030A_b_11_1024L",1,10
```

波形ファイル"MV887030A_b_11_1024L.xml"のグループ番号 1 の波形パターンに付加されているギャップ長を問い合わせます。

```
:SOUR:GPRF:GEN:ARB:WAV:PATT:GAP? "MV887030A_b_11_1024L",1
> 10
```

備考

ファイル名は拡張子（xml）を除いた文字列です。
指定したファイルがロードされていない場合，実行エラーとなります。
指定したグループ番号が存在しない場合，パラメータエラーとなります。

:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:NAME?

ARB Memory Pattern name

機能

波形メモリにロードされている波形ファイルのグループ番号を指定して，パターン名（タイトル）を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:NAME?
<file_name>,<group_number>
```

レスポンス

<pattern_name>

パラメータ

<file_name>	波形ファイル名
<group_number>	グループ番号
範囲	波形ファイルで定義されたグループ番号の範囲
<pattern_name>	パターン名（タイトル）

使用例

波形メモリにロードされている波形ファイル"MV887015A_C2K_0002.xml"のグループ番号 2 のタイトルを問い合わせます。

:SOUR:GPRF:GEN:ARB:WAV:PATT:NAME? "MV887015A_C2K_0002",2

> FL Wave RL Request：PCB +1dB

備考

ファイル名は拡張子（xml）を除いた文字列です。

指定したファイルがロードされていない場合，実行エラーとなります。

以下の処理が実行されている場合，レスポンスは"***"となります。

・波形ロード

・波形メモリデフラグ

指定したグループ番号が存在しない場合，パラメータエラーとなります。

:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:NOTE?

ARB Memory Pattern Note

機能

波形メモリにロードされている波形ファイルのグループ番号を指定して，パターンの情報を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:NOTE?
<file_name>,<group_number>
```

レスポンス

<note>

パラメータ

<file_name>	波形ファイル名
<group_number>	グループ番号
範囲	波形ファイルで定義されたグループ番号の範囲
<note>	メモ

備考

ファイル名は拡張子（xml）を除いた文字列です。
指定したファイルがロードされていない場合，実行エラーとなります。
以下の処理が実行されている場合，レスポンスは"***"となります。
・波形ロード
・波形メモリデフラグ
指定したグループ番号が存在しない場合，パラメータエラーとなります。

:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect

ARB Memory Pattern Selection

機能

波形メモリにロードされている波形ファイルの再生するパターンを選択します。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect
<file_name>[,<group_number>[,<length>]]
```

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect?
```

レスポンス

<file_name>,<group_number>,<length>

パラメータ

<file_name>	波形ファイル名
<group_number>	グループ番号
範囲	波形ファイルで定義されたグループ番号の範囲
初期値	1
<length>	連続再生する波形パターンの数
範囲	1～波形ファイルに含まれる波形パターン数
初期値	1

使用例

波形ファイル"MV887012A_GSM_0002.xml"のグループ番号 2 の波形パターンを再生パターンに設定します。

:SOUR:GPRF:GEN:ARB:WAV:PATT:SEL "MV887012A_GSM_0002",2,1

現在，再生しているパターンの情報を問い合わせます。

:SOUR:GPRF:GEN:ARB:WAV:PATT:SEL?

> "MV887012A_GSM_0002",2,1

備考

ファイル名は拡張子（xml）を除いた文字列です。

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

波形パターンが未選択の場合，レスポンスは""NO FILE",0,0"となります。

指定したグループ番号が存在しない場合，パラメータエラーとなります。

:SOURce:GPRF:GENerator:ARB:WAVeform:RESTart

ARB Restart

機能

選択しているパターンを先頭から再生します。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:RESTart
```

使用例

選択しているパターンを先頭から再生します。

:SOUR:GPRF:GEN:ARB:WAV:REST

備考

動作モードが以下の状態の場合，実行エラーとなります。

- シーケンスモード

以下の処理が実行されている場合，実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:WAVeform:SCLock:RATE?

ARB Sampling Rate

機能

選択中のパターンのサンプリングレートを問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SCLock:RATE?
```

レスポンス

<sampling_rate>

単位	Hz

パラメータ

<sampling_rate>	サンプリングレート
分解能	0.001 Hz

使用例

選択中のパターンのサンプリングレートを問い合わせます。

:SOUR:GPRF:GEN:ARB:WAV:SCL:RATE?

> 3250000.000

備考

パターンが選択されていない場合のレスポンスは，0.000 Hz になります。

以下の処理が実行されている場合，レスポンスは"–999.999"となります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:WAVeform:SSWitch

ARB Subsection Switch

機能

波形パターンのサブセクションの選択方法を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SSWitch
<switch>[,<subsection_number>]
```

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SSWitch?
```

レスポンス

<switch>,<subsection_number>

パラメータ

<switch>	サブセクションの選択方法
FIXED	固定
ALTERNATIVE	サブセクション 0 と 1 を交互
ILPCONTROL	Inner Loop Power Control を使用
初期値	FIXED
<subsection_number>	サブセクション番号
範囲	0～3
分解能	1
初期値	0

使用例

波形パターンのサブセクションの選択方法を 固定 に設定し，サブセクション番号を 2 とします。
:SOUR:GPRF:GEN:ARB:WAV:SSW FIXED,2

波形パターンのサブセクションの選択方法を問い合わせます。
:SOUR:GPRF:GEN:ARB:WAV:SSW?
> FIXED,2

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:CANCel

Waveform Synchronization Cancel

機能

波形パターンの同期出力開始待ち状態を解除します。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:CANCel
```

使用例

波形パターンの同期出力開始待ち状態を解除します。

:SOUR:GPRF:GEN:ARB:WAV:SYNC:STAR:CANC

:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect

Waveform Synchronization Select Pattern

機能

波形メモリにロードされている波形ファイルから波形パターンを選択し，同期方法を設定してパターン同期を開始します。また，波形パターンを問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect
<mode>,<filename>[,<group_number>[,<length>]]
```

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect?
```

レスポンス

<filename>,<group_number>,<length>

パラメータ

<mode>	同期モード
PRIMARY	プライマリとしてパターン同期出力を開始します (スロット 1 のモジュールのみ設定できます)。
SECONDARY_INT	セカンダリとしてパターン同期出力を準備します (MT8870A の信号に同期します)。
SECONDARY_EXT	セカンダリとしてパターン同期出力を準備します (外部トリガに同期します)。
<file_name>	波形ファイル名
<group_number>	グループ番号
範囲	波形ファイルで定義されたグループ番号の範囲
初期値	1
<length>	連続再生する波形パターンの数
範囲	1～波形ファイルに含まれる波形パターン数
初期値	1

使用例

同期モードをプライマリにして，波形ファイル"MV887012A_GSM_0002.xml"の波形パターンを再生パターンに設定します。

:SOUR:GPRF:GEN:ARB:WAV:SYNC:STAR:PATT:SEL PRIMARY,"MV887012A_GSM_0002"

現在，再生しているパターンの情報を問い合わせます。
:SOUR:GPRF:GEN:ARB:WAV:SYNC:STAR:PATT:SEL?
> "MV887012A_GSM_0002",1,1

備考

モジュールが挿入されているスロット番号により，設定できる同期モードが異なります。

スロット番号 / 同期モード	Slot1	Slot2	Slot3	Slot4
PRIMARY	○	×	×	×
SECONDARY_INT	×	○	○	○
SECONDARY_EXT	○	○	○	○

○:設定可能　×:設定不可

本コマンドを実行すると，以下のコマンドで設定した波形パターンが変更されます。
:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect

本コマンドを実行すると，外部トリガに関連する下記コマンドの設定値が変更されます。
:ROUTe:TRIGger:PORT
:ROUTe:TRIGger:OUTPut:SOURce
:ROUTe:TRIGger:INPut:SOURce

<mode>の設定により，モジュールの外部トリガ設定は以下のように変わります。

<mode> / コマンド	PRIMARY	SECONDARY_INT	SECONDARY_EXT
:ROUTe:TRIGger:PORT	OUTPUT	INPUT	INPUT*
:ROUTe:TRIGger:OUTPut:SOURce	SGSYNC	(変更なし)	(変更なし)
:ROUTe:TRIGger:INPut:SOURce	(変更なし)	SLOT1	TRGIN1

＊:　SECONDARY_EXT を指定すると，スロット 1 の外部トリガが INPUT に変わります。

関連コマンド

:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect
:SOURce:GPRF:GENerator:ARB:WAVeform:RESTart
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:STATe
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:CANCel

:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:STATe?

Waveform Synchronization State

機能

波形パターンの同期出力開始待ち状態を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:STATe?
```

レスポンス

<status>

パラメータ

<status>	同期待ち状態
0	同期出力開始待ち状態
1	同期出力開始待ち解除状態

使用例

波形パターンの同期出力開始待ち状態を問い合わせます。

```
:SOUR:GPRF:GEN:ARB:WAV:SYNC:STAT:STAT?
> 1
```

:SOURce:GPRF:GENerator:BBMode

Modulation

機能

ベースバンドモードを変調波出力，または CW 出力に設定します。

コマンド

```
:SOURce:GPRF:GENerator:BBMode <mode>
```

クエリ

```
:SOURce:GPRF:GENerator:BBMode?
```

レスポンス

<mode>

パラメータ

<mode>		ベースバンドモード
	CW	CW 出力にします。
	ARB	ARB 波形による変調波出力にします。
	初期値	CW

使用例

ベースバンドモードの設定を変調波出力にします。

:SOUR:GPRF:GEN:BBM ARB

ベースバンドモードを問い合わせます。

:SOUR:GPRF:GEN:BBM?

> ARB

備考

次のコマンドで波形ファイルが選択されていない場合，ARB を設定してもパネルのテストポートから信号が出力されません。

:SOURce:GPRF:GENerator:ARB:WAVeform:PATTern:SELect

:SOURce:GPRF:GENerator:ARB:WAVeform:SYNChronize:STARt:PATTern:SELect

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:MODE

VSG Mode

機能

VSG の動作モードを設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:MODE <switch>
```

クエリ

```
:SOURce:GPRF:GENerator:MODE?
```

レスポンス

<switch>

パラメータ

<switch>	動作モード
NORMAL	ノーマルモード
SEQUENCE	シーケンスモード
初期値	NORMAL

使用例

VSG の動作モードを ノーマルモード に設定します。
:SOUR:GPRF:GEN:MODE NORMAL

VSG の動作モードを問い合わせます。
:SOUR:GPRF:GEN:MODE?
> NORMAL

備考

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:RFSettings:DM:POLarity

Spectrum Reverse

機能

出力波形の IQ 反転を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:RFSettings:DM:POLarity <mode>
```

クエリ

```
:SOURce:GPRF:GENerator:RFSettings:DM:POLarity?
```

レスポンス

<mode>

パラメータ

<mode>	モード
NORMAL	IQ を反転しません。
INVERT	IQ を反転します。
初期値	NORMAL

使用例

出力波形の IQ を反転状態にします。
:SOUR:GPRF:GEN:RFS:DM:POL INVERT

出力波形の IQ 反転状態を問い合わせます。
:SOUR:GPRF:GEN:RFS:DM:POL?
> INVERT

備考

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:RFSettings:FREQuency

Frequency

機能

RF 周波数を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:RFSettings:FREQuency <freq>
```

クエリ

```
:SOURce:GPRF:GENerator:RFSettings:FREQuency?
```

レスポンス

<freq>

単位　Hz

パラメータ

<freq>	周波数
範囲	400 MHz～6.0 GHz 400 MHz～6.0 GHz (MU887002A-007 搭載時 Test Port 1～4) 400 MHz～7.3 GHz (MU887002A-007 搭載時 Test Port 5～12)
分解能	1 Hz
サフィックスコード	HZ, Z, KHZ, KZ, MHZ, MZ, GHZ, GZ 省略した場合は HZ として扱われます。
初期値	1 GHz

詳細

MU887002A-007 搭載時，以下の制限事項があります。設定コマンドは有効ですが実行エラーとなります。

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では，入力または出力のどちらかにしか設定できません。

使用例

RF 周波数を 3 GHz に設定します。

```
:SOUR:GPRF:GEN:RFS:FREQ 3GHZ
```

RF 周波数の設定値を問い合わせます。

```
:SOUR:GPRF:GEN:RFS:FREQ?
> 3000000000
```

備考

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:RFSettings:LEVel

Level

機能

RF 出力レベルを設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:RFSettings:LEVel <level>
```

クエリ

```
:SOURce:GPRF:GENerator:RFSettings:LEVel?
```

レスポンス

<level>

単位　　dBm

パラメータ

<level>　　出力レベル

範囲	(–130.0 dBm – CableLoss)～(0.0 dBm – CableLoss – AWGN_RF_Adjust_Gain)
分解能	0.1 dBm
サフィックスコード	DBM, DBUVEMF, DBUVTERM，省略した場合は DBM として扱われます。
初期値	–120.0 dBm

詳細

CableLoss

External Loss On/Off の設定により下記となります。

設定	COMMON，ON	損失値 [dB] (≧0) Register Loss Value into Cable Loss List で設定された値を補間した値
	OFF	0 [dB]

AWGN_RF_Adjust_Gain

AWGN Switch の設定により下記となります。

設定	ON	調整値 [dB] (≧0) AWGN C/N Ratio で設定された C/N 値と，再生中の波形に依存する値
	OFF	0 [dB]

使用例

RF 出力レベルを–50.0 dBm に設定します。

```
:SOUR:GPRF:GEN:RFS:LEV -50.0DBM
```

RF 出力レベルの設定値を問い合わせます。

```
:SOUR:GPRF:GEN:RFS:LEV?
> -50.0
```

:SOURce:GPRF:GENerator:RFSettings:LEVel:SETTing?

Level Setting

機能

ノーマルモードで使用している時の出力レベルが確度保証範囲内かどうかを問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:RFSettings:LEVel:SETTing?
```

レスポンス

<status>

パラメータ

<status>	出力レベルの状態
NORMAL	通常状態，または波形を再生していない状態
UNLEVELD	レベル確度保証外

使用例

ノーマルモードで使用している時の出力レベルの状態を問い合わせます。

:SOUR:GPRF:GEN:RFS:LEV:SETT?

> NORMAL

:SOURce:GPRF:GENerator:SEQuence:CANCel

Sequence Execution Cancel

機能

シーケンス処理を中止します。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:CANCel
```

使用例

シーケンス処理を中止します。

:SOUR:GPRF:GEN:SEQ:CANC

備考

動作モードが以下の状態の場合，実行エラーとなります。

- ノーマルモード

:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern

Sequence Table Combination Selection

機能

実行するシーケンステーブルの数とシーケンステーブルの実行順序を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern <num>,<table_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern?
```

レスポンス

<num>,<table_n>

パラメータ

<num>	実行するシーケンステーブル数
範囲	1～16
初期値	1
<table_n>	シーケンステーブル番号
<table1>	1 番目に実行するシーケンステーブル番号
<table2>	2 番目に実行するシーケンステーブル番号
….	
<table16>	16 番目に実行するシーケンステーブル番号
範囲	1～16
初期値	1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16

使用例

実行するシーケンステーブルの数を 3, シーケンステーブルの実行順序を 1,2,3 に設定します。
:SOUR:GPRF:GEN:SEQ:COMB:PATT 3,1,2,3

実行するシーケンステーブルの数とシーケンステーブルの実行順序を問い合わせます。
:SOUR:GPRF:GEN:SEQ:COMB:PATT?
> 3,1,2,3

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:SYNChronize

Sequence Synchronization

機能

シーケンス実行時の波形リストテーブルの開始タイミングを設定します。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern:SYNChronize <mode>
```

パラメータ

<mode>		同期モード
	NONE	他のモジュールと同期しません。
	PRIMARY	プライマリとして波形リストテーブルを開始します (スロット 1 に装着されているモジュールのみ設定できます)。
	SECONDARY_INT	セカンダリとして波形リストテーブルを開始します (MT8870A の信号に同期します)。
	SECONDARY_EXT	セカンダリとして波形リストテーブルを開始します (外部トリガに同期します)。
	初期値	NONE

使用例

シーケンステーブル実行時に，波形リストテーブルの開始タイミングを外部トリガに設定します。

:SOUR:GPRF:GEN:SEQ:COMB:PATT:SYNC SECONDARY_EXT

備考

モジュールが挿入されているスロット番号により，設定できる同期モードが異なります。

スロット番号 / 同期モード	Slot1	Slot2	Slot3	Slot4
NONE	○	○	○	○
PRIMARY	○	×	×	×
SECONDARY_INT	×	○	○	○
SECONDARY_EXT	○	○	○	○

○:設定可能　×:設定不可

本コマンドを実行すると，外部トリガに関連する下記コマンドの設定値が変更されます。

:ROUTe:TRIGger:PORT
:ROUTe:TRIGger:OUTPut:SOURce
:ROUTe:TRIGger:INPut:SOURce

<mode>の設定により，モジュールの外部トリガ設定は以下のように変わります。

コマンド / <mode>	NONE	PRIMARY	SECONDARY_INT	SECONDARY_EXT
:ROUTe:TRIGger:PORT	(変更なし)	OUTPUT	INPUT	INPUT*
:ROUTe:TRIGger:OUTPut:SOURce	(変更なし)	SGSYNC	(変更なし)	(変更なし)
:ROUTe:TRIGger:INPut:SOURce	(変更なし)	(変更なし)	SLOT1	TRGIN1

＊:　SECONDARY_EXT を指定すると，スロット 1 の外部トリガが INPUT に変わります。

:SOURce:GPRF:GENerator:SEQuence:EXECute

Sequence Execution

機能

実行順が 1 番目の SG シーケンサデータからシーケンスの実行を開始します。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:EXECute [<mode>]
```

パラメータ

<mode>	シーケンスの実行モード
NORMAL	通常実行モード
FORCED	強制実行モード
	省略した場合は NORMAL として扱われます。

詳細

通常実行モードは，SG シーケンサデータにエラーがない場合にシーケンスの実行を開始します。
強制実行モードは，SG シーケンサデータのレベル設定値が範囲外の場合でも，シーケンスの実行を開始します。この場合，レベル設定値が上限値を超えるときは出力できる上限のレベルで，レベル設定値が下限値未満のときは出力できる下限のレベルで信号を出力します。
レベル設定値が範囲外の場合は，:SOURce:GPRF:GENerator:SEQuence:RX:ERRor?のレスポンスが 8 となるときです。レベルのほかにもエラーがある場合は，強制実行モードでもシーケンスの実行を開始しません。

使用例

実行順が 1 番目の SG シーケンサデータからシーケンスの実行を開始します。
:SOUR:GPRF:GEN:SEQ:EXEC

関連コマンド

:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern

備考

シーケンス動作中に本コマンドを送信しても，シーケンスは再スタートしません。この場合，本コマンドは無視されます。
実行する SG シーケンサデータの数と順番は，
:SOURce:GPRF:GENerator:SEQuence:COMBination:PATTern で設定します。
シーケンス動作中にセグメント番号 1 から再スタートする場合は，
:SOURce:GPRF:GENerator:SEQuence:RLISt を使用します。
SG シーケンサデータにエラーがない場合，通常実行モードと強制実行モードの動作は同じです。
レベル設定値が範囲外のエラーがある場合に通強制実行モードでシーケンスを実行すると，コマンドは実行エラーとなります。

動作モードが以下の状態の場合，実行エラーとなります。
・ ノーマルモード

:SOURce:GPRF:GENerator:SEQuence:LOAD

Load Sequence Parameter

機能

SG シーケンサデータをファイルからロードして，指定したシーケンステーブルに展開します。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:LOAD <list_table_no>,<file_name>
```

パラメータ

<list_table_no>	SG シーケンサデータのロード先シーケンステーブルの番号
範囲	1～16
<file_name>	SG シーケンサデータのファイル名

使用例

ファイル名"SeqFile.xml"の SG シーケンサデータを，番号 1 のシーケンステーブルにロードします。

:SOUR:GPRF:GEN:SEQ:LOAD 1,"SeqFile.xml"

備考

ファイル名の拡張子（xml）は省略可能です。

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

指定したファイルが存在しない場合，パラメータエラーとなります。

:SOURce:GPRF:GENerator:SEQuence:REINitialization

Sequence Reinitialization

機能

シーケンス実行が終了した後の動作モードを設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:REINitialization <on_off>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:REINitialization?
```

レスポンス

<on_off>

パラメータ

<on_off>	シーケンス実行後の状態
0	シーケンスモードを継続します。
1	ノーマルモードに変更します。
OFF	シーケンスモードを継続します。
ON	ノーマルモードに変更します。
初期値	0

使用例

シーケンスの実行後に動作モードをノーマルモードに変更します。

:SOUR:GPRF:GEN:SEQ:REIN ON

シーケンスを実行した後の動作モード設定を問い合わせます。

:SOUR:GPRF:GEN:SEQ:REIN ON?

> 1

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RLISt

Sequence Re-Execution

機能

シーケンス動作を停止して，実行順序 1 番目の SG シーケンサデータからシーケンスの実行を再開します。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RLISt
```

使用例

シーケンスの実行を再開します。
:SOUR:GPRF:GEN:SEQ:RLIS

備考

シーケンス動作中に本コマンドを設定すると，シーケンスモードのシーケンス動作を停止します。その後，:SOURce:GPRF:GENerator:SEQuence:EXECute で指定した実行モードで，シーケンスの先頭からシーケンスの実行を再開します。

動作モードが以下の状態の場合，実行エラーとなります。

- ノーマルモード

:SOURce:GPRF:GENerator:SEQuence:RX:BBMode

Base Band Mode of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントのベースバンドモードを設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:BBMode
<list_table_no>,<segment_no>,<mode>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:BBMode? <list_table_no>,<segment_no>
```

レスポンス

<mode>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<mode>	ベースバンドモード
CW	CW 出力にします。
ARB	ARB 波形による変調波出力にします。
初期値	CW

使用例

シーケンステーブル番号 1，セグメント 10 のベースバンドモードを CW に設定します。

:SOUR:GPRF:GEN:SEQ:RX:BBM 1,10,CW

シーケンステーブル番号 1，セグメント 10 のベースバンドモードを問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:BBM? 1,10

> CW

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:BBMode:ALL

Base Band Mode Group of All Segments in Sequence

機能

SG シーケンサデータの各セグメントのベースバンドモードを一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:BBMode:ALL <list_table_no>,<mode_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:BBMode:ALL? <list_table_no>
```

レスポンス

<mode_n>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<mode_n>	セグメント番号 n のベースバンドモード
<mode_1>	セグメント 1 のベースバンドモード
<mode_2>	セグメント 2 のベースバンドモード
…	
<mode_10001>	セグメント 10001 のベースバンドモード
CW	CW 出力にします。
ARB	ARB 波形出力にします。
初期値	CW

詳細

コマンドのパラメータ<mode_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までのベースバンドモードが設定されます。この場合，セグメント 11 以降のベースバンドモードの設定は変わりません。

使用例

シーケンステーブル番号 1 の各セグメントのベースバンドモードを一括して設定します。

:SOUR:GPRF:GEN:SEQ:RX:BBM:ALL 1,CW,ARB,ARB,CW

シーケンステーブル番号 1 の各セグメントのベースバンドモードを問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:BBM:ALL? 1

> CW,ARB,ARB,CW,CW,...,CW

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

・波形ロード

:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition

End Condition of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントの終了条件を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition
<list_table_no>,<segment_no>,<condition>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition?
<list_table_no>,<segment_no>
```

レスポンス

<condition>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<condition>	終了条件
SNUMBER	設定されたステップ数の時間が経過すると終了します。
TRIGGER	トリガを受信すると終了します。
初期値	SNUMBER

使用例

シーケンステーブル番号 1，セグメント 10 の終了条件を ステップ数 にします。
:SOUR:GPRF:GEN:SEQ:RX:ENDC 1,10,SNUMBER

シーケンステーブル番号 1，セグメント 10 の終了条件を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:ENDC? 1,10
> SNUMBER

関連コマンド

:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition:ALL

End Condition Group of All Segments in Sequence

機能

SG シーケンサデータの各セグメントの終了条件を一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition:ALL
<list_table_no>,<condition_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:ENDCondition:ALL? <list_table_no>
```

レスポンス

<condition_n>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<condition_n>	セグメント番号 n の終了条件
<condition_1>	セグメント 1 の終了条件
<condition_2>	セグメント 2 の終了条件
….	
<condition_10001>	セグメント 10001 の終了条件
SNUMBER	ステップ数の時間経過を終了条件とします。
TRIGGER	トリガ発生を終了条件とします。
初期値	SNUMBER

詳細

コマンドのパラメータ<condition_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までの終了条件が設定されます。この場合，セグメント 11 以降の終了条件の設定は変わりません。

使用例

シーケンステーブル番号 1 の各セグメントの終了条件を一括して設定します。
:SOUR:GPRF:GEN:SEQ:RX:ENDC:ALL 1,SNUMBER,SNUMBER,TRIGGER
シーケンステーブル番号 1 の各セグメントの終了条件を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:ENDC:ALL? 1
> SNUMBER,SNUMBER,TRIGGER,SNUMBER,...,SNUMBER

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:ERRor?

Sequence Error Check

機能

シーケンス実行時に動作を保証できない設定（エラー）が SG シーケンサデータに有るか，問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:ERRor? <list_table_no>
```

レスポンス

<error_count>,<error_x_res>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<error_count>	エラーを判定したセグメント数
<error_x_res>	各セグメントの判定結果
<error_n_res>	シーケンスをスタートするセグメント番号の判定結果
<error_(n+1)_res>	
…	
<error_m_res>	シーケンスを終了するセグメント番号の判定結果
	判定結果は，2 進数の足し算でエラーを表現します（ビット表示)。
1	Loop のネストを発生させています。
2	Loop のジャンプ先が実行シーケンスの範囲外です。
4	トリガ待ちのセグメントで Loop を設定しています。
8	レベルが設定範囲外の値です。
16	使用できない波形パターンを設定しています。

使用例

シーケンステーブル番号 1 のエラーチェック結果を問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:ERR? 1

> 2,0,0,1,3,...0

:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency

Frequency of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントの周波数を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency
<list_table_no>,<segment_no>,<freq>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency?
<list_table_no>,<segment_no>
```

レスポンス

<freq>

単位	Hz

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<freq>	周波数
範囲	400 MHz～6.0 GHz 400 MHz～6.0 GHz (MU887002A-007 搭載時 Test Port 1～4) 400 MHz～7.3 GHz (MU887002A-007 搭載時 Test Port 5～12)
分解能	1 Hz
サフィックスコード	HZ, Z, KHZ, KZ, MHZ, MZ, GHZ, GZ 省略した場合は HZ として扱われます。
初期値	1 GHz

詳細

MU887002A-007 搭載時，以下の制限事項があります。設定コマンドは有効ですが実行エラーとなります。

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では，入力または出力のどちらかにしか設定できません。

使用例

シーケンステーブル番号 1，セグメント 10 の周波数の設定を 3 GHz にします。

:SOUR:GPRF:GEN:SEQ:RX:FREQ 1,10,3GHZ

シーケンステーブル番号 1，セグメント 10 の周波数の設定値を問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:FREQ? 1,10

> 3000000000

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency:ALL

Frequency Group of All Segments in Sequence

機能

SG シーケンサデータの各セグメントの周波数を一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency:ALL <list_table_no>,<freq_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:FREQuency:ALL? <list_table_no>
```

レスポンス

<freq_n>

単位	Hz

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<freq_n>	セグメント番号 n の周波数
<freq_1>	セグメント 1 の周波数
<freq_2>	セグメント 2 の周波数
…	
<freq_10001>	セグメント 10001 の周波数
範囲	400 MHz～6.0 GHz 400 MHz～6.0 GHz (MU887002A-007 搭載時 Test Port 1～4) 400 MHz～7.3 GHz (MU887002A-007 搭載時 Test Port 5～12)
分解能	1 Hz
サフィックス	HZ, Z, KHZ, KZ, MHZ, MZ, GHZ, GZ, 省略した場合は HZ として扱われます。
初期値	1 GHz

詳細

コマンドのパラメータ<freq_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までの周波数が設定されます。この場合，セグメント 11 以降の周波数の設定は変わりません。

MU887002A-007 搭載時，以下の制限事項があります。設定コマンドは有効ですが実行エラーとなります。

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では，入力または出力のどちらかにしか設定できません。

使用例

シーケンステーブル番号 1 の各セグメントの周波数列を一括して設定します。

```
:SOUR:GPRF:GEN:SEQ:RX:FREQ:ALL 1,3GHZ,3.5GHZ,3.8GHZ
```

シーケンステーブル番号 1 の各セグメントの周波数列を問い合わせます。

```
:SOUR:GPRF:GEN:SEQ:RX:FREQ:ALL? 1
> 3000000000,3500000000,3800000000,1000000000,1000000000,...,1000000000
```

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:GOTO

Sequence Start Index for continuous mode

機能

シーケンス繰り返し時（Continuous モード時）の 2 回目以降の開始セグメント番号を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:GOTO <list_table_no>,<index>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:GOTO? <list_table_no>
```

レスポンス

<index>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<index>	セグメント番号
範囲	1～10001
初期値	1

使用例

シーケンステーブル番号 1 の 2 回目以降の開始セグメントを 10 に設定します。
:SOUR:GPRF:GEN:SEQ:RX:GEN:GOTO 1,10

シーケンステーブル番号 1 の 2 回目以降の開始セグメントを問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:GEN:GOTO? 1
> 10

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:REPetition

Sequence Repetition Mode

機能

シーケンスの繰り返し方法を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:REPetition
<list_table_no>,<mode>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:REPetition? <list_table_no>
```

レスポンス

<mode>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<mode>	シーケンスの繰り返し方法
CONTINUOUS	シーケンスを繰り返します。
SINGLE	1 回のみとします。
初期値	CONTINUOUS

使用例

シーケンステーブル番号 1 のシーケンス繰り返し方法を シーケンスを繰り返す に設定します。

:SOUR:GPRF:GEN:SEQ:RX:GEN:REP 1,CONTINUOUS

シーケンステーブル番号 1 のシーケンスの繰り返し方法を問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:GEN:REP? 1

> CONTINUOUS

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:SSTop

Sequence Start and Stop Segment

機能

シーケンスの開始セグメント番号と終了セグメント番号の両方を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:SSTop
<list_table_no>,<start_seg>,<stop_seg>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:SSTop? <list_table_no>
```

レスポンス

<start_seg>,<stop_seg>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<start_seg>	開始セグメント番号
範囲	1～10001
分解能	1
初期値	1
<stop_seg>	終了セグメント番号
範囲	<start_seg>～10001
分解能	1
初期値	1

使用例

シーケンステーブル番号 1 の開始セグメントを 10 に，終了セグメントを 30 に設定します。

:SOUR:GPRF:GEN:SEQ:RX:GEN:SST 1,10,30

シーケンステーブル番号 1 の開始セグメントと終了セグメントを問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:GEN:SST? 1

> 10,30

備考

終了セグメント番号が開始セグメント番号より小さい場合は，パラメータエラーとなります。

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STARt

Sequence Start Segment

機能

シーケンスの開始セグメントの番号を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STARt
<list_table_no>,<start_seg>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STARt? <list_table_no>
```

レスポンス

<start_seg>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<start_seg>	開始セグメント番号
範囲	1～終了セグメント番号
分解能	1
初期値	1

使用例

シーケンステーブル番号 1 の開始セグメントを 10 に設定します。

:SOUR:GPRF:GEN:SEQ:RX:GEN:STAR 1,10

シーケンステーブル番号 1 の開始セグメントを問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:GEN:STAR? 1

> 10

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STEP:LENGth

Sequence Step Length

機能

ステップ長の時間を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STEP:LENGth
<list_table_no>,<mode>[,<time>]
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STEP:LENGth? <list_table_no>
```

レスポンス

<mode>,<time>

<time>の単位　　ms

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<mode>	ステップ長設定モード
TIME	時間による設定。ステップ長の時間にはパラメータ<time>を設定します。
GSLOT	GSM のスロット時間（15/26 ms）
GFRAME	GSM のフレーム時間（60/13 ms）
WSLOT	WCDMA のスロット時間（10/15 ms）
ESLOT	EVDO のスロット時間（80/48 ms）
EFRAME	EVDO のフレーム時間（80/3 ms）
初期値	TIME
<time>	ステップ長
範囲	0.5～80 ms
分解能	0.00001 ms
サフィックスコード	S, MS, US, NS 省略した場合は MS として扱われます。
初期値	10 ms

使用例

シーケンステーブル番号 1 のステップ長を 1.00000 ms に設定します。
:SOUR:GPRF:GEN:SEQ:RX:GEN:STEP:LENG 1,TIME,1.00000

シーケンステーブル番号 1 のステップ長を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:GEN:STEP:LENG? 1
> TIME,1.00000

備考

SG シーケンサデータ内の共通設定です。SG シーケンサデータの全セグメントに対して本コマンドで設定した時間が適用されます。

<mode>が TIME 以外の場合は，<time>パラメータの設定を省略可能とします。
<mode>が TIME 以外の場合に，<time>パラメータが設定された場合は，設定値を無視します。
<mode>を TIME 以外に設定した場合，ステップ長は以下となります。

GSLOT: 0.57692 [ms] (= 15/26 [ms])
GFRAME: 4.61538 [ms] (= 60/13 [ms])
WSLOT: 0.66667 [ms] (= 10/15 [ms])
ESLOT: 1.66667 [ms] (= 80/48 [ms])
EFRAME: 26.66667 [ms] (= 80/3 [ms])

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STOP

Sequence Stop Segment

機能

シーケンスの終了セグメントの番号を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STOP
<list_table_no>,<stop_seg>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STOP? <list_table_no>
```

レスポンス

<stop_seg>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<stop_seg>	終了セグメント番号
範囲	開始セグメント番号～10001
初期値	1

使用例

シーケンステーブル番号 1 の終了セグメントを 30 に設定します。
:SOUR:GPRF:GEN:SEQ:RX:GEN:STOP 1,30

シーケンステーブル番号 1 の終了セグメントを問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:GEN:STOP? 1
> 30

備考

終了セグメント番号が開始セグメント番号より小さい場合は，パラメータエラーとなります。

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:LEVel

Level of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントのレベルを設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:LEVel
<list_table_no>,<segment_no>,<level>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:LEVel? <list_table_no>,<segment_no>
```

レスポンス

<level>

単位	dBm

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<level>	レベル
範囲	–230.0～100.0 dBm*
分解能	0.1 dB
サフィックスコード	DBM 省略した場合は DBM として扱われます。
初期値	–120.0 dBm

＊： テストポート，ケーブルロス値の組み合わせで設定可能な範囲

使用例

シーケンステーブル 1，セグメント 10 の出力レベルを –50.0 dBm に設定します。

```
:SOUR:GPRF:GEN:SEQ:RX:LEV 1,10,-50.0DBM
```

シーケンステーブル 1，セグメント 10 の出力レベルの設定値を問い合わせます。

```
:SOUR:GPRF:GEN:SEQ:RX:LEV? 1,10
> -50.0
```

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:LEVel:ALL

Level Group of All Segments in Sequence

機能

SG シーケンサデータの各セグメントのレベルを一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:LEVel:ALL <list_table_no>,<level_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:LEVel:ALL? <list_table_no>
```

レスポンス

<level_n>
　単位　　dBm

パラメータ

<list_table_no>　　シーケンステーブル番号
　範囲　　1～16
<level_n>　　セグメント番号 n のレベル
　<level_1>　　セグメント 1 のレベル
　<level_2>　　セグメント 2 のレベル
　….
　<level_10001>　　セグメント 10001 のレベル
　　範囲　　–230.0～100.0 dBm*
　　分解能　　0.1 dB
　　サフィックスコード　DBM，省略した場合は DBM として扱われます。
　　初期値　　–120.0 dBm

＊：　テストポート，ケーブルロス値の組み合わせで設定可能な範囲

詳細

コマンドのパラメータ<level_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までのレベルが設定されます。この場合，セグメント 11 以降のレベルの設定は変わりません。

使用例

シーケンステーブル 1 の各セグメントのレベルを一括して設定します。
:SOUR:GPRF:GEN:SEQ:RX:LEV:ALL 1,-50.0DBM,-50.0DBM,-50.0DBM
シーケンステーブル 1 の各セグメントのレベル設定を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:LEV:ALL? 1
> -50.0,-50.0,-50.0,-120.0,-120.0,...,-120.0

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl

Loop Control of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントの繰り返し制御（Loop）を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl
<list_table_no>,<segment_no>,<control>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl?
<list_table_no>,<segment_no>
```

レスポンス

<control>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<control>	繰り返し制御
LOOP	繰り返し制御をします。
NSEGMENT	繰り返し制御を行わず，次のセグメントを実行します。
初期値	NSEGMENT

使用例

シーケンステーブル番号 1，セグメント 10 の繰り返し制御を Loop 制御に設定します。
:SOUR:GPRF:GEN:SEQ:RX:NSLC 1,10,LOOP

シーケンステーブル番号 1，セグメント 10 の繰り返し制御の設定を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:NSLC? 1,10
> LOOP

関連コマンド

:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt
:SOURce:GPRF:GENerator:SEQuence:RX:ERRor?

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:ALL

Loop Control Group of All Segment in Sequence

機能

SG シーケンサデータの各セグメントの繰り返し制御（Loop）を一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:ALL
<list_table_no>,<control_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:ALL? <list_table_no>
```

レスポンス

<control_n>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<control_n>	セグメント番号 n の繰り返し制御
<control_1>	セグメント 1 の繰り返し制御
<control_2>	セグメント 2 の繰り返し制御
….	
<control_10001>	セグメント 10001 の繰り返し制御
LOOP	繰り返し制御をします。
NSEGMENT	繰り返し制御を行わず，次のセグメントを実行します。
初期値	NSEGMENT

詳細

コマンドのパラメータ<control_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までの繰り返し制御が設定されます。この場合，セグメント 11 以降の繰り返し制御の設定は変わりません。

使用例

シーケンステーブル番号 1 の各セグメントの繰り返し制御を一括して設定します。
:SOUR:GPRF:GEN:SEQ:RX:NSLC:ALL 1,LOOP,LOOP,NSEGMENT,LOOP
シーケンステーブル番号 1 の各セグメントの繰り返し制御を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:NSLC:ALL? 1
>LOOP,LOOP,NSEGMENT,LOOP,NSEGMENT,NSEGMENT,...,NSEGMENT

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt

Loop Count of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントの繰り返し制御（Loop）時の繰り返し回数を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt
<list_table_no>,<segment_no>,<count>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt?
<list_table_no>,<segment_no>
```

レスポンス

<count>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<count>	繰り返し回数
範囲	0～100*
初期値	0

＊： 繰り返し回数が 0 の場合，繰り返し制御は行いません。

使用例

シーケンステーブル番号 1，セグメント 10 の繰り返し回数を 2 に設定します。

:SOUR:GPRF:GEN:SEQ:RX:NSLC:COUN 1,10,2

シーケンステーブル番号 1，セグメント 10 の繰り返し回数を問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:NSLC:COUN? 1,10

> 2

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt:ALL

Loop Count Group of All Segments in Sequence

機能

SG シーケンサデータの各セグメントの繰り返し制御（Loop）時の繰り返し回数を一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt:ALL
<list_table_no>,<count_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:COUNt:ALL? <list_table_no>
```

レスポンス

<count_n>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<count_n>	セグメント番号 n の繰り返し回数
<count_1>	セグメント 1 の繰り返し回数
<count_2>	セグメント 2 の繰り返し回数
….	
<count_10001>	セグメント 10001 の繰り返し回数
範囲	0～100*
初期値	0

＊:　繰り返し回数が 0 の場合，繰り返し制御は行いません。

詳細

コマンドのパラメータ<count_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までの繰り返し回数が設定されます。この場合，セグメント 11 以降の繰り返し回数の設定は変わりません。

使用例

シーケンステーブル番号 1 の各セグメントの繰り返し回数を一括して設定します。

:SOUR:GPRF:GEN:SEQ:RX:NSLC:COUN:ALL 1,2,2,2

シーケンステーブル番号 1 の各セグメントの繰り返し回数を問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:NSLC:COUN:ALL? 1

> 2,2,2,0,0,...,0

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment

Segment Number of Segment in Sequence for Loop Control

機能

SG シーケンサデータの指定した 1 セグメントの繰り返し制御（Loop）時のジャンプ先セグメント番号を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment
<list_table_no>,<segment_no>,<jump_segment>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment?
<list_table_no>,<segment_no>
```

レスポンス

<jump_segment>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<jump_segment>	ジャンプ先セグメント番号
範囲	1～segment_no
初期値	1

詳細

ジャンプ先セグメント番号が設定するセグメント番号より大きい場合は，パラメータエラーとなります。

使用例

シーケンステーブル番号 1，セグメント 10 の Loop 制御時のジャンプ先セグメント番号を 5 に設定します。
:SOUR:GPRF:GEN:SEQ:RX:NSLC:LSEG 1,10,5

シーケンステーブル番号 1，セグメント 10 の Loop 制御時のジャンプ先セグメント番号を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:NSLC:LSEG? 1,10
> 5

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment:ALL

Loop Jump Segment Group of All Segments in Sequence for Loop Control

機能

SG シーケンサデータの各セグメントの繰り返し制御（Loop）時のジャンプ先セグメント番号を一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment:ALL
<list_table_no>,<jump_segment_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:NSLControl:LSEGment:ALL?
<list_table_no>
```

レスポンス

<jump_segment_n>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<jump_segment_n>	セグメント番号 n のジャンプ先セグメント番号
<jump_segment_1>	セグメント 1 のジャンプ先セグメント番号
<jump_segment_2>	セグメント 2 のジャンプ先セグメント番号
….	
<jump_segment_10001>	セグメント 10001 のジャンプ先セグメント番号
範囲	1～n*
初期値	1

＊：　ジャンプ先セグメント番号が設定するセグメント番号より大きい場合は，パラメータエラーとなります。

詳細

コマンドのパラメータ<jump_segment_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までのジャンプ先セグメント番号が設定されます。この場合，セグメント 11 以降のジャンプ先セグメント番号の設定は変わりません。

使用例

シーケンステーブル番号 12 の各セグメントのジャンプ先セグメント番号を一括して設定します。

```
:SOUR:GPRF:GEN:SEQ:RX:NSLC:LSEG:ALL 12,1,1,2
```

シーケンステーブル番号 12 の各セグメントのジャンプ先セグメント番号を問い合わせます。

```
:SOUR:GPRF:GEN:SEQ:RX:NSLC:LSEG:ALL? 12
> 1,1,2,1,1,...,1
```

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe

Output Port of Segment in Sequence

機能

SG シーケンサデータの指定したセグメントの出力ポートを設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe
<list_table_no>,<segment_no>,<port>[,<port>,...,<port>]
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe? <list_table_no>,<segment_no>
```

レスポンス

<port>[,<port>,...,<port>]

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
初期値	PORT1

詳細

RF 信号を出力する正面パネルのポートを設定します。

MU887002A-007 搭載時，以下の制限事項があります。設定コマンドは有効ですが実行エラーとなります。

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では，入力または出力のどちらかにしか設定できません。

使用例

シーケンステーブル番号 1，セグメント 10 の出力ポートを PORT1 に設定します。

:SOUR:GPRF:GEN:SEQ:RX:OUTP:STAT 1,10,PORT1

シーケンステーブル番号 1，セグメント 10 の出力ポートを問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:OUTP:STAT? 1,10

> PORT1

シーケンステーブル番号 1，セグメント 10 の出力ポートを PORT1，PORT7，PORT11 に設定します。

:SOUR:GPRF:GEN:SEQ:RX:OUTP:STAT 1,10,PORT1,PORT11,PORT7

シーケンステーブル番号 1，セグメント 10 の出力ポートを問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:OUTP:STAT? 1,10

> PORT1,PORT7,PORT11

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe:ALL

Output Port Group of All Segments in Sequence

機能

SG シーケンサデータの全セグメントの出力ポートを一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe:ALL
<list_table_no>,<port_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:OUTPut:STATe:ALL? <list_table_no>
```

レスポンス

<port_n>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<port_n>	セグメント番号 n のポート設定
<port_1>	セグメント 1 のポート設定
<port_2>	セグメント 2 のポート設定
…	
<port_10001>	セグメント 10001 のポート設定
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
初期値	PORT1

詳細

コマンドのパラメータ<port_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までのポートが設定されます。この場合，セグメント 11 以降のポート設定は変わりません。
1 つのセグメントに複数のポートを設定する場合は，“&”でつなぎます。
MU887002A-007 搭載時，以下の制限事項があります。設定コマンドは有効ですが実行エラーとなります。

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では，入力または出力のどちらかにしか設定できません。

使用例

シーケンステーブル番号 10 の全セグメントの出力ポートを一括して設定します。
:SOUR:GPRF:GEN:SEQ:RX:OUTP:STAT:ALL 10,PORT2,PORT3,PORT3

シーケンステーブル番号 10 の全セグメントの出力ポートを問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:OUTP:STAT:ALL? 10
> PORT2,PORT3,PORT3,PORT1,PORT1,...,PORT1

シーケンステーブル番号 10 の全セグメントの出力ポートを一括して設定します。

設定内容	セグメント	1	2	3	4	5	6～10001
	出力 Port	2，4	3，5，11	3	4，5	11	1

:SOUR:GPRF:GEN:SEQ:RX:OUTP:STAT:ALL

10,PORT2&PORT4,PORT3&PORT5&PORT11,PORT3,PORT4&PORT5,PORT11

シーケンステーブル番号 10 の全セグメントの出力ポートを問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:OUTP:STAT:ALL? 10
> PORT2&PORT4,PORT3&PORT11&PORT5,PORT3,PORT4&PORT5,PORT11,
PORT1,PORT1,...,PORT1

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt

Steps of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントのステップ数を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt
<list_table_no>,<segment_no>,<count>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt?
<list_table_no>,<segment_no>
```

レスポンス

<count>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<count>	セグメントの継続ステップ数
範囲	1～100
初期値	1

使用例

シーケンステーブル番号 1，セグメント 10 のステップ数を 2 に設定します。

:SOUR:GPRF:GEN:SEQ:RX:STEP:COUN 1,10,2

シーケンステーブル番号 1，セグメント 10 のステップ数を問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:STEP:COUN? 1,10

> 2

関連コマンド

1 ステップあたりの時間は，:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STEP:LENGth で設定します。

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt:ALL

Steps Group of All Segments in Sequence

機能

SG シーケンサデータの各セグメントのステップ数を一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt:ALL
<list_table_no>,<count_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:STEP:COUNt:ALL? <list_table_no>
```

レスポンス

<count_n>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<count_n>	セグメント番号 n の継続ステップ数
<count_1>	セグメント 1 の継続ステップ数
<count_2>	セグメント 2 の継続ステップ数
….	
<count_10001>	セグメント 10001 の継続ステップ数
範囲	1～100
初期値	1

詳細

コマンドのパラメータ<count_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までの継続ステップ数が設定されます。この場合，セグメント 11 以降の継続ステップ数の設定は変わりません。

使用例

シーケンステーブル番号 1 の各セグメントのステップ数を一括して設定します。
:SOUR:GPRF:GEN:SEQ:RX:STEP:COUN:ALL 1,2,2,2

シーケンステーブル番号 1 の各セグメントのステップ数を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:STEP:COUN:ALL? 1
> 2,2,2,1,1,...,1

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay

Trigger Delay of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントに遅延時間を設定，または問い合わせます。
遅延時間は，SG シーケンサの遷移に使用するトリガ源のトリガ発生から，トリガを受け付けるまでの時間です。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay
<list_table_no>,<segment_no>,<delay>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay?
<list_table_no>,<segment_no>
```

レスポンス

<delay>

単位	ms

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<delay>	遅延時間
範囲	0.000～1000.000 ms
分解能	0.001 ms
サフィックスコード	S, MS, US, NS 省略した場合は MS として扱われます。
初期値	0 ms

使用例

シーケンステーブル番号 1，セグメント 10 の遅延時間を 1.000 ms に設定します。
:SOUR:GPRF:GEN:SEQ:RX:TRIG:DEL 1,10,1.000

シーケンステーブル番号 1，セグメント 10 の遅延時間を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:TRIG:DEL? 1,10
> 1.000

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay:ALL

Trigger Delay Group of All Segments in Sequence

機能

SG シーケンサデータの遅延時間を各セグメントに一括して設定，または問い合わせます。
遅延時間は，SG シーケンサの遷移に使用するトリガ源のトリガ発生から，トリガを受け付けるまでの時間です。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay:ALL
<list_table_no>,<delay_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:DELay:ALL? <list_table_no>
```

レスポンス

<delay_n>
　単位　　ms

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<delay_n>	セグメント番号 n のディレイ値
<delay_1>	セグメント 1 のディレイ値
<delay_2>	セグメント 2 のディレイ値
….	
<delay_10001>	セグメント 10001 のディレイ値
範囲	0.000～1000.000 ms
分解能	0.001 ms
サフィックスコード	S, MS, US, NS，省略した場合は MS として扱われます。
初期値	0 ms

詳細

コマンドのパラメータ<delay_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までのディレイ値が設定されます。この場合，セグメント 11 以降のディレイ値の設定は変わりません。

使用例

シーケンステーブル番号 5 の各セグメントに遅延時間を一括して設定します。
:SOUR:GPRF:GEN:SEQ:RX:TRIG:DEL:ALL 5,1.000,2.000,3.000,4.000,5.000

シーケンステーブル番号 5 の各セグメントの遅延時間を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:TRIG:DEL:ALL? 5
> 1.000,2.000,3.000,4.000,5.000,0.000,0.000,...,0.000

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce

Trigger Source of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントのトリガ源を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce
<list_table_no>,<segment_no>,<trigger>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce?
<list_table_no>,<segment_no>
```

レスポンス

<trigger>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<trigger>	トリガ源
EXTERNAL	外部トリガ
VIDEO	レベルトリガ
WFGEND	波形グループエンド
WFM1	波形マーカ 1
WFM2	波形マーカ 2
WFM3	波形マーカ 3
初期値	EXTERNAL

詳細

セグメントの終了条件がトリガの場合に使用するトリガを選択します。

外部トリガは，MT8870A 背面パネルのコネクタに入力される信号の電圧変化をトリガとします。
レベルトリガは，MT8870A へ入力される RF レベルが規定値を超えたときをトリガとします。
波形グループエンドは，再生する波形パターンが切り替わるときをトリガとします。波形パターンが切り替わるときのトリガ出力は，:SOURce:GPRF:GENerator:SEQuence:WAVeform:GETRigger で設定します。
波形マーカは，波形ファイルの IQ データに埋め込まれた信号をトリガとします。

使用例

シーケンステーブル番号 1，セグメント 10 のトリガ源を外部トリガに設定します。
:SOUR:GPRF:GEN:SEQ:RX:TRIG:SOUR 1,10,EXTERNAL

シーケンステーブル番号 1，セグメント 10 のトリガ源を問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:TRIG:SOUR? 1,10
> EXTERNAL

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
・ シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce:ALL

Trigger Source Group of All Segments in Sequence

機能

SG シーケンサデータの各セグメントのトリガ源を一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce:ALL
<list_table_no>,<trigger_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce:ALL? <list_table_no>
```

レスポンス

<trigger_n>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<trigger_n>	セグメント番号 n のトリガ源
<trigger_1>	セグメント 1 のトリガ源
<trigger_2>	セグメント 2 のトリガ源
….	
<trigger_10001>	セグメント 10001 のトリガ源
EXTERNAL	外部トリガ
VIDEO	レベルトリガ
WFGEND	波形グループエンド
WFM1	波形マーカ 1
WFM2	波形マーカ 2
WFM3	波形マーカ 3
初期値	EXTERNAL

詳細

コマンドのパラメータ<trigger_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までのトリガ源が設定されます。この場合，セグメント 11 以降のトリガ源の設定は変わりません。

使用例

シーケンステーブル番号 7 の各セグメントのトリガ源を一括して設定します。

:SOUR:GPRF:GEN:SEQ:RX:TRIG:SOUR:ALL 7,WFM3,WFM2,WFM1,WFM2,WFM3

シーケンステーブル番号 7 の各セグメントのトリガ源を問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:TRIG:SOUR:ALL? 7

> WFM3,WFM2,WFM1,WFM2,WFM3,EXTERNAL,EXTERNAL,...,EXTERNAL

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

・ シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

・ 波形ロード ・波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger

Waveform Change Trigger of Segment in Sequence

機能

SG シーケンサデータの指定した 1 セグメントの波形切り替えトリガを設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger
<list_table_no>,<segment_no>,<on_off>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger?
<list_table_no>,<segment_no>
```

レスポンス

<on_off_res>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<segment_no>	セグメント番号
範囲	1～10001
<on_off>	波形切り替えトリガ設定
0	波形切り替えトリガを発生しません。
1	次のセグメントに切り替わるときにトリガを発生します。
OFF	波形切り替えトリガを発生しません。
ON	次のセグメントに切り替わるときにトリガを発生します。
初期値	0
<on_off_res>	波形切り替えトリガ設定
0	トリガを発生しません。
1	次のセグメントに切り替わるときにトリガを発生します。

使用例

シーケンステーブル番号 1，セグメント 10 の波形切り替えトリガを 発生なし に設定します。
:SOUR:GPRF:GEN:SEQ:RX:WCTR 1,10,OFF

シーケンステーブル番号 1，セグメント 10 の波形切り替えトリガを問い合わせます。
:SOUR:GPRF:GEN:SEQ:RX:WCTR? 1,10
> 0

関連コマンド

波形切り替えトリガで変調波形を切り替えるには，
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce で波形パターンのトリガ源を SGRequest に設定します。

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
・ シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
・ 波形ロード　　・ 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger:ALL

Waveform Change Trigger Group of All Segments in Sequence

機能

SG シーケンサデータの各セグメントの波形切り替えトリガを一括して設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger:ALL
<list_table_no>,<on_off_n>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:RX:WCTRigger:ALL? <list_table_no>
```

レスポンス

<on_off_n_res>

パラメータ

<list_table_no>	シーケンステーブル番号
<on_off_n>	セグメント番号 n の波形切り替えトリガ設定
<on_off_1>	セグメント 1 の波形切り替えトリガ設定
<on_off_2>	セグメント 2 の波形切り替えトリガ設定
….	
<on_off_10001>	セグメント 10001 の波形切り替えトリガ設定
0	トリガを発生しません。
1	次のセグメントに切り替わるときにトリガを発生します。
OFF	トリガを発生しません。
ON	次のセグメントに切り替わるときにトリガを発生します。
初期値	0
<on_off_n_res>	セグメント番号 n の波形切り替えトリガ設定
<on_off_1_res>	セグメント 1 の波形切り替えトリガ設定
<on_off_2_res>	セグメント 2 の波形切り替えトリガ設定
….	
<on_off_10001_res>	セグメント 10001 の波形切り替えトリガ設定
0	トリガを発生しません。
1	次のセグメントに切り替わるときにトリガを発生します。

詳細

コマンドのパラメータ<on_off_n>は最大 10001 個設定できます。パラメータを 10 個設定した場合は，セグメント 1 からセグメント 10 までの波形切り替えトリガが設定されます。この場合，セグメント 11 以降の波形切り替えトリガの設定は変わりません。

使用例

シーケンステーブル番号 3 の各セグメントの波形切り替えトリガを一括して設定します。

:SOUR:GPRF:GEN:SEQ:RX:WCTR:ALL 3,OFF,ON,OFF,ON,ON

シーケンステーブル番号 3 の各セグメントの波形切り替えトリガを問い合わせます。

:SOUR:GPRF:GEN:SEQ:RX:WCTR:ALL? 3

> 0,1,0,1,1,0,0,...,0

関連コマンド

:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:STATus?

Sequence Status

機能

シーケンスモードのシーケンス動作状態を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:STATus?
```

レスポンス

<status>

パラメータ

<status>	シーケンスモードのシーケンス動作状態
0	停止中（Inactive）
1	実行中（Active）

使用例

シーケンスモードのシーケンス動作状態を問い合わせます。

```
:SOUR:GPRF:GEN:SEQ:STAT?
> 1
```

:SOURce:GPRF:GENerator:SEQuence:STORe

Save Sequence Parameter

機能

指定したシーケンステーブルの SG シーケンサデータをファイルに保存します。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:STORe <list_table_no>,<file_name>
```

パラメータ

<list_table_no>	SG シーケンサデータを保存するシーケンステーブルの番号
範囲	1～16
<file_name>	SG シーケンサデータの保存ファイル名

使用例

シーケンステーブル番号 1 の SG シーケンサデータを ファイル名"SeqFile.xml" で保存します。

:SOUR:GPRF:GEN:SEQ:STOR 1,"SeqFile.xml"

備考

保存ファイル名の拡張子（xml）は省略可能です。

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

同名のファイルがすでに存在する場合，実行エラーとなります。

:SOURce:GPRF:GENerator:SEQuence:WAVeform:ENDCondition

Pattern End Condition of Sequence Waveform

機能

波形リストテーブルの波形パターンが終了して，次に登録している波形パターンに切り替わる条件を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:ENDCondition
<list_table_no>,<index>,<condition>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:ENDCondition?
<list_table_no>,<index>
```

レスポンス

<condition>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<index>	波形インデックス番号
範囲	1～520
<condition>	終了条件
NONE	無し (波形は切り替わりません)
REPEAT	繰り返し回数が終了したら次の波形へ遷移します。
TRIGGER	トリガにより次の波形へ遷移します。
初期値	REPEAT

使用例

シーケンステーブル番号 1，波形インデックス番号 10 の波形パターンの終了条件を なし(切り替わらない)に設定します。

:SOUR:GPRF:GEN:SEQ:WAV:ENDC 1,10,NONE

シーケンステーブル番号 1，波形インデックス番号 10 の波形パターンの終了条件を問い合わせます。

:SOUR:GPRF:GEN:SEQ:WAV:ENDC? 1,10

> NONE

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:DM:POLarity

Sequence Waveform Spectrum Reverse

機能

シーケンス動作時に出力波形の IQ を反転するかを設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:DM:POLarity
<list_table_no>,<mode>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:DM:POLarity?
<list_table_no>
```

レスポンス

<mode>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<mode>	モード
NORMAL	IQ を反転しません。
INVERT	IQ を反転します。
初期値	NORMAL

使用例

シーケンス動作時の出力波形を IQ 反転なし に設定します。
:SOUR:GPRF:GEN:SEQ:WAV:GEN:DM:POL 1,NORMAL

シーケンス動作時の出力波形を問い合わせます。
:SOUR:GPRF:GEN:SEQ:WAV:GEN:DM:POL? 1
> NORMAL

備考

波形リストテーブル内の共通設定です。シーケンス実行中は，すべての波形パターンに対して IQ 反転設定が適用されます。

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:PCOunt?

Number of Registered Waveform

機能

波形リストテーブルの波形インデックス１から連続して登録されている波形パターンの数を読み出します。

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:PCOunt? <list_table_no>
```

レスポンス

<count_res>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<count_res>	波形リストテーブルの波形インデックス１から連続して登録されている波形パターンの数

使用例

シーケンステーブル番号１の波形インデックス１から連続して登録されている波形パターン数を問い合わせます。

:SOUR:GPRF:GEN:SEQ:WAV:GEN:PCO? 1

> 3

:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:DELay

Trigger Delay of Sequence Waveform Pattern

機能

波形リストテーブルに登録された波形パターンのトリガ発生遅延時間を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:DELay
<list_table_no>,<delay>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:DELay?
<list_table_no>
```

レスポンス

<delay>

単位	ms

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<delay>	遅延時間
範囲	0.000 ms～1000.000 ms
分解能	0.001 ms
サフィックスコード	S, MS, US, NS 省略した場合は MS として扱われます。
初期値	0 ms

使用例

シーケンステーブル番号 1 の波形パターンの遅延時間を 100.000 ms に設定します。

:SOUR:GPRF:GEN:SEQ:WAV:GEN:TRIG:DEL 1,100.000

シーケンステーブル番号 1 の波形パターンの遅延時間を問い合わせます。

:SOUR:GPRF:GEN:SEQ:WAV:GEN:TRIG:DEL? 1

> 100.000

備考

波形リストテーブル内の共通設定です。

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce

Trigger Source for Sequence Waveform Pattern

機能

波形リストテーブルの波形パターン切り替えに使用するトリガ源を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce
<list_table_no>,<trigger>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce?
<list_table_no>
```

レスポンス

<trigger>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<trigger>	トリガ源
EXTERNAL	外部トリガ
VIDEO	レベルトリガ
SGREQUEST	SG シーケンサからのトリガ
MANUAL	ユーザコマンドによるトリガ
初期値	EXTERNAL

使用例

シーケンステーブル番号 1 の波形パターンのトリガ源を 外部トリガ に設定します。
:SOUR:GPRF:GEN:SEQ:WAV:GEN:TRIG:SOUR 1,EXTERNAL

シーケンステーブル番号 1 の波形パターンのトリガ源を問い合わせます。
:SOUR:GPRF:GEN:SEQ:WAV:GEN:TRIG:SOUR? 1
> EXTERNAL

備考

波形リストテーブル内の共通設定です。

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:WAVeform:GETRigger

Group End Trigger of Sequence Waveform Pattern

機能

波形リストテーブルの波形パターンが切り替わるときの，SG シーケンサに対するトリガ出力（Group End Trigger）を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GETRigger
<list_table_no>,<index>,<on_off>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:GETRigger?
<list_table_no>,<index>
```

レスポンス

<on_off_res>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<index>	波形インデックス番号
範囲	1～520
<on_off>	トリガ出力
0	トリガを発生しません。
1	トリガを発生します。
OFF	トリガを発生しません。
ON	トリガを発生します。
初期値	0
<on_off_res>	トリガ出力
0	トリガを発生しません。
1	トリガを発生します。

使用例

シーケンステーブル番号 1，波形インデックス番号 10 の波形パターン終了時のトリガ出力を 発生なし に設定します。

:SOUR:GPRF:GEN:SEQ:WAV:GETR 1,10,OFF

シーケンステーブル番号 1，波形インデックス番号 10 の波形パターン終了時のトリガ出力を問い合わせます。

:SOUR:GPRF:GEN:SEQ:WAV:GETR? 1,10

> 0

関連コマンド

本コマンドで設定したトリガを SG シーケンステーブルで使用するには，
:SOURce:GPRF:GENerator:SEQuence:RX:TRIGger:SOURce でトリガを WFGend に設定します。

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:WAVeform:IREPetition

Repetition Count of Sequence Waveform Pattern

機能

波形パターンの繰り返し回数を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:IREPetition
<list_table_no>,<index>,<count>
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:IREPetition?
<list_table_no>,<index>
```

レスポンス

<count>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<index>	波形インデックス番号
範囲	1～520
<count>	繰り返し回数
範囲	1～65535
初期値	1

使用例

シーケンステーブル番号 1，波形インデックス番号 10 の波形パターンの繰り返し回数を 5 に設定します。
:SOUR:GPRF:GEN:SEQ:WAV:IREP 1,10,5

シーケンステーブル番号 1，波形インデックス番号 10 の波形パターンの繰り返し回数を問い合わせます。
:SOUR:GPRF:GEN:SEQ:WAV:IREP? 1,10
> 5

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:WAVeform:MTEXecute

Manual Trigger for Sequence Waveform

機能

波形パターンを切り替えるトリガを発生します。
波形リストテーブルのトリガ源が，ユーザコマンドによるトリガ（MANUAL）に設定されている場合に使用できます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:MTEXecute
```

使用例

波形パターンの切り替えトリガを発生します。
:SOUR:GPRF:GEN:SEQ:WAV:MTEX

関連コマンド

:SOURce:GPRF:GENerator:SEQuence:WAVeform:GENeral:TRIGger:SOURce

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
- ノーマルモード
- シーケンスモード - 停止中

:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:DELete

Sequence Waveform Pattern Delete

機能

波形リストテーブルに設定（登録）している波形パターンを削除します。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:DELete
<list_table_no>,<index>
```

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<index>	波形インデックス番号
範囲	1～520

使用例

シーケンステーブル番号 1，波形インデックス番号 10 の波形パターンを削除します。
:SOUR:GPRF:GEN:SEQ:WAV:PATT:DEL 1,10

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。
- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。
- 波形ロード
- 波形メモリデフラグ

:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:SELect

Sequence Waveform Pattern Configuration

機能

波形リストテーブルの波形インデックス番号を指定して，波形ファイル名とグループ番号を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:SELect
<list_table_no>,<index>,<file_name>[,<group_number>]
```

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:PATTern:SELect?
<list_table_no>,<index>
```

レスポンス

<file_name>,<group_number>

パラメータ

<list_table_no>	シーケンステーブル番号
範囲	1～16
<index>	波形インデックス番号
範囲	1～520
<file_name>	波形ファイル名（文字列）
<group_number>	グループ番号
範囲	波形ファイルで定義されたグループ番号の範囲

詳細

ファイル名は拡張子（xml）を除いた文字列です。
グループ番号を省略した場合は 1 が設定されます。

使用例

シーケンステーブル番号 3，波形インデックス番号 10 に，ファイル名"MV887011A_WCDMA_0002"の波形パターンのグループ番号 1 を設定します。
:SOUR:GPRF:GEN:SEQ:WAV:PATT:SEL 3,10,"MV887011A_WCDMA_0002",1
シーケンステーブル番号 3，波形インデックス番号 10 の波形ファイル名，グループ番号を問い合わせます。
:SOUR:GPRF:GEN:SEQ:WAV:PATT:SEL? 3,10
> "MV887011A_WCDMA_0002",1

備考

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

指定したグループ番号が存在しない場合，パラメータエラーとなります。

:SOURce:GPRF:GENerator:SEQuence:WAVeform:SYNChronize:STARt:STATe?

Sequence Waveform Synchronization State

機能

波形パターンの同期出力開始待ち状態を問い合わせます。

クエリ

```
:SOURce:GPRF:GENerator:SEQuence:WAVeform:SYNChronize:STARt:STATe?
```

レスポンス

<status>

パラメータ

<status>	同期待ち状態
0	同期出力開始待ち状態
1	同期出力開始待ち解除状態

使用例

波形パターンの同期出力開始待ち状態を問い合わせます。

:SOUR:GPRF:GEN:SEQ:WAV:SYNC:STAR:STAT?

> 1

:SOURce:GPRF:GENerator:STATe

RF Output

機能

RF 信号出力の On / Off を設定，または問い合わせます。

コマンド

```
:SOURce:GPRF:GENerator:STATe <on_off>
```

クエリ

```
:SOURce:GPRF:GENerator:STATe?
```

レスポンス

<on_off_res>

パラメータ

<on_off>	RF 出力の On / Off
0	Off
1	On
OFF	Off
ON	On
初期値	0
<on_off_res>	RF 出力の On / Off
0	Off
1	On

詳細

MU887002A-007 搭載時，以下の制限事項があります。設定コマンドは有効ですが実行エラーとなります。

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では，入力または出力のどちらかにしか設定できません。

使用例

RF 出力の設定を On にします。

```
:SOUR:GPRF:GEN:STAT ON
```

RF 出力の状態を問い合わせます。

```
:SOUR:GPRF:GEN:STAT?
> 1
```

備考

本設定が OFF の場合，内部的に出力レベルを最小値に設定します。

動作モードが以下の状態の場合，設定コマンドは実行エラーとなります。

- シーケンスモード - 動作中

以下の処理が実行されている場合，設定コマンドは実行エラーとなります。

- 波形ロード
- 波形メモリデフラグ

第6章 Native コマンドリファレンス

この章では，Native コマンドの詳細について説明します。

6.1 コマンド一覧 6-2
6.1.1 IEEE488.2共通コマンド 6-2
6.1.2 基本操作コマンド 6-3
6.1.3 VSGコマンド 6-11
6.2 コマンドの詳細 6-12
6.2.1 IEEE488.2共通コマンド 6-13
6.2.2 アルファベット順コマンド 6-23

6.1 コマンド一覧

6.1.1 IEEE488.2共通コマンド

機能	コマンド	クエリ	レスポンス
Clear Status Command	*CLS	-----	-----
Standard Event Status Enable Command	*ESE register	*ESE?	register
Standard Event Status Register Query	-----	*ESR?	register
Identification Query	-----	*IDN? [index]	ANRITSU,MT8870A[:slot:MU887002A],serial[:serial2],firm[:firm2]
Operation Complete	*OPC	*OPC?	operation
Preset	*RST	-----	-----
Service Request Enable Command	*SRE register	*SRE?	register
Status Byte Register	-----	*STB?	register
Trigger	*TRG	-----	-----
Self Test	-----	*TST?	test_result
Wait to Continue	*WAI	-----	-----

6.1.2　基本操作コマンド

Calibration

機能	コマンド	クエリ	レスポンス
Execute BAND Calibration	BANDCAL	-----	-----
BAND Calibration Temperature	BANDCAL_TEMP temperature	-----	-----
BAND Calibration Last Time	-----	BANDCAL_TM?	year,month,day,hour,min
BAND Calibration Error Last Time	-----	BANDCALERR_TM?	year,month,day,hour,min
Band Calibration Error Count	-----	BANDCALERRCNT?	count
Result of Band Calibration	-----	BANDCALRESULT?	Result
Temperature at Full Calibration Execution	-----	CALTEMP?	temp
Elapsed Time after Full Calibration	-----	CALTIME?	time
Execute FULL Calibration	FULLCAL	-----	-----
FULL Calibration Last Time	-----	FULLCAL_TM?	year,month,day,hour,min
FULL Calibration Error Last Time	-----	FULLCALERR_TM?	year,month,day,hour,min
Full Calibration Error Count	-----	FULLCALERRCNT?	count
Result of Full Calibration	-----	FULLCALRESULT?	Result

Communication

機能	コマンド	クエリ	レスポンス
Delimiter	DELM code	DELM?	code
Domain Name	DOMAINNAME name	DOMAINNAME?	name
GPIB Address	GPIBADDR gp_address	GPIBADDR?	gp_address
GPIB Address	GPIBADDRALL address1,address2,address3,address4	GPIBADDRALL?	address1,address2,address3,address4
Host Name	HOSTNAME host	HOSTNAME?	host
Get Current IPv4 Address	-----	IPV4_CURRENT_ADDR?	num,ipaddr
Get Current IPv4 Default Gateway	-----	IPV4_CURRENT_DEFAULTGATEWAY?	gateway
Get Current IPv4 Subnet Mask	-----	IPV4_CURRENT_SUBNETMASK?	mask
IPv4 DNS Server Auto	IPV4_DNS_AUTO_SW switch	IPV4_DNS_AUTO_SW?	switch
IPv4 DNS primary address	IPV4_DNS_PRIMARY ip4addr	IPV4_DNS_PRIMARY?	ip4addr
IPv4 DNS secondary address	IPV4_DNS_SECONDARY ip4addr	IPV4_DNS_SECONDARY?	ip4addr
IPv4 Address	IPV4_STATIC_ADDR_ALLMODULE ipv4_addr1,ipv4_addr2,ipv4_addr3,ipv4_addr4	IPV4_STATIC_ADDR_ALLMODULE?	ipv4_addr1,ipv4_addr2,ipv4_addr3,ipv4_addr4
IPv4 Default Gateway	IPV4_STATIC_DEFAULTGATEWAY gateway	IPV4_STATIC_DEFAULTGATEWAY?	gateway
IPv4 Subnet Mask	IPV4_STATIC_SUBNETMASK mask	IPV4_STATIC_SUBNETMASK?	mask
IPv4 Type	IPV4_TYPE ipv4type	IPV4_TYPE?	ipv4type

Communication (Cont'd)

機能	コマンド	クエリ	レスポンス
Get Current IPv6 Address and Prefix	-----	IPV6_CURRENT_ADDR?	num,ipaddr
Get IPv6 Default Router Address	-----	IPV6_CURRENT_DEFAULTROUTER?	ip6addr
Get IPv6 Link Local Address	-----	IPV6_CURRENT_LINK_LOCAL_ADDR?	ip6addr
IPv6 DNS Server Auto	IPV6_DNS_AUTO_SW switch	IPV6_DNS_AUTO_SW?	switch
IPv6 DNS primary address	IPV6_DNS_PRIMARY ip6addr	IPV6_DNS_PRIMARY?	ip6addr
IPv6 DNS secondary address	IPV6_DNS_SECONDARY ip6addr	IPV6_DNS_SECONDARY?	ip6addr
IPv6 Address	IPV6_STATIC_ADDR_ALLMODULE ipv6_addr1,ipv6_addr2,ipv6_addr3,ipv6_addr4	IPV6_STATIC_ADDR_ALLMODULE?	ipv6_addr1,ipv6_addr2,ipv6_addr3,ipv6_addr4
IPv6 Default Router	IPV6_STATIC_DEFAULTROUTER ip6addr	IPV6_STATIC_DEFAULTROUTER?	ipv6_addr
IPv6 Type	IPV6_TYPE ipv6type	IPV6_TYPE?	ipv6type
IP Version	IPVER ipver	IPVER?	ipver
MAC Address	-----	MACADDR?	macaddr
Network Restart	NETRESTART	-----	-----
Raw Socket Port Number	PORTSETTING port	PORTSETTING?	port
Terminator of Message	TRM code	TRM?	code

Event Status Register

機能	コマンド	クエリ	レスポンス
End Event Status (Generator) Enable Command	ESE0 register	ESE0?	register
Error Event Status (Generator) Enable Command	ESE1 register	ESE1?	register
End Event Status (Measure) Enable Command	ESE2 register	ESE2?	register
Error Event Status (Measure) Enable Command	ESE3 register	ESE3?	register
End Event Status (Generator) Register Query	-----	ESR0?	register
Error Event Status (Generator) Register Query	-----	ESR1?	register
End Event Status (Measurement) Register Query	-----	ESR2?	register
Error Event Status (Measurement) Register Query	-----	ESR3?	register

FTP

機能	コマンド	クエリ	レスポンス
FTP User List	-----	FTPUSER_LIST?	name
Change FTP User Name	FTPUSER_CHANGE before,after	-----	-----
Change FTP User Password	FTPUSER_PASSWD name,before,after	-----	-----

Information

機能	コマンド	クエリ	レスポンス
License Information	-----	KEYINFO? model	kind,days
Log Clear	LOGCLEAR	-----	-----
Log Reading	-----	LOGREAD? [size]	log
Log Size Reading	-----	LOGSIZE?	log
Main Frame Information	-----	MFINFO? [index]	model name, model number,…
Main Frame Model Number.	-----	MFMODEL?	model
Main Frame Serial No.	-----	MFSERIAL?	serial
Main Frame Temperature	-----	MFTEMP?	temp
Main Frame Running Time Count Read	-----	MFTMCNT?	time
Module Information	-----	SYSINFO? [index]	-----
Module Model Number.	-----	SYSMODEL?	model
Module Serial No.	-----	SYSSERIAL?	serial
Module Slot Number	-----	SYSSLOT?	slot
Module Temperature	-----	SYSTEMP?	temp
Module Running Time Counter Read	-----	SYSTMCNT?	time
TRX 1/2	-----	SYSTRX?	trx
Package Version	-----	SYSVER?	ver
Chassis Power ON Count	-----	MFPOWERONCNT?	num
Count of Module Insertion	-----	SYSPLUGCNT?	num
Module Power ON Count	-----	SYSPOWERONCNT?	num
Firmware Version	-----	MCFV?	ver
Option Number	-----	MCOPT? model	n,num
Type Name of Software	-----	MCSOPT?	n,model
Type Name of Waveform	-----	MCWOPT?	n,model
Drive Size Reading	-----	DSIZE? [target]	total,blank
Lower Limit of Firmware Update	-----	UPDATELOWERLIMIT?	lowerlimit

Interface Setting

機能	コマンド	クエリ	レスポンス
Set In/Out of Trigger Connector	MFTRIGPORT trigger	MFTRIGPORT?	trigger
Set Direction of RF Connector	PORT input,output[,output,…,output]	PORT?	input,output[,output,…,output]
Select Source of Trigger Input	SYSTRIGINSRC source	SYSTRIGINSRC?	source
Select Source of Trigger Output	SYSTRIGOUTSRC source	SYSTRIGOUTSRC?	source

Loss Correction

機能	コマンド	クエリ	レスポンス
Delete Loss Value from Cable Loss List	DELLOSSTBL [a[,b]]	-----	-----
External Loss On/Off	EXTLOSSW sw	EXTLOSSW?	sw
Loss Table	LOSSTBL index	LOSSTBL?	index
Number of Loss Table	-----	LOSSTBLCNT? index	num
Register Loss Value into Cable Loss List	LOSSTBLVAL frequency,Port1_2/out,Port1_2/in,Port3_4/out[,Port3_4/in]	LOSSTBLVAL? n	frequency,Port1_2/out,Port1_2/in,Port3_4/out[,Port3_4/in]
Register Loss Value into All Cable Loss List	LOSSTBLVALALL frequency,Port1out,Port1in,Port2out,Port2in,Port3out,Port3in,Port4out,Port3in,Port4in[,Port5out,Port5in,Port6out,Port6in,Port7out,Port7in,Port8out,Port8in,Port9out,Port9in,Port10out,Port10in,Port11out,Port11in,Port12out,Port12in]	LOSSTBLVALALL? n	frequency,Port1out,Port1in,Port2out,Port2in,Port3out,Port3in,Port4out,Port3in,Port4in[,Port5out,Port5in,Port6out,Port6in,Port7out,Port7in,Port8out,Port8in,Port9out,Port9in,Port10out,Port10in,Port11out,Port11in,Port12out,Port12in]

Reference Clock

機能	コマンド	クエリ	レスポンス
Adjust Reference Clock	ADJREFCLK val	ADJREFCLK?	val
Adjust Reference Clock Default	-----	ADJREFCLKDEF?	val
Frequency Reference	FREQREF source	FREQREF?	source
Frequency Reference	REF source	REF?	source_res
Reference Signal Query	-----	FREQREFSOURCE?	source
Reference Signal Query	-----	EXTREF?	source

System

機能	コマンド	クエリ	レスポンス
Set Date	DATETIME year,month,day,hour,minute,second	DATETIME?	year,month,day,hour,minute,second
Beep	BEEP sw	BEEP?	sw
Delete File	FDEL filepath	-----	-----
Check Firmware Update Package	-----	FIRMCHECK? filename	result
Firmware Update	FIRMUPDATE target[,option]	FIRMUPDATE?	result
File List	-----	FLIST? path	num,filelist
Go to Local	GTL	-----	-----
License Key Check	-----	KEYCHECK? filename	result
License Key Install	KEYINST	-----	-----
Reboot	REBOOT	-----	-----
Shutdown	SHUTDOWN	-----	-----
System Error	-----	SYSERR?	code,msg
System All Errors	-----	SYSERRALL?	code,msg
System Error Count	-----	SYSERRCNT?	num
System Error Code	-----	SYSERRCODE?	code

System (Cont'd)

機能	コマンド	クエリ	レスポンス
System All Error Codes	-----	SYSERRCODEALL?	code
System Last Error	-----	SYSLASTERROR?	msg
Language Selection of Remote Command	SYST:LANG mode	SYST:LANG?	mode
User Area Delete	SYSTEMDEFAULT	-----	-----
System Test	-----	SYSTEST?	status

6.1.3 VSGコマンド

Native コマンドモードでは，SCPI コマンドの省略形を使用します。

例：

:SOURce:GPRF:GENerator:RFSettings:LEVel	SOUR:GPRF:GEN:RFS:LEV
:SOURce:GPRF:GENerator:SEQuence:RX:GENeral:STEP:LENGth	SOUR:GPRF:GEN:SEQ:RX:GEN:STEP:LENG

6.2 コマンドの詳細

この節では Native コマンドを，アルファベット順に説明しています。

■コマンド表の見方

EX ..... コマンド名(ヘッダー)

Example ..... コマンドの機能名

機能 ..... コマンドの機能

コマンド ..... プログラミングコマンドの構文

クエリ ..... クエリの構文

レスポンス ..... レスポンスの構文

パラメータ ..... パラメータの定義

詳細 ..... コマンドの制約事項など

使用例 ..... コマンドの使用例

関連コマンド ..... 関連するコマンドの紹介

■サフィックスコード一覧

サフィックスコード	単位
GZ	GHz
KZ	kHz
MZ	MHz
Z	Hz

6.2.1 IEEE488.2共通コマンド

*CLS

Clear Status Command

機能

標準イベントステータス・レジスタとステータスバイト・レジスタ（出力キュー，MAV ビットを除く）をクリアします。

コマンド

```
*CLS
```

詳細

プログラムメッセージターミネータの後，あるいはクエリメッセージユニットの前に*CLS コマンドを送信すると，すべてのイベント・ステータスバイト・レジスタ，およびイベント・キューがクリアされます。

使用例

標準イベントステータス・レジスタとステータスバイト・レジスタ(出力キュー，MAV ビットを除く)をクリアする場合

*CLS

関連コマンド

*RST　ステータスバイトレジスタ，イベントステタスレジスタはクリアされません。

*ESE　標準イベントステータスレジスタのみクリアされます。

*ESE

Standard Event Status Enable Command

機能

標準イベントステータスのイネーブル・レジスタを設定します。設定したイベントが発生すると，ステータスバイト・レジスタの bit5 にあるエンドサマリビット（ESB）の値が 1（真）になります。

コマンド

```
*ESE register
```

クエリ

```
*ESE?
```

レスポンス

register 標準イベントステータス・イネーブル・レジスタ
値=bit0 + bit1 + ... + bit7
bit0 = 2^0 = 1　　オペレーション終了
bit1 = 2^1 = 2　　リクエストコントロール
bit2 = 2^2 = 4　　クエリエラー
bit3 = 2^3 = 8　　デバイスエラー
bit4 = 2^4 = 16　　実行エラー
bit5 = 2^5 = 32　　コマンドエラー
bit6 = 2^6 = 64　　ユーザリクエスト
bit7 = 2^7 = 128　　電源オン

パラメータ

register	標準イベントステータス・イネーブル・レジスタ
範囲	0～255
分解能	1
初期値	0

詳細

標準イベントステータス・イネーブル・レジスタの bit0，1，2，3，4，5，6，7 に対応する値 2^0 = 1，2^1 = 2，2^2 = 4，2^3 = 8，2^4 = 16，2^5 = 32，2^6 = 64，2^7 = 128 の中から，イネーブルにしたいビットの値の総和がパラメータとなります。

使用例

標準イベントステータス・イネーブル・レジスタのビット 1 をイネーブルにします。

*ESE 2
*ESE?
> 2

*ESR?

Standard Event Status Register Query

機能

標準イベントステータス・レジスタを問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
*ESR?
```

レスポンス

register　標準イベントステータス・レジスタ
値=bit0 + bit1 + ... + bit7
bit0 = 2^0 = 1　　オペレーション終了
bit1 = 2^1 = 2　　リクエストコントロール
bit2 = 2^2 = 4　　クエリエラー
bit3 = 2^3 = 8　　デバイスエラー
bit4 = 2^4 = 16　　実行エラー
bit5 = 2^5 = 32　　コマンドエラー
bit6 = 2^6 = 64　　ユーザリクエスト
bit7 = 2^7 = 128　　電源オン

パラメータ

register	標準イベントステータス・レジスタ
範囲	0～255

詳細

標準イベントステータス・レジスタの bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8, 24 = 16, 2^5 = 32, 2^6 = 64, 2^7 = 128 の中から, 発生したイベントのビットの値の総和がパラメータとなります。

使用例

標準イベントステータス・レジスタの値を読み出します。

```
*ESR?
```

*IDN?

Identification Query

機能

デバイスの識別情報を問い合わせます。

クエリ

```
*IDN? [index]
```

レスポンス

パラメータが 1 の場合

ANRITSU,MT8870A,serial,firm

ANRITSU	会社名
MT8870A	シャーシの形名
serial	シャーシのシリアル番号（シャーシ固有の番号）
firm	シャーシの FPGA バージョン

パラメータが 2 の場合

ANRITSU,MT8870A:slot:MU887002A,serial1:serial2,firm1:firm2

ANRITSU	会社名
MT8870A	シャーシの形名
slot	モジュールが挿入されているスロット番号
MU887002A	モジュールの形名
serial1	シャーシのシリアル番号（シャーシ固有の番号）
serial2	モジュールのシリアル番号（モジュール固有の番号）
firm1	シャーシの FPGA バージョン
firm2	モジュールのパッケージバージョン

パラメータを省略した場合

ANRITSU,MU887002A,serial,firm

ANRITSU	会社名
MU887002A	モジュールの形名
serial	モジュールのシリアル番号（モジュール固有の番号）
firm	モジュールのパッケージバージョン

パラメータ

index	取得情報のインデックス
範囲	1～2

使用例

本器の識別情報を問い合わせます。

*IDN?

備考

MU887002A の場合，TRX1，TRX2 ともモジュールのシリアル番号は同じ値が返ります。

*OPC

Operation Complete

機能

*OPCコマンドを送信すると，選択されたペンディング中のデバイス動作がすべて終了したとき，標準イベントステータス・レジスタの中の bit0，すなわち，オペレーション終了がセットされます。
*OPC?クエリは選択されたペンディング中の動作がすべて終了したら，出力キューに 1 をセットし，MAV が発生するまで待ちます。

コマンド

```
*OPC
```

クエリ

```
*OPC?
```

レスポンス

operation　　操作完了フラグ

使用例

操作の完了を問い合わせます。

```
*OPC?
> 0
```

*RST

Preset

機能

初期化を実行します。

コマンド

*RST

詳細

端末との接続状態によっては初期化できないことがあります。

次の設定は*RST では初期化されません。
通信（IPv4, IPv6, GPIB）に関する設定，日付，時刻
シーケンスモードの場合は，シーケンス処理を停止してノーマルモードに変わります。
モジュール間の波形パターン同期待ちの場合は，同期待ち状態が解除されます。

使用例

初期化を実行します。
*RST

関連コマンド

REBOOT	*RST と同じ機能
SHUTDOWN	*RST と同じ機能

*SRE

Service Request Enable Command

機能

サービスリクエスト・イネーブル・レジスタを設定します。サービスリクエスト・イネーブル・レジスタで設定したイベントが発生すると，ステータスバイト・レジスタの MSS (bit6) が 1 (真) になります。

コマンド

```
*SRE register
```

クエリ

```
*SRE?
```

レスポンス

register　サービスリクエスト・イネーブル・レジスタ
値=bit0 + bit1 + ... + bit7

	SCPI コマンドモード	Native コマンドモード
bit0 = 2^0 = 1	未使用	END イベントステータス・レジスタ (信号発生器)
bit1 = 2^1 = 2	未使用	ERR イベントステータス・レジスタ (信号発生器)
bit2 = 2^2 = 4	エラー・イベントキュー	END イベントステータス・レジスタ (測定)
bit3 = 2^3 = 8	クエスチョナブルステータス・レジスタ	ERR イベントステータス・レジスタ (測定)
bit4 = 2^4 = 16	MAV	MAV
bit5 = 2^5 = 32	ESB	ESB
bit6 = 2^6 = 64	未使用	未使用
bit7 = 2^7 = 128	実行状態レジスタ	未使用

パラメータ

register	サービスリクエスト・イネーブル・レジスタ
範囲	0～255
分解能	1
初期値	0

詳細

サービスリクエスト・イネーブル・レジスタの bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 16, 2^5 = 32, 2^6 = 64, 2^7 = 128 の中から，イネーブルにしたいビット値の総和をパラメータにします。

使用例

終了 (END) イベントが発生したときに MSS のビットを 1 (真) にします。

```
*SRE 4
```

*STB?

Status Byte Register

機能

ステータスバイト・レジスタの値を問い合わせます。ステータスバイト・レジスタの bit6 は MSS (マスタサマリステータス) として読み出されます。

クエリ

```
*STB?
```

レスポンス

register　ステータスバイト・レジスタ
値=bit0 + bit1 + ... + bit7

	SCPI コマンドモード	Native コマンドモード
bit0 = 2^0 = 1	未使用	END イベントステータス・レジスタ (信号発生器)
bit1 = 2^1 = 2	未使用	ERR イベントステータス・レジスタ (信号発生器)
bit2 = 2^2 = 4	エラー・イベントキュー	END イベントステータス・レジスタ (測定)
bit3 = 2^3 = 8	クエスチョナブルステータス・レジスタ	ERR イベントステータス・レジスタ (測定)
bit4 = 2^4 = 16	MAV	MAV
bit5 = 2^5 = 32	ESB	ESB
bit6 = 2^6 = 64	MSS	MSS
bit7 = 2^7 = 128	実行状態レジスタ	未使用

パラメータ

register	ステータスバイト・レジスタ
範囲	0～255

詳細

サービスリクエスト・イネーブル・レジスタの bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 16, 2^5 = 32, 2^6 = 64, 2^7 = 128 の中から, イネーブルにしたいビット値の総和をパラメータにします。

使用例

ステータスバイト・レジスタの値を読み出す場合

```
*STB?
> 0
```

*TRG

Trigger

機能

トリガをかけます。

コマンド

```
*TRG
```

*TST?

Self Test

機能

自己診断テストの結果を問い合わせます。

クエリ

```
*TST?
```

レスポンス

test_result	自己診断テストの結果
0	正常終了
1	異常終了

使用例

自己診断テストの結果を問い合わせます。

```
*TST?
> 0
```

関連コマンド

SYSTEST, :SYSTem:SELF:TEST

*WAI

Wait to Continue

機能

コマンドが実行されているとき，次のコマンドを待機させます。

コマンド

```
*WAI
```

詳細

実行中のすべての処理が完了するまで次のコマンドの処理を開始しません。

使用例

*WAI コマンドを使用して測定の完了を待ちます。

SNGLS;*WAI

CDMA2K_TXPWR?

6.2.2 アルファベット順コマンド

ADJREFCLK

Adjust Reference Clock

機能

基準周波数補正値を設定，または問い合わせます。

コマンド

```
ADJREFCLK val
```

クエリ

```
ADJREFCLK?
```

レスポンス

val

パラメータ

val	調整値
範囲	0～1023
分解能	1
初期値	512

詳細

MT8870A に装着されている全モジュールで共通の設定です。

使用例

ADJREFCLK 512
ADJREFCLK?
> 512

備考

*RST コマンドで初期化されません。

ADJREFCLKDEF?

Adjust Reference Clock Default

機能

工場出荷時の基準周波数補正値を取得します。

クエリ

```
ADJREFCLKDEF?
```

レスポンス

val　　工場出荷時の調整値

パラメータ

val	工場出荷時の調整値
範囲	0～1023
分解能	1

詳細

MT8870A に装着されている全モジュールで共通の値です。

使用例

ADJREFCLKDEF?
> 512

BANDCAL

Execute BAND Calibration

機能

バンド校正（Band Calibration）を実行します。

コマンド

BANDCAL

使用例

Band Calibration を実行します。
BANDCAL

BANDCALERRCNT?

Band Calibration Error Count

機能

Band Calibration を失敗した回数を問い合わせます。

クエリ

BANDCALERRCNT?

レスポンス

count

パラメータ

count		Band Calibration を失敗した回数
	範囲	0～

使用例

Band Calibration を失敗した回数を問い合わせます。
BANDCALERRCNT?
> 0

BANDCALERR_TM?

BAND Calibration Error Last Time

機能

最後に Band Calibration を失敗した日時を問い合わせます。

クエリ

```
BANDCALERR_TM?
```

レスポンス

year,month,day,hour,min

パラメータ

year	年 (2001～)
month	月 (1～12)
day	日 (1～31)
hour	時 (0～23)
min	分 (0～59)

使用例

最後に Band Calibration を失敗した時刻を問い合わせます。

BANDCALERR_TM?

> 2011,09,13,21,45

備考

Band Calibration を 1 度も失敗していない場合, レスポンスは"***"になります。

BANDCALRESULT?

Result of Band Calibration

機能

Band Calibration の結果を問い合わせます。

クエリ

```
BANDCALRESULT?
```

レスポンス

Result

パラメータ

Result	Band Calibration の結果
PASS	Calibration 成功
FAIL	Calibration 失敗
UNEXECUTED	Calibration 未実行

使用例

Band Calibration の結果を問い合わせます。
BANDCALRESULT?
> PASS

BANDCAL_TEMP

BAND Calibration Temperature

機能

内部温度が，前回 Band Calibration を実行した時点より指定値以上変化している場合，Band Calibration を実行します。
内部温度変化が指定値未満の場合は，Band Calibration を実行しません。

コマンド

```
BANDCAL_TEMP temperature
```

パラメータ

temperature	温度
範囲	1.0～50.0°C
分解能	0.1°C

使用例

内部温度が，前回バンド校正を実行した時点より 2℃以上変化している場合，Band Calibration を実行します。
BANDCAL_TEMP 2.0

BANDCAL_TM?

BAND Calibration Last Time

機能

最後に Band Calibration を実行した日時を問い合わせます。

クエリ

```
BANDCAL_TM?
```

レスポンス

year,month,day,hour,min

パラメータ

year	年（2001～）
month	月（1～12）
day	日（1～31）
hour	時（0～23）
min	分（0～59）

使用例

最後に Band Calibration を実行したときの時刻を問い合わせます。
BANDCAL_TM?
> 2011,09,13,11,45

備考

Band Calibration を 1 度も実行していない場合，レスポンスは"***"になります。

BEEP

Beep

機能

ブザーのオン/オフを設定，または問い合わせます。

コマンド

```
BEEP sw
```

クエリ

BEEP?

レスポンス

sw

パラメータ

sw	ブザーのオン/オフ
ON	ブザーを鳴らします。
OFF	ブザーを鳴らしません。

詳細

モジュールごとに設定できます。
オンに設定すると，次の場合にブザーが鳴ります。
「ピ」は短い音，「ピー」は長い音を表します。

- 電源投入時の自己診断で異常がある場合：「ピー」
- 電源投入時のソフトウェア起動時：「ピ，ピ，ピ」
- 電源切断時，または REBOOT，SHUTDOWN コマンド送信時：「ピ，ピ」
- 送信したコマンドにエラーがある場合：「ピ」

使用例

```
BEEP ON
BEEP?
>ON
```

備考

*RST コマンドで初期化されません。

CALTEMP?

Temperature at Full Calibration Execution

機能

電源を入れてから最後に Full Calibration を実行したときのモジュールの温度を問い合わせます。

クエリ

CALTEMP?

レスポンス

temp

パラメータ

temp	温度（℃）

使用例

電源を入れてから最後に Full Calibration を実行したときのモジュールの温度を問い合わせます。

CALTEMP?

> 32.31

関連コマンド

電源を入れてから Full Calibration を 1 度も実行していない場合, レスポンスは"***"になります。

CALTIME?

Elapsed Time after Full Calibration

機能

最後に Full Calibration を実行してからの経過時間を問い合わせます。

クエリ

CALTIME?

レスポンス

time

パラメータ

time	経過時間　秒（seconds）単位の時間	

使用例

最後に Full Calibration を実行してからの経過時間を問い合わせます。

CALTIME?

> 1280

備考

Full Calibration を 1 度も実行していない場合，レスポンスは"****"になります。

DATETIME

Set Date

機能

日時を設定，または問い合わせます。

コマンド

```
DATETIME year,month,day,hour,minute,second
```

クエリ

```
DATETIME?
```

レスポンス

year,month,day,hour,minute,second

パラメータ

year	年
範囲	2012～2027
分解能	1
month	月
範囲	1～12
分解能	1
day	日
範囲	1～31
分解能	1
hour	時
範囲	0～23
分解能	1
minute	分
範囲	0～59
分解能	1
second	秒
範囲	0～59
分解能	1

使用例

DATETIME 2012,9,13,11,6,17
DATETIME?
> 2012,9,13,11,6,17

備考

日時の設定に失敗した場合，レスポンスは"***"になります。

DELLOSSTBL

Delete Loss Value from Cable Loss List

機能

損失補正テーブルのリストに設定された補正値を消去します。
削除範囲は，登録されている共通損失補正テーブルのリスト No.に相当し，削除したい範囲を指定します。
削除範囲を指定しなかった場合は，損失補正テーブルのリストの値を全削除します。

コマンド

```
DELLOSSTBL [a[,b]]
```

パラメータ

a		削除範囲指定
	範囲	1～1000
	分解能	1
b		削除範囲指定
	範囲	a～1000
	分解能	1

使用例

DELLOSSTBL 1,1000

DELM

Delimiter

機能

デリミタ（メッセージ末尾に付加される文字）を設定，または問い合わせます。

コマンド

```
DELM code
```

クエリ

```
DELM?
```

レスポンス

code

パラメータ

code	終端コード
LF	LF (Line Feed)
CRLF	CR/LF (Carriage Return and Line Feed)
NONE	None (EOI のみ)
初期値	CRLF

詳細

モジュールごとに設定できます。

MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，終端コードは初期値に変更されます。

使用例

DELM CRLF
DELM?
>CRLF

備考

TRM と同じ処理をします。

*RST，REBOOT，NETRESTART コマンドで初期化されません。

イーサネット接続でパラメータを NONE に設定した場合，本器からのレスポンスメッセージはデリミタが CRLF に設定されています。

DOMAINNAME

Domain Name

機能

ドメイン名を設定, または問い合わせます。

コマンド

```
DOMAINNAME name
```

クエリ

```
DOMAINNAME?
```

レスポンス

name　ドメイン名

パラメータ

name　ドメイン名

最大　63 バイト

初期値　"DOMAIN"

設定時に以下の命名規則に適合していない場合はパラメータ・エラーとなります。

- 最初の文字は英数字であること。
- 途中の文字は英数字またはハイフンであること。
- 最後の文字は英数字であること。
- 文字を""で囲っていること。

詳細

MT8870A に装着されている全モジュールにドメイン名を設定します。
本コマンドの設定を有効にするには, NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると, ドメイン名は初期値に変更されます。

使用例

```
DOMAINNAME "Domain"
NETRESTART
DOMAINNAME?
> Domain
```

DSIZE?

Drive Size Reading

機能

記憶媒体の空き容量を問い合わせます。

クエリ

```
DSIZE? [target]
```

レスポンス

total,blank

パラメータ

target	空き容量を確認するエリア
LOG	ログを記録するエリア
USER	波形ファイル，インストーラファイルなどに使用するエリア
初期値	USER
total	指定エリアの総容量（バイト単位）
blank	指定エリアの空き容量（バイト単位）

使用例

DSIZE? LOG
> 2097152,734002

ESE0

End Event Status (Generator) Enable Command

機能

終了イベントステータス (信号発生器) のイネーブル・レジスタを設定します。

コマンド

```
ESE0 register
```

クエリ

```
ESE0?
```

レスポンス

register　値=bit0 + bit1 + ... + bit7
ビットの割り当ては，ESR0?を参照してください。

パラメータ

register	終了イベントステータス (信号発生器)・イネーブル・レジスタ
範囲	0～255
分解能	1
初期値	0

詳細

終了イベントステータス 0・イネーブル・レジスタの bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 $2^0 = 1$, $2^1 = 2$, $2^2 = 4$, $2^3 = 8$, $2^4 = 16$, $2^5 = 32$, $2^6 = 64$, $2^7 = 128$ の中から，イネーブルにしたいビットの値の総和がパラメータとなります。

使用例

終了イベントステータス・イネーブル・レジスタのビット 0 をイネーブルにします。

```
ESE0 1
ESE0?
> 1
```

ESE1

Error Event Status (Generator) Enable Command

機能

エラーイベントステータス（信号発生器）のイネーブル・レジスタを設定します。

コマンド

```
ESE1 register
```

クエリ

```
ESE1?
```

レスポンス

register　値=bit0 + bit1 + ... + bit7
ビットの割り当ては，ESR1?を参照してください。

パラメータ

register	エラーイベントステータス（信号発生器)・イネーブル・レジスタ
範囲	0～255
分解能	1
初期値	0

詳細

エラーイベントステータス・イネーブル・レジスタの bit0，1，2，3，4，5，6，7 に対応する値 $2^0 = 1$，$2^1 = 2$，$2^2 = 4$，$2^3 = 8$，$2^4 = 16$，$2^5 = 32$，$2^6 = 64$，$2^7 = 128$ の中から，イネーブルにしたいビットの値の総和がパラメータとなります。

使用例

エラーイベントステータス・イネーブル・レジスタのビット 2 をイネーブルにします。

```
ESE1 4
ESE1?
> 4
```

ESE2

End Event Status (Measure) Enable Command

機能

終了イベントステータス (測定) のイネーブル・レジスタを設定します。

コマンド

```
ESE2 register
```

クエリ

```
ESE2?
```

レスポンス

register　値=bit0 + bit1 + ... + bit7

ビットの割り当ては, アプリケーションによって変わります。

アプリケーションソフトウェアの取扱説明書の ESR2? を参照してください。

パラメータ

register	終了イベントステータス (測定)・イネーブル・レジスタ
範囲	0～255
分解能	1
初期値	0

詳細

終了イベントステータス・イネーブル・レジスタの bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 $2^0 = 1$, $2^1 = 2$, $2^2 = 4$, $2^3 = 8$, $2^4 = 16$, $2^5 = 32$, $2^6 = 64$, $2^7 = 128$ の中から, イネーブルにしたいビットの値の総和がパラメータとなります。

使用例

終了イベントステータス・イネーブル・レジスタ 2 のビット 4 をイネーブルにします。

```
ESE2 16
ESE2?
> 16
```

ESE3

Error Event Status (Measure) Enable Command

機能

エラーイベントステータス（測定）のイネーブル・レジスタを設定します。

コマンド

```
ESE3 register
```

クエリ

```
ESE3?
```

レスポンス

register　値=bit0 + bit1 + ... + bit7

ビットの割り当ては，アプリケーションによって変わります。

アプリケーションソフトウェアの取扱説明書の ESR3? を参照してください。

パラメータ

register	エラーイベントステータス・イネーブル・レジスタ
範囲	0～255
分解能	1
初期値	0

詳細

エラーイベントステータス・イネーブル・レジスタの bit0，1，2，3，4，5，6，7 に対応する値 $2^0 = 1$，$2^1 = 2$，$2^2 = 4$，$2^3 = 8$，$2^4 = 16$，$2^5 = 32$，$2^6 = 64$，$2^7 = 128$ の中から，イネーブルにしたいビットの値の総和がパラメータとなります。

使用例

エラーイベントステータス・イネーブル・レジスタ 0 のビット 6 をイネーブルにします。

```
ESE3 64
ESE3?
> 64
```

ESR0?

End Event Status (Generator) Register Query

機能

終了イベントステータス・レジスタ（信号発生器）を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

ESR0?

レスポンス

register　値=bit0 + bit1 + ... + bit7

bit0 = 2^0 = 1	ファイル読み込み終了
bit1 = 2^1 = 2	デフラグ完了
bit2 = 2^2 = 4	波形パターンの同期出力準備中
bit3 = 2^3 = 8	未使用（アプリケーション用に予約）
bit4 = 2^4 = 16	未使用（アプリケーション用に予約）
bit5 = 2^5 = 32	未使用（アプリケーション用に予約）
bit6 = 2^6 = 64	未使用（アプリケーション用に予約）
bit7 = 2^7 = 128	未使用（アプリケーション用に予約）

パラメータ

register	終了イベントステータス・レジスタ（信号発生器）
範囲	0～255

詳細

終了イベントステータス・レジスタ(信号発生器)の bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 16, 2^5 = 32, 2^6 = 64, 2^7 = 128 の中から，発生したイベントのビットの値の総和がパラメータとなります。

使用例

終了イベントステータス・レジスタ（信号発生器）の値を読み出します。

ESR0?
> 1

ESR1?

Error Event Status (Generator) Register Query

機能

エラーイベントステータス・レジスタ（信号発生器）を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
ESR1?
```

レスポンス

register　値=bit0 + bit1 + ... + bit7

bit0 = 2^0 = 1	読み込んだファイルにエラーがあります。
bit1 = 2^1 = 2	未使用（アプリケーション用に予約）
bit2 = 2^2 = 4	未使用（アプリケーション用に予約）
bit3 = 2^3 = 8	未使用（アプリケーション用に予約）
bit4 = 2^4 = 16	未使用（アプリケーション用に予約）
bit5 = 2^5 = 32	未使用（アプリケーション用に予約）
bit6 = 2^6 = 64	未使用（アプリケーション用に予約）
bit7 = 2^7 = 128	未使用（アプリケーション用に予約）

パラメータ

register	エラーイベントステータス・レジスタ（信号発生器）
範囲	0～255

詳細

エラーイベントステータス・レジスタ(信号発生器)の bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 16, 2^5 = 32, 2^6 = 64, 2^7 = 128 の中から，発生したイベントのビットの値の総和がパラメータとなります。

使用例

エラーイベントステータス・レジスタ（信号発生器）の値を読み出します。

```
ESR1?
> 0
```

ESR2?

End Event Status (Measurement) Register Query

機能

終了イベントステータス・レジスタ（測定）を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
ESR2?
```

レスポンス

register　値=bit0 + bit1 + ... + bit7

bit0 = 2^0 = 1	測定終了
bit1 = 2^1 = 2	未使用（アプリケーション用に予約）
bit2 = 2^2 = 4	未使用（アプリケーション用に予約）
bit3 = 2^3 = 8	未使用（アプリケーション用に予約）
bit4 = 2^4 = 16	未使用（アプリケーション用に予約）
bit5 = 2^5 = 32	未使用（アプリケーション用に予約）
bit6 = 2^6 = 64	未使用（アプリケーション用に予約）
bit7 = 2^7 = 128	未使用（アプリケーション用に予約）

ビットの割り当ては，アプリケーションによって変わります。
アプリケーションソフトウェア取扱説明書の ESR2?の説明を参照してください。

パラメータ

register	終了イベントステータス・レジスタ（測定）
範囲	0～255

詳細

終了イベントステータス・レジスタ（測定）の bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 16, 2^5 = 32, 2^6 = 64, 2^7 = 128 の中から，発生したイベントのビットの値の総和がパラメータとなります。

使用例

終了イベントステータス・レジスタ（測定）の値を読み出します。

```
ESR2?
> 0
```

ESR3?

Error Event Status (Measurement) Register Query

機能

エラーイベントステータス・レジスタ（測定）を問い合わせます。読み出された値から発生したイベントを特定することができます。

クエリ

```
ESR3?
```

レスポンス

register　値=bit0 + bit1 + ... + bit7

bit0 = 2^0 = 1　　レベルオーバー
bit1 = 2^1 = 2　　レベルアンダー
bit2 = 2^2 = 4　　タイムアウト
bit3 = 2^3 = 8　　未使用（アプリケーション用に予約）
bit4 = 2^4 = 16　　未使用（アプリケーション用に予約）
bit5 = 2^5 = 32　　未使用（アプリケーション用に予約）
bit6 = 2^6 = 64　　未使用（アプリケーション用に予約）
bit7 = 2^7 = 128　　未使用（アプリケーション用に予約）

ビットの割り当ては，アプリケーションによって変わります。
アプリケーションソフトウェア取扱説明書の ESR3?の説明を参照してください。

パラメータ

register	エラーイベントステータス・レジスタ（測定）
範囲	0～255

詳細

エラーイベントステータス・レジスタ（測定）の bit0, 1, 2, 3, 4, 5, 6, 7 に対応する値 2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8, 2^4 = 16, 2^5 = 32, 2^6 = 64, 2^7 = 128 の中から，発生したイベントのビットの値の総和がパラメータとなります。

使用例

エラーイベントステータス・レジスタ(測定)の値を読み出します。

```
ESR3?
> 4
```

EXTLOSSW

External Loss On/Off

機能

損失補正テーブルのリストの全ポートの DL/UL の損失データを使用して，レベル補正するかを設定，または問い合わせます。

コマンド

```
EXTLOSSW sw
```

クエリ

```
EXTLOSSW?
```

レスポンス

sw

パラメータ

sw	全ポートの DL/UL の損失値の有効/無効
COMMON	補正値を使用します。
ON	補正値を使用します。
OFF	補正値を使用しません。
初期値	OFF

詳細

複数の Port から信号出力する場合，Port 間の損失補正テーブル値の差が 12 dB を超えると信号は設定した出力レベルよりも低く出力されます。この場合，ランプ 2 が橙色に点灯します。
ただし，8 dB を超えた場合でも，出力レベルの誤差が大きくなる可能性があります。

使用例

EXTLOSSW ON
EXTLOSSW?
> ON

備考

損失補正テーブルのリストに設定されていない周波数の補正値には，前後の周波数の補正値を用いて線形補間で算出した値を使用します。
最小周波数未満の周波数の場合には最小周波数の補正値を，最大周波数より大きい周波数の場合には最大周波数の補正値を使用します。

EXTREF?

Reference Signal Query

機能

周波数基準信号源の状態を問い合わせます。

クエリ

EXTREF?

レスポンス

source

パラメータ

source	基準信号の状態
INT	内部基準信号源でロックしています。
VALI	外部基準信号源でロックしています。
INVALID	外部基準信号源でアンロック状態です。

詳細

すべてのスロットで共通です。

使用例

EXTREF?
> INT

関連コマンド

FREQREFSOURCE

FDEL

Delete File

機能

指定したファイルを削除します。

コマンド

```
FDEL filepath
```

パラメータ

filepath	ファイル名（パス情報を含みます） 最大 1024 文字 以下の文字は使用できません。 ¥ : * ? " < > |

詳細

ファイル名はパスを含めた形式で指定します。

使用例

FLIST? "CELLULAR/parameter"
> 3,setup.000,setup.001,setup.002
FDEL "CELLULAR/parameter/setup.002"
FLIST? "CELLULAR/parameter"
> 2,setup.000,setup.001

FIRMCHECK?

Check Firmware Update Package

機能

ファームウェアパッケージのチェックを行い，正常であれば更新準備を行います。

クエリ

```
FIRMCHECK? filename
```

レスポンス

result

パラメータ

filename	ファームウェアパッケージファイル名
result	ファームウェアパッケージファイルのチェック結果
0	正常
1	指定したファイルが見つからない。
2	ファイルが違う。
3	ファイルが壊れているまたは本器用のインストーラではない。
4	ファイルは正常だがインストールできないバージョンである。
5	ファイルは正常だが，TRX 2 では更新準備を行うことができない。

詳細

事前に FTP によりファームウェアパッケージファイルを指定フォルダに転送する必要があります。転送先フォルダは，"/install/package/" です。
FIRMUPDATE コマンドによりファームウェアをアップデートする前に，再起動したり電源を切ったりした場合は，更新準備は無効になります。
MU887002Aは，V3.00.00以降のファームウェアパッケージでなければインストールできません。
MU887002A-007搭載時は，V4.00.00以降のファームウェアパッケージでなければインストールできません。

使用例

FIRMCHECK? "package.inst"
>0

FIRMUPDATE

Firmware Update

機能

ファームウェアパッケージのアップデートを実行，またはアップデート結果を問い合わせます。

コマンド

```
FIRMUPDATE target[,option]
```

クエリ

```
FIRMUPDATE?
```

レスポンス

result

パラメータ

target	アップデート先モジュール
SINGLE	単体モジュール
ALL	MT8870A に装着されている全モジュール
option	オプション
DIFF	バージョンが違う場合だけアップデートします。
FORCE	強制的にアップデートします。
result	最後にインストールしたファームウェアの更新結果
0	正常
1	更新失敗
2	異常発生

詳細

事前に FIRMCHECK?コマンドで更新準備をする必要があります。
本コマンドを実行すると，システムが再起動します。再起動中は全モジュールがリモート制御できなくなります。
MT8870A 内の全モジュールをアップデートするか，コマンド送信先の単体モジュールをアップデートするかを選択してください。
FIRMUPDATE?で最後にインストールしたファームウェアパッケージのアップデート結果を取得できます。

使用例

FIRMUPDATE ALL,DIFF

FLIST?

File List

機能

指定したパスに保存されているファイルの一覧を問い合わせます。

クエリ

```
FLIST? path
```

レスポンス

num,filelist

パラメータ

path	パス
num	ファイル数
filelist	ファイル一覧

詳細

レスポンスが 10485760 文字以上のリストになる場合は，レスポンスの最後に"..."が付きます。
パスには"*.h"のようにファイル拡張子を指定できます。

使用例

FLIST? "CELLULAR/parameter"
> 3,setup.000,setup.001,setup.002
FLIST? "CELLULAR/parameter/*001"
> 1,setup.001

FREQREF

Frequency Reference

機能

周波数基準信号源を選択します。

コマンド

```
FREQREF source
```

クエリ

```
FREQREF?
```

レスポンス

source

パラメータ

source	内部基準信号/外部基準信号
INT	MT8870A の内部基準信号
AUTO	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
初期値	AUTO

詳細

MT8870A に装着されている全モジュールに基準信号源を設定します。モジュールごとに基準信号源を設定できません。

使用例

FREQREF INT
FREQREF?
> INT

関連コマンド

REF

備考

*RST コマンドで初期化されません。

FREQREFSOURCE?

Reference Signal Query

機能

周波数基準信号源の状態を問い合わせます。

クエリ

FREQREFSOURCE?

レスポンス

source

パラメータ

source	基準信号の状態
INT	内部基準信号源でロックしています。
EXT	外部基準信号源でロックしています。
EXTU	外部基準信号源でアンロック状態です。

詳細

すべてのスロットで共通です。

使用例

FREQREFSOURCE?
> EXT

関連コマンド

EXTREF

FTPUSER_CHANGE

Change FTP User Name

機能

FTP ユーザ名を変更します。

コマンド

```
FTPUSER_CHANGE before,after
```

パラメータ

before	変更前のユーザ名
after	変更後のユーザ名

詳細

ユーザ名に使用できる文字と文字数は以下のとおりです。
文字数は 1～31 文字までです。""で囲って設定します。
先頭文字は英字[大・小]とドット（.）とアンダースコア（_）です。
2 文字目以降は英数字[大・小]とドット（.）とアンダースコア（_）です。
すべて数字は不可とします。

使用例

user1 を user2 に変更します。
FTPUSER_CHANGE "user1","user2"

FTPUSER_LIST?

FTP User List

機能

すべての FTP ユーザ名を問い合わせます。

クエリ

```
FTPUSER_LIST?
```

レスポンス

name,name>,.....　ユーザ名

使用例

FTPUSER_LIST?
> user1,user2,user3

FTPUSER_PASSWD

Change FTP User Password

機能

FTP ユーザのパスワードを変更します。

コマンド

```
FTPUSER_PASSWD name,before,after
```

パラメータ

name	パスワードを変更するユーザ名
before	変更前のパスワード
after	変更後のパスワード

詳細

パスワードに使用できる文字と文字数は以下のとおりです。
文字数は 1～16 文字までです。""で囲って設定します。
使用可能文字は英数字[大・小]と記号（! #$%'()-=^~@[{;+]},.?_）です。

使用例

FTPUSER_PASSWD "user1","abcde","fghij"

FULLCAL

Execute FULL Calibration

機能

全帯域で校正を実行します。

コマンド

```
FULLCAL
```

詳細

MU887002A-007 搭載時，校正の実行時間は MU887002A-007 非搭載時よりも長くなります。

使用例

全帯域で校正を実行します。

FULLCAL

FULLCALERRCNT?

Full Calibration Error Count

機能

Full Calibration を失敗した回数を問い合わせます。

クエリ

```
FULLCALERRCNT?
```

レスポンス

count

パラメータ

count		Full Calibration を失敗した回数
	範囲	0 以上

使用例

Full Calibration を失敗した回数を問い合わせます。

FULLCALERRCNT?

> 0

FULLCALERR_TM?

FULL Calibration Error Last Time

機能

最後に Full Calibration を失敗した日時を問い合わせます。

クエリ

```
FULLCALERR_TM?
```

レスポンス

year,month,day,hour,min

パラメータ

year	年 (2001～)
month	月 (1～12)
day	日 (1～31)
hour	時 (0～23)
min	分 (0～59)

使用例

最後に Full Calibration を失敗したときの時刻を問い合わせます。

FULLCALERR_TM?

> 2012,03,10,11,45

備考

Full Calibration を 1 度も失敗していない場合, レスポンスは"***"になります。

FULLCALRESULT?

Result of Full Calibration

機能

Full Calibration の結果を問い合わせます。

クエリ

```
FULLCALRESULT?
```

レスポンス

result

パラメータ

result	Full Calibration の結果
PASS	Calibration 成功
FAIL	Calibration 失敗
UNEXECUTED	Calibration 未実行

使用例

Full Calibration の結果を問い合わせます。
FULLCALRESULT?
> PASS

FULLCAL_TM?

FULL Calibration Last Time

機能

最後に Full Calibration を実行した日時を問い合わせます。

クエリ

```
FULLCAL_TM?
```

レスポンス

year,month,day,hour,min

パラメータ

year	年 (2001～)
month	月 (1～12)
day	日 (1～31)
hour	時 (0～23)
min	分 (0～59)

使用例

最後に Full Calibration を実行したときの時刻を問い合わせます。

FULLCAL_TM?

> 2012,03,13,11,45

備考

Full Calibration を 1 度も実行していない場合, レスポンスは"***"になります。

GPIBADDR

GPIB Address

機能

GPIB アドレスを設定，または問い合わせます。

コマンド

```
GPIBADDR gp_address
```

クエリ

```
GPIBADDR?
```

レスポンス

gp_address　　GPIB アドレス

パラメータ

gp_address	GPIB アドレス	
範囲	1～30	
初期値	スロット 1	1
	スロット 2	1
	スロット 3	1
	スロット 4	1

詳細

MU887002A では設定はできますが，GPIB は使用できません。
モジュールの TRX ごとに GPIB アドレス設定できます。
GPIB アドレスは，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの GPIB アドレスが適用されます。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，GPIB アドレスは初期値に変更されます。

使用例

```
GPIBADDR 8
NETRESTART

GPIBADDR?
> 8
```

GPIBADDRALL

GPIB Address

機能

全モジュールの TRX の GPIB アドレスを設定，または問い合わせます。

コマンド

```
GPIBADDRALL address1,address2,address3,address4
```

クエリ

```
GPIBADDRALL?
```

レスポンス

address1,address2,address3,address4

パラメータ

address1	スロット 1 GPIB アドレス
範囲	1～30
初期値	1
address2	スロット 2 GPIB アドレス
範囲	1～30
初期値	1
address3	スロット 3 GPIB アドレス
範囲	1～30
初期値	1
address4	スロット 4 GPIB アドレス
範囲	1～30
初期値	1

詳細

MU887002A では設定はできますが，GPIB は使用できません。
全モジュールの TRX の GPIB アドレスを設定できます。
GPIB アドレスは，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの GPIB アドレスが適用されます。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，GPIB アドレスは初期値に変更されます。

使用例

GPIBADDRALL 1,2,3,4
NETRESTART

GPIBADDRALL?
> 1,2,3,4

GTL

Go to Local

機能

パネル操作ができるようにします。

コマンド

GTL

詳細

パネルの Remote ランプが消灯します。

使用例

GTL

HOSTNAME

Host Name

機能

ホスト名を設定，または問い合わせます。

コマンド

```
HOSTNAME host
```

クエリ

```
HOSTNAME?
```

レスポンス

host　　ホスト名

パラメータ

host　　ホスト名

最小　　1 バイト
最大　　63 バイト
初期値　　"HOST"

設定時に以下の命名規則に適合していない場合はパラメータ・エラーとなります。

- 最初の文字は英数字であること。
- 途中の文字は英数字またはハイフンであること。
- 最後の文字は英数字であること。
- 文字を""で囲っていること。

詳細

設定したホスト名は，スロット番号が後ろに追加され，HOSTNAME"-"スロット番号となります。
MT8870A に装着されている全モジュールにホスト名を設定します。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，ホスト名は初期値に変更されます。

使用例

HOSTNAME "ANRITSU-PC"
NETRESTART
HOSTNAME?
> ANRITSU-PC-1

備考

制御用コンピュータの Windows のネットワークに，本コマンドで設定したホスト名が表示されます。

IPV4_CURRENT_ADDR?

Get Current IPv4 Address

機能

モジュールの IPv4 アドレスを問い合わせます。

クエリ

```
IPV4_CURRENT_ADDR?
```

レスポンス

num	設定された IP アドレスの個数
ipaddr	IP アドレス

パラメータ

num	設定された IP アドレスの個数
ipaddr	IP アドレス

詳細

モジュールの IPv4 アドレスを取得します。

モジュールの IPv4 アドレスがオフに設定されている場合, レスポンスは 0 です。

使用例

IPV4_CURRENT_ADDR?
> 2,192.168.0.1,172.16.0.31

IPV4_CURRENT_DEFAULTGATEWAY?

Get Current IPv4 Default Gateway

機能

現在のデフォルト・ゲートウェイ・アドレスを問い合わせます。

クエリ

```
IPV4_CURRENT_DEFAULTGATEWAY?
```

レスポンス

gateway

パラメータ

gateway	デフォルト・ゲートウェイ・アドレス

詳細

通信先モジュールのデフォルト・ゲートウェイ・アドレスを取得します。
モジュールの IPv4 アドレスがオフに設定されている場合, レスポンスは 0 です。

使用例

IPV4_CURRENT_DEFAULTGATEWAY?
> 192.168.0.1

IPV4_CURRENT_SUBNETMASK?

Get Current IPv4 Subnet Mask

機能

現在のサブネットマスクを問い合わせます。

クエリ

```
IPV4_CURRENT_SUBNETMASK?
```

レスポンス

mask

パラメータ

mask	サブネットマスク

詳細

通信先モジュールのサブネットマスクを取得します。

モジュールの IPv4 アドレスがオフに設定されている場合, レスポンスは 0 です。

使用例

```
IPV4_CURRENT_SUBNETMASK?
> 255.255.255.0
```

IPV4_DNS_AUTO_SW

IPv4 DNS Server Auto

機能

DNS サーバのアドレスを自動的に取得するかどうかを設定，または問い合わせます。

コマンド

```
IPV4_DNS_AUTO_SW switch
```

クエリ

```
IPV4_DNS_AUTO_SW?
```

レスポンス

switch　DNS サーバアドレスの自動取得機能のオン/オフ設定

パラメータ

switch	DNS サーバのアドレス自動取得オンオフ設定
ON	オンにします。
OFF	オフにします。
初期値	OFF

詳細

MT8870A に装着されている全モジュールに設定します。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS サーバアドレスの自動取得は OFF に変更されます。

使用例

IPV4_DNS_AUTO_SW ON
NETRESTART

IPV4_DNS_AUTO_SW?
>ON

IPV4_DNS_PRIMARY

IPv4 DNS Primary Address

機能

DNS のプライマリ・アドレス（IPv4）を設定，または問い合わせます。

コマンド

```
IPV4_DNS_PRIMARY ip4addr
```

クエリ

```
IPV4_DNS_PRIMARY?
```

レスポンス

ip4addr	DNS プライマリ・アドレス

パラメータ

ip4addr	DNS プライマリ・アドレス xxx.xxx.xxx.xxx

詳細

MT8870A に装着されている全モジュールに DNS プライマリ・アドレスを設定します。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS プライマリ・アドレスは 0 に変更されます。

使用例

IPV4_DNS_PRIMARY "192.168.10.1"
NETRESTART
IPV4_DNS_PRIMARY?
>192.168.10.1

IPV4_DNS_SECONDARY

IPv4 DNS Secondary Address

機能

DNS のセカンダリ・アドレス（IPv4）を設定，または問い合わせます。

コマンド

```
IPV4_DNS_SECONDARY ip4addr
```

クエリ

```
IPV4_DNS_SECONDARY?
```

レスポンス

ip4addr

パラメータ

ip4addr	DNS セカンダリ・アドレス xxx.xxx.xxx.xxx

詳細

MT8870A に装着されている全モジュールに DNS セカンダリ・アドレスを設定します。
本コマンドの設定を有効にするには，:NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS セカンダリ・アドレスは 0 に変更されます。

使用例

IPV4_DNS_SECONDARY "192.168.10.2"
NETRESTART
IPV4_DNS_SECONDARY?
>192.168.10.2

IPV4_STATIC_ADDR_ALLMODULE

IPv4 Address

機能

モジュールの固定 IPv4 アドレスを設定，または問い合わせます。

コマンド

```
IPV4_STATIC_ADDR_ALLMODULE ipv4_addr1,ipv4_addr2,ipv4_addr3,ipv4_addr4
```

クエリ

```
IPV4_STATIC_ADDR_ALLMODULE?
```

レスポンス

ipv4_addr1,ipv4_addr2,ipv4_addr3,ipv4_addr4

パラメータ

ipv4_addr1	スロット 1 の IPv4 アドレス xxx.xxx.xxx.xxx
ipv4_addr2	スロット 2 の IPv4 アドレス xxx.xxx.xxx.xxx
ipv4_addr3	スロット 3 の IPv4 アドレス xxx.xxx.xxx.xxx
ipv4_addr4	スロット 4 の IPv4 アドレス xxx.xxx.xxx.xxx

詳細

MT8870A の各スロットの IPv4 アドレスを設定します。モジュールが挿入されていないスロットにも IPv4 アドレスを設定できます。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
IPv4 アドレスは，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの IPv4 アドレスが適用されます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，このコマンドの設定はクリアされます。

使用例

IPV4_STATIC_ADDR_ALLMODULE
"192.168.1.101","192.168.1.102","192.168.1.103","192.168.1.104"
NETRESTART

IPV4_STATIC_ADDR_ALLMODULE?
> 192.168.1.101,192.168.1.102,192.168.1.103,192.168.1.104

IPV4_STATIC_DEFAULTGATEWAY

IPv4 Default Gateway

機能

固定 IPv4 設定のデフォルト・ゲートウェイを設定，または問い合わせます。

コマンド

```
IPV4_STATIC_DEFAULTGATEWAY gateway
```

クエリ

```
IPV4_STATIC_DEFAULTGATEWAY?
```

レスポンス

gateway

パラメータ

gateway	デフォルト・ゲートウェイ・アドレス xxx.xxx.xxx.xxx

詳細

MT8870A に装着されている全モジュールにデフォルトゲートウェイを設定します。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
パラメータに"" (空文字) を指定すると，デフォルトゲートウェイの設定を初期化できます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，デフォルトゲートウェイは 0 に変更されます。

使用例

IPV4_STATIC_DEFAULTGATEWAY "192.168.0.1"
NETRESTART

IPV4_STATIC_DEFAULTGATEWAY?
> 192.168.0.1

IPV4_STATIC_SUBNETMASK

IPv4 Subnet Mask

機能

固定 IPv4 アドレスのサブネットマスクを設定，または問い合わせます。

コマンド

```
IPV4_STATIC_SUBNETMASK mask
```

クエリ

```
IPV4_STATIC_SUBNETMASK?
```

レスポンス

mask	サブネットマスク

パラメータ

mask	サブネットマスク xxx.xxx.xxx.xxx

詳細

MT8870A に装着されている全モジュールにサブネットマスクを設定します。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，サブネットマスクは 255/255/255/0 に変更されます。

使用例

IPV4_STATIC_SUBNETMASK "255.255.255.0"
NETRESTART

IPV4_STATIC_SUBNETMASK?
> 255.255.255.0

IPV4_TYPE

IPv4 Type

機能

IPv4 アドレスの種別を設定，または問い合わせます。

コマンド

```
IPV4_TYPE ipv4type
```

クエリ

```
IPV4_TYPE?
```

レスポンス

ipv4type

パラメータ

ipv4type	IPv4 種別
STATIC	固定 IP です。
DHCP	DHCP サーバから取得して設定します。
初期値	STATIC

詳細

MT8870A の全スロットに IPv4 アドレスの種別を設定します。
STATIC を設定した場合は，次のコマンドでアドレス，サブネットマスク，デフォルトゲートウェイを設定します。

IPV4_STATIC_ADDR_ALLMODULE
IPV4_STATIC_SUBNETMASK
IPV4_STATIC_DEFAULTGATEWAY

本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
IPv4 アドレスの種別は，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの IPv4 アドレスの種別が適用されます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，IPv4 アドレスの種別は初期値に変更されます。

使用例

IPV4_TYPE DHCP
NETRESTART

再起動後
IPV4_TYPE?
> DHCP

IPV6_CURRENT_ADDR?

Get Current IPv6 Address and Prefix

機能

現在の IPv6 アドレス/プリフィックス長を問い合わせます。

クエリ

`IPV6_CURRENT_ADDR?`

レスポンス

num,ipaddr,…

パラメータ

num	設定された IP アドレスの個数
ipaddr	IP アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx

詳細

通信先モジュールの IPv6 アドレス/プリフィックス長を問い合わせます。
モジュールの IPv6 アドレスがオフに設定されている場合, レスポンスは 0 です。

使用例

IPV6_CURRENT_ADDR?
> 2,2001:DB8::1/64,2001:DB8::2/64

IPV6_CURRENT_DEFAULTROUTER?

Get IPv6 Default Router Address

機能

現在のデフォルト・ルータ・アドレスを問い合わせます。

クエリ

```
IPV6_CURRENT_DEFAULTROUTER?
```

レスポンス

ip6addr

パラメータ

ip6addr	デフォルト・ルータ・アドレス xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

詳細

通信先モジュールの IPv6 アドレス/プリフィックス長を問い合わせます。
モジュールの IPv6 アドレスがオフに設定されている場合, レスポンスは 0 です。

使用例

IPV6_CURRENT_DEFAULTROUTER?
> 2001:DB8::1

IPV6_CURRENT_LINK_LOCAL_ADDR?

Get IPv6 Link Local Address

機能

現在のリンク・ローカル・アドレス/プリフィックス長を問い合わせます。

クエリ

```
IPV6_CURRENT_LINK_LOCAL_ADDR?
```

レスポンス

ip6addr

パラメータ

ip6addr	リンク・ローカル・アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx

詳細

通信先モジュールのリンク・ローカル・アドレス/プリフィックス長を問い合わせます。
モジュールの IPv6 アドレスがオフに設定されている場合，レスポンスは 0 です。

使用例

IPV6_CURRENT_LINK_LOCAL_ADDR?
> 2001:DB8::1/64

IPV6_DNS_AUTO_SW

IPv6 DNS Server Auto

機能

DNS サーバのアドレスを自動的に取得するかどうかを設定，または問い合わせます。

コマンド

```
IPV6_DNS_AUTO_SW switch
```

クエリ

```
IPV6_DNS_AUTO_SW?
```

レスポンス

switch　DNS サーバアドレスの自動取得機能のオン/オフ設定

パラメータ

switch	DNS サーバのアドレス自動取得オンオフ設定
ON	オンにします。
OFF	オフにします。
初期値	OFF

詳細

MT8870A に装着されている全モジュールに設定します。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS サーバアドレスの自動取得は OFF に変更されます。

使用例

IPV6_DNS_AUTO_SW ON
NETRESTART

IPV6_DNS_AUTO_SW?
>ON

IPV6_DNS_PRIMARY

IPv6 DNS Primary Address

機能

DNS のプライマリ・アドレス（IPv6）を設定，または問い合わせます。

コマンド

```
IPV6_DNS_PRIMARY ip6addr
```

クエリ

```
IPV6_DNS_PRIMARY?
```

レスポンス

ip6addr	DNS プライマリ・アドレス

パラメータ

ip6addr	DNS プライマリ・アドレス xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

詳細

MT8870A に装着されている全モジュールに DNS プライマリ・アドレスを設定します。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，DNS プライマリ・アドレスは 0 に変更されます。

使用例

IPV6_DNS_PRIMARY "fe80::215:c5ff:febb:42cf"
NETRESTART

IPV6_DNS_PRIMARY?
> fe80::215:c5ff:febb:42cf

IPV6_DNS_SECONDARY

IPv6 DNS Secondary Address

機能

DNS セカンダリ・アドレス (IPv6) を設定, または問い合わせます。

コマンド

```
IPV6_DNS_SECONDARY ip6addr
```

クエリ

```
IPV6_DNS_SECONDARY?
```

レスポンス

ip6addr	DNS セカンダリ・アドレス

パラメータ

ip6addr	DNS セカンダリ・アドレス xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

詳細

MT8870A に装着されている全モジュールに DNS セカンダリ・アドレスを設定します。
本コマンドの設定を有効にするには, NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると, DNS セカンダリ・アドレスは 0 に変更されます。

使用例

IPV6_DNS_SECONDARY "fe80::215:c5ff:febb:42ce"
NETRESTART

IPV6_DNS_SECONDARY?
> fe80::215:c5ff:febb:42ce

IPV6_STATIC_ADDR_ALLMODULE

IPv6 Address

機能

固定 IPv6 設定の IPv6 アドレス/プリフィックス長を設定，または問い合わせます。

コマンド

```
IPV6_STATIC_ADDR_ALLMODULE ipv6_addr1,ipv6_addr2,ipv6_addr3,ipv6_addr4
```

クエリ

```
IPV6_STATIC_ADDR_ALLMODULE?
```

レスポンス

ipv6_addr1,ipv6_addr2,ipv6_addr3,ipv6_addr4

パラメータ

ipv6_addr1	スロット 1 の IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx
ipv6_addr2	スロット 2 の IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx
ipv6_addr3	スロット 3 の IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx
ipv6_addr4	スロット 4 の IPv6 アドレス/プリフィックス長 xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx/xxx

詳細

MT8870A のスロットの IPv6 アドレスを設定します。モジュールが挿入されていないスロットにも IPv6 アドレスを設定できます。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
IPv6 アドレスは，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの IPv6 アドレスが適用されます。

使用例

```
IPV6_STATIC_ADDR_ALLMODULE
"2001:DB8::1/64","2001:DB8::2/64","2001:DB8::3/64","2001:DB8::4/64"
NETRESTART

IPV6_STATIC_ADDR_ALLMODULE?
> 2001:DB8::1/64,2001:DB8::2/64,2001:DB8::3/64,2001:DB8::4/64
```

IPV6_STATIC_DEFAULTROUTER

IPv6 Default Router

機能

固定 IPv6 設定のデフォルト・ルータ・アドレスを設定，または問い合わせます。

コマンド

```
IPV6_STATIC_DEFAULTROUTER ip6addr
```

クエリ

```
IPV6_STATIC_DEFAULTROUTER?
```

レスポンス

ip6addr

パラメータ

ip6addr	デフォルト・ルータ・アドレス xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

詳細

MT8870A に装着されている全モジュールにデフォルト・ルータ・アドレスを設定します。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
パラメータに"" (空文字) を指定すると，デフォルト・ルータ・アドレスの設定を初期化できます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，デフォルト・ルータ・アドレスは 0 に変更されます。

使用例

IPV6_STATIC_DEFAULTROUTER "2001:DB8::1"
NETRESTART

IPV6_STATIC_DEFAULTROUTER?
> 2001:DB8::1

IPV6_TYPE

IPv6 Type

機能

IPv6 アドレスの種別を設定，または問い合わせます。

コマンド

```
IPV6_TYPE ipv6type
```

クエリ

```
IPV6_TYPE?
```

レスポンス

ipv6type

パラメータ

ipv6type	IPv6 アドレスの種別
STATIC	固定 IP です。
RA_EUI64	EUI-64 (RFC4862) に基づく RA による自動割り当てです。
RA_RANDOM	プライバシ拡張（RFC4941）に基づく RA による自動割り当てです。
DHCPV6	DHCP サーバからアドレスを取得して設定します。
初期値	STATIC

詳細

MT8870A の全スロットに IPv6 アドレスの種別を設定します。
STATIC を設定した場合は，次のコマンドでアドレスを設定します。
IPV6_STATIC_ADDR_ALLMODULE
IPV6_STATIC_DEFAULTROUTER

本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。
IPv6 アドレスの種別は，MT8870A のスロットに対して設定します。
モジュールを別のスロットに移動すると，そのスロットの IPv6 アドレスの種別が適用されます。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると，IPv6 アドレスの種別は初期値に変更されます。

使用例

IPV6_TYPE RA_EUI64
NETRESTART

IPV6_TYPE?
> RA_EUI64

IPVER

IP Version

機能

IPv4, IPv6 のオン/オフを設定, または問い合わせます。
オンに設定すると, その IP Version でモジュールと通信できます。

コマンド

```
IPVER ipver
```

クエリ

```
IPVER?
```

レスポンス

ipver

パラメータ

ipver	IP Version
V4	IPv4 のみオンにします。
V6	IPv6 のみオンにします。
V4V6	IPv4, IPv6 ともにオンにします。
初期値	V4

詳細

MT8870A に装着されているすべてのモジュールに IP バージョンのオン/オフを設定します。
本コマンドの設定を有効にするには, NETRESTART コマンドを送信します。
MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると, IP Version は初期値に変更されます。

使用例

IPVER V4V6
NETRESTART

再起動後
IPVER?
> V4V6

備考

*RST, REBOOT コマンドで初期化されません。

KEYCHECK?

License Key Check

機能

ライセンスのチェックを行い，正常であればインストール準備を行います。

クエリ

```
KEYCHECK? filename
```

レスポンス

result

パラメータ

filename	ライセンスファイル名
result	ライセンスファイルのチェック結果
–1	同じ本体の別モジュールで実行したためインストールできない。
0	正常
1	指定したファイルが見つからない。
2	ファイルが違う。
3	不正なファイル または 別の MT8870 用のファイル。
4	すでにライセンスが存在する。
5	ライセンス期間外
6	書き込めるライセンス数の上限に達している。

詳細

事前に FTP によりソフトウェアライセンスファイルを指定フォルダに転送する必要があります。転送先フォルダは，"/install/license/" です。
KEYINST コマンドによりインストールする前に，再起動したり電源を切ったりした場合は，インストール準備は無効になります。

使用例

```
KEYCHECK? "Licenses.xml"
>0
KEYINST
```

KEYINFO?

License Information

機能

ライセンスの期限情報を問い合わせます。

クエリ

```
KEYINFO? model
```

レスポンス

kind,days

パラメータ

model	ライセンス形名 ・ソフトウェアライセンス ・ソフトウェアオプションライセンス ・SG 波形ライセンス を指定することができます。
kind	ライセンスの種類
0	Normal ライセンス
1	期限付きライセンス
2	トライアルライセンス
days	残り日数
1 以上	ライセンスの残り日数 1 の場合は，ライセンス最終日です。
0	ライセンスの期限切れ
–1	期限無し(Normal ライセンスの場合)

詳細

ライセンス形名には，ライセンスの期限を問い合わせる製品の形名（オプションを含む）を指定します。
例: MX887011A,MX887012A

使用例

KEYINFO? "MX887012A"
>0,-1

KEYINST

License Key Install

機能

ライセンスをインストールします。

コマンド

```
KEYINST
```

詳細

事前に KEYCHECK?コマンドでインストール準備をする必要があります。
本コマンド実行後は，システム再起動により全モジュールのリモートの接続が切断されます。

使用例

```
KEYCHECK? "Licenses.xml"
>0
KEYINST
```

備考

ハードウェアオプションライセンスは，モジュールに保持されます。オプションの違いにより，同一 MT8870A 内の特定モジュールだけアプリケーションが起動しない場合があります。
ソフトウェアライセンス，ソフトウェアオプションライセンス，SG 波形ライセンスは，MT8870A に保持されます。モジュールを他の MT8870A に移動した場合，ライセンスの違いによりアプリケーションが起動しない場合があります。

LOGCLEAR

Log Clear

機能

ログを消去します。

コマンド

```
LOGCLEAR
```

詳細

モジュールのログファイルを消去します。

使用例

```
LOGCLEAR
```

LOGREAD?

Log Reading

機能

ログ情報を問い合わせます。

クエリ

```
LOGREAD? [line]
```

レスポンス

#XY<Y byte binary data><terminator>

X および Y はアスキー文字で，数値 X は数値 Y の桁数です。Y が 12500 の場合，X は 5 になります。
バイナリデータ<Y byte binary data>は，ログファイル内の文字列です。

パラメータ

line	最新のログ取得行数
範囲	1～999999
terminator	ターミネータ

詳細

パラメータでログ取得行数を指定した場合は，その行数分のログを取得します。ログ取得サイズを省略した場合は，すべてのログを取得します。
ログファイル内の改行コードは，タブに置き換えて表示されます。

使用例

LOGREAD?
> CELLULAR Measurement Software License expired.¥n

LOGREAD? 2
> CELLULAR Measurement Software License expired.¥n
> R[*IDN?]¥n

LOGSIZE?

Log Size Reading

機能

ログサイズを問い合わせます。

クエリ

```
LOGSIZE?
```

レスポンス

size

パラメータ

size	ログのサイズ

詳細

レスポンスは，ログファイルのバイトサイズです。

使用例

```
LOGSIZE?
> 48
```

LOSSTBL

Loss Table

機能

損失補正テーブルのインデックスを設定，または問い合わせます。

コマンド

```
LOSSTBL index
```

クエリ

```
LOSSTBL?
```

レスポンス

```
index
```

パラメータ

index	補正テーブルのインデックス
範囲	1～16
分解能	1
初期値	1

使用例

LOSSTBL 1
LOSSTBL?
> 1

LOSSTBLCNT?

List Count in Loss Table

機能

損失補正テーブルに設定されているリストの数を問い合わせます。

クエリ

```
LOSSTBLCNT? index
```

レスポンス

num

パラメータ

index		補正対象とするテーブルのインデックス番号
	範囲	1～16
	分解能	1
num		レベル周波数補正テーブルに登録されているリストの数
	範囲	0～1000

使用例

LOSSTBLCNT? 1
> 3

LOSSTBLVAL

Register Loss Value into Cable Loss List

機能

損失補正テーブルのリストに値を設定/問い合わせします。
損失補正テーブルのリスト内に同一周波数がある場合は上書きされます。
[Port3_4/in]の設定を省略した場合，[Port3_4/out]の値を損失補正テーブルのリストに設定します。

損失補正テーブルのリストは，任意の周波数ポイントの損失値を入力し，設定ポイント間の周波数について線形補間により自動的に補正値を決定します。
Port 1 と Port 2 の補正値は同じ値が設定されます。Port 3 と Port 4 の補正値も同じ値が設定されます。

コマンド

```
LOSSTBLVAL frequency,Port1_2/out,Port1_2/in,Port3_4/out[,Port3_4/in]
```

クエリ

```
LOSSTBLVAL? n
```

レスポンス

frequency,Port1_2/out,Port1_2/in,Port3_4/out,Port3_4/in

パラメータ

frequency	設定する周波数ポイント
範囲	10 MHz～7300 MHz
分解能	1 Hz
サフィックスコード	HZ, Z, KHZ, KZ, MHZ, MZ, GHZ, GZ 省略した場合は HZ として扱われます。
Port1_2/out	Port1/2 の DL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
Port1_2/in	Port1/2 の UL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
Port3_4/out	Port3/4 の DL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
Port3_4/in	Port3/4 の UL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
n	問い合わせたい補正テーブルポイント
範囲	1～1000

分解能　　1

詳細

複数の Port から信号出力する場合，Port 間の損失補正テーブル値の差が 12 dB を超えると信号は設定した出力レベルよりも低く出力されます。この場合，ランプ 2 が橙色に点灯します。
ただし，8 dB を超えた場合でも，出力レベルの誤差が大きくなる可能性があります。

使用例

損失補正テーブルのリストに次の表の値を設定します。

No	周波数	Port1 (dB)		Port2 (dB)		Port3 (dB)		Port4 (dB)	
		out	in	out	in	out	in	out	in
1	1000000000	10.00	11.00	10.00	11.00	12.00	13.00	12.00	13.00

LOSSTBLVAL 1GHz,10,11,12,13
LOSSTBLVAL? 1
> 1000000000,10.00,11.00,12.00,13.00

備考

損失補正テーブルのリストは，各測定ソフトウェア共通の補正値として保持されます。
周波数に対して昇順となるように，損失補正テーブルのリストは自動で並び替わります。

LOSSTBLVALALL

Register Loss Value into All Cable Loss List

機能

損失補正テーブルのリストに値を設定/問い合わせします。
損失補正テーブルのリスト内に同一周波数がある場合は上書きされます。

損失補正テーブルのリストは，任意の周波数ポイントの損失値を入力し，設定ポイント間の周波数について線形補間により自動的に補正値を決定します。
Port 1～Port 12 で別々の補正値を設定できます。

コマンド

```
LOSSTBLVALALL
frequency,Port1out,Port1in,Port2out,Port2in,Port3out,Port3in,Port4out,Port4in[,Port5out,Port5in,Port6out,Port6in,Port7out,Port7in,Port8out,Port8in,Port9out,Port9in,Port10out,Port10in,Port11out,Port11in,Port12out,Port12in]
```

クエリ

```
LOSSTBLVALALL? n
```

レスポンス

frequency,Port1out,Port1in,Port2out,Port2in,Port3out,Port3in,Port4out,Port4in[,Port5out,Port5in,Port6out,Port6in,Port7out,Port7in,Port8out,Port8in,Port9out,Port9in,Port10out,Port10in,Port11out,Port11in,Port12out,Port12in]

パラメータ

frequency	設定する周波数ポイント
範囲	10 MHz～7300 MHz
分解能	1 Hz
サフィックスコード	HZ, Z, KHZ, KZ, MHZ, MZ, GHZ, GZ 省略した場合は HZ として扱われます。
Port1out	Port 1 の DL の損失値
…	…
Port12out	Port 12 の DL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
Port1in	Port 1 の UL の損失値
…	…
Port12in	Port 12 の UL の損失値
範囲	–100.00～100.00 dB
分解能	0.01 dB
サフィックスコード	DB 省略した場合は DB として扱われます。
n	問い合わせたい補正テーブルポイント
範囲	1～1000
分解能	1

詳細

Port 1～4 の設定は必須です。損失値を設定しない Port は，0 dB とします。
Port 5～12 は，Port 5～12 すべて設定しない場合のみ省略できます。損失値を設定する場合は，すべてに設定します。損失値を設定しない Port は，0 dB とします。
複数の Port から信号出力する場合，Port 間の損失補正テーブル値の差が 12 dB を超えると信号は設定した出力レベルよりも低く出力されます。この場合，ランプ 2 が橙色に点灯します。
ただし，8 dB を超えた場合でも，出力レベルの誤差が大きくなる可能性があります。

使用例

損失補正テーブルのリストに次の表の値を設定します。

No	周波数	Port 1 (dB)		Port 2 (dB)		Port 3 (dB)		Port 4 (dB)	
		out	in	out	in	out	in	out	in
1	1000000000	10.00	11.00	12.00	13.00	14.00	15.00	16.00	17.00
		Port 5 (dB)		Port 6 (dB)		Port 7 (dB)		Port 8 (dB)	
		out	in	out	in	out	in	out	in
		18.00	19.00	20.00	21.00	22.00	23.00	24.00	25.00
		Port 9 (dB)		Port 10 (dB)		Port 11 (dB)		Port 12 (dB)	
		out	in	out	in	out	in	out	in
		26.00	27.00	28.00	29.00	30.00	31.00	32.00	33.00

LOSSTBLVALALL 1GHZ,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33
LOSSTBLVALALL? 1
> 1000000000,10.00,11.00,12.00,13.00,14.00,15.00,16.00,17.00,18.00,19.00,20.00,21.00,22.00,23.00,24.00,25.00,26.00,27.00,28.00,29.00,30.00,31.00,32.00,33.00

備考

損失補正テーブルのリストは，各測定ソフトウェア共通の補正値として保持されます。
周波数に対して昇順となるように，損失補正テーブルのリストは自動で並び替わります。

MACADDR?

MAC Address

機能

MAC アドレスを問い合わせます。

クエリ

```
MACADDR?
```

レスポンス

macaddr

パラメータ

macaddr	MAC アドレス

詳細

通信先モジュールの MAC アドレスを問い合わせます。

使用例

MACADDR?
> 00-00-00-00-00-00

MCFV?

Firmware Version

機能

ファームウェアバージョンを問い合わせます。

クエリ

```
MCFV?
```

レスポンス

ver

パラメータ

ver	ファームウェアバージョン（"xx.xx.xx"で記述）

使用例

MCFV?
> 1.00.00

MCOPT?

Option Number

機能

ソフトウェアのオプション番号を問い合わせます。

クエリ

```
MCOPT? model
```

レスポンス

n,num,...,num

パラメータ

model	ソフトウェア形名
n	オプションの数
num	オプション番号

使用例

下記がインストールされている場合

- MX887011A
- オプション無し

MCOPT? MX887011A
> 0

備考

オーダリング情報に登録されている形名のみを取得できます。
無効なソフトウェア形名をパラメータに設定した場合はパラメータエラーとなります。

MCSOPT?

Type Name of Software

機能

ソフトウェア形名を問い合わせます。

クエリ

```
MCSOPT?
```

レスポンス

n,model,...,model

パラメータ

n	ソフトウェア数
model	ソフトウェア形名

使用例

MCSOPT?
> 3,MX887010A,MX887011A,MX887012A

備考

オーダリング情報に登録されている形名のみを取得できます。

MCWOPT?

Type Name of Waveform

機能

波形ライセンスを問い合わせます。

クエリ

```
MCWOPT?
```

レスポンス

n,model,...,model

パラメータ

n	波形ライセンス数
model	波形ライセンス形名

使用例

MCWOPT?
> 2,MV887011A,MV887012A

備考

オーダリング情報に登録されている形名のみを取得できます。

MFINFO?

Main Frame Information

機能

MT8870A の情報を問い合わせます。

クエリ

```
MFINFO? [index]
```

レスポンス

パラメータが 1 の場合

model name	品名（30 文字以内）

パラメータが 2 の場合

model number	形名（20 文字以内）

パラメータが 3 の場合

BIOS version	常に 0 です。

パラメータが 4 の場合

OS version	常に 0 です。

パラメータが 5 の場合

number of HW,name/rev,name/rev,...,name/rev

number of HW:	ハードウェアの数
name:	ハードウェア名称（30 文字以内, ""でくくられる）
rev:	ハードウェアレビジョン（10 文字以内）

パラメータが 6 の場合

number of FPGA,name/ver,name/ver,...,name/ver

number of FPGA:	FPGA の数
name:	FPGA 名称（30 文字以内, ""でくくられる）
ver:	FPGA バージョン（10 文字以内）

パラメータが 7 の場合

number of HW option,num/name/sw,num/name/sw,...,num/name/sw

number of HW option:	ハードウェアオプションの数
num:	オプション番号（0～999）
name:	オプション名称（50 文字以内, ""でくくられる）
sw:	オプションスイッチ
ON:	オプション有り
OFF:	オプション無し

パラメータが 8 の場合

number of Software license
model/serial/name/ver,model/serial/name/ver,...,model/serial/name/ver

number of Software license:	ソフトウェアライセンスの数
model:	ソフトウェア形名（30 文字以内）
serial:	シリアル番号（10 文字）
name:	ソフトウェアライセンス名称（50 文字以内, ""でくくられる）
ver:	ソフトウェアバージョン（"xx.xx"で記述）

パラメータが 9 の場合

number of SW option,num/name/sw,num/name/sw,...,num/name/sw

number of SW option:	ソフトウェアオプションの数

num:　　オプション形名（20 文字以内）例)MX887000A-001
name:　　オプション名称（50 文字以内, ""でくくられる)
sw:　　オプションスイッチ
ON:　　オプション有り
OFF:　　オプション無し

パラメータが 10 の場合

number of SG wave license
,model/serial/name/maxver,model/serial/name/maxver,...,model/serial/name/maxver
number of SG wave license:　　SG 波形ライセンスの数
model:　　SG 波形ライセンス形名（30 文字以内）
serial:　　シリアル番号（10 文字)
name:　　SG 波形ライセンス名称（50 文字以内, ""でくくられる)
maxver:　　使用可能な最大バージョン（0～999)

パラメータが 0, または省略した場合

上記のすべてのレスポンスをコンマで連結した文字列

パラメータ

index　　取得情報のインデックス
範囲　　0～10

使用例

```
MFINFO? 1
> Universal Wireless Test Set
MFINFO? 2
> MT8870A
MFINFO? 3
> 1.0
MFINFO? 4
> 2.6
MFINFO? 5
> 2,"Main Board"/01,"Option Board"/FE
MFINFO? 6
> 2,"High Speed Control FPGA"/01,"Wide Band FPGA"/FF
MFINFO? 7
> 2,001/"Rubidium"/ON,009/"GPIB Interface"/ON
MFINFO? 8
> 2,MX887010A/6123456789/"Cellular Measurement Software"/01.01,
MX887020A/6123456789/"Vector SG Software"/01.00
MFINFO? 9
> 2,MX887000A-001/"Extended Digitizing Option"/ON,
MX887000A-002/"Phase Noise Measurement"/OFF
MFINFO? 10
> 1,MX887070A/6123456789/"Standard wave license"/5
```

関連コマンド

SYSINFO?

備考

インストールされている期限付きライセンスの期限を過ぎている場合は，本コマンドでライセンス情報を取得できません。

MFMODEL?

Main Frame Model Number.

機能

MT8870A のモデル番号を問い合わせます。

クエリ

```
MFMODEL?
```

レスポンス

model

パラメータ

model	モデル番号

使用例

```
MFMODEL?
> MT8870A
```

MFPOWERONCNT?

Chassis Power ON Count

機能

MT8870A で電源を投入した回数を問い合わせます。

クエリ

```
MFPOWERONCNT?
```

レスポンス

num

パラメータ

num	MT8870A の電源投入回数

使用例

```
MFPOWERONCNT?
> 1971
```

MFSERIAL?

Main Frame Serial No.

機能

MT8870A のシリアル番号を問い合わせます。

クエリ

```
MFSERIAL?
```

レスポンス

serial

パラメータ

serial	シリアル番号 10 桁の数字です。　例 6200112233

使用例

```
MFSERIAL?
> 6123456789
```

MFTEMP?

Main Frame Temperature

機能

MT8870A の温度を問い合わせます。

クエリ

`MFTEMP?`

レスポンス

temp

パラメータ

temp	MT8870A の温度（°C）
分解能:	0.01

使用例

MFTEMP?
> 47.12

MFTMCNT?

Main Frame Running Time Count Read

機能

MT8870A の積算稼働時間を問い合わせます。

クエリ

`MFTMCNT?`

レスポンス

time

パラメータ

time	積算稼働時間（分単位）

使用例

MFTMCNT?
> 1234

MFTRIGPORT

Set In/Out of Trigger Connector

機能

MT8870A 背面パネルのトリガコネクタの入出力を設定，または問い合わせます。

コマンド

```
MFTRIGPORT trigger
```

クエリ

```
MFTRIGPORT?
```

レスポンス

trigger

パラメータ

trigger	トリガコネクタの入出力設定
INPUT	入力
OUTPUT	出力
初期値	INPUT

詳細

モジュールの TRX ごとにトリガコネクタの入出力を設定できます。

次のコマンドによってトリガコネクタの入出力が変更されることがあります。

SOUR:GPRF:GEN:ARB:WAV:SYNC:STAR:PATT:SEL

SOUR:GPRF:GEN:SEQ:COMB:PATT:SYNC

使用例

MFTRIGPORT INPUT

MFTRIGPORT?

> INPUT

NETRESTART

Network Restart

機能

ネットワークを再起動します。

コマンド

NETRESTART

詳細

本コマンドを送信すると，システムが再起動します。再起動中は全モジュールのリモート制御ができません。
リモート制御を続ける場合は，本コマンドの送信後，しばらくしてからリモートを再接続してください。

使用例

NETRESTART

備考

*RST，および REBOOT では通信設定が更新されません。

PORT

Set Direction of RF Connector

機能

RF 信号を入力するテストポートと出力するテストポートを設定，または問い合わせます。

コマンド

```
PORT input,output[,output,…,output]
```

クエリ

```
PORT?
```

レスポンス

input,output[,output,…,output]

パラメータ

input	テストポート番号
範囲	OFF, Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
OFF	RF 信号を入力するポートなし。
初期値	PORT1
output	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
初期値	PORT1

詳細

Test Port 1～Test Port 12 は入力と出力の両方に設定できます。

MU887002A-007 搭載時，以下の制限事項があります。設定コマンドは有効ですが実行エラーとなります。

- テストポート 1～4 は 6 GHz を超える RF 周波数に設定できません。
- テストポート 5～12 は 5.9 GHz 以上の RF 周波数では，入力または出力のどちらかにしか設定できません。

使用例

RF 信号を入力するテストポートを Test Port 1，RF 信号を出力するテストポートを Test Port 2 に設定します。

```
PORT PORT1,PORT2
PORT?
> PORT1,PORT2
```

RF信号入力テストポートなし，RF信号出力テストポートをTest Port 4，Test Port 6，Test Port 10に設定します。

```
:ROUT:PORT:CONN:DIR OFF,PORT10,PORT4,PORT6
:ROUT:PORT:CONN:DIR?
> OFF,PORT4,PORT6,PORT10
```

PORTSETTING

Raw Socket Port Number

機能

RawSocket 通信時のポート番号を設定，または問い合わせます。

コマンド

```
PORTSETTING port
```

クエリ

```
PORTSETTING?
```

レスポンス

port　　ポート番号

パラメータ

port	ポート番号
範囲	5001～60000
初期値	56001

詳細

MT8870A と RawSocket 通信を行う場合の TCP/IP のポート番号を設定します。
ポート番号は IPv4 および IPv6 で共通です。
本コマンドの設定を有効にするには，NETRESTART コマンドを送信します。

使用例

PORTSETTING 56001
NETRESTART
PORTSETTING?
> 56001

REBOOT

Reboot

機能

システムを再起動します。

コマンド

REBOOT

詳細

本コマンドを送信すると，システムが再起動します。再起動中は MT8870A に装着されている全モジュールがリモート制御できません。
本コマンドの送信後，MT8870A の Error ランプが消灯すると，リモート制御ができます。

使用例

REBOOT

関連コマンド

SHUTDOWN

REF

Frequency Reference

機能

周波数基準信号源を設定，または問い合わせます。

コマンド

```
REF source
```

クエリ

```
REF?
```

レスポンス

source_res

パラメータ

source	内部基準信号／外部基準信号
10MHZINT	MT8870A の内部基準信号
10MHZEXT	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
13MHZEXT	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
AUTO	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
source_res	内部基準信号／外部基準信号
10MHZINT	MT8870A の内部基準信号
AUTO	背面パネルの Ref Input コネクタに入力される基準信号（検出されない場合は，内部基準信号を使用）
初期値	AUTO

詳細

MT8870A に装着されている全モジュールに基準信号源を設定します。モジュールごとに基準信号源を設定できません。

使用例

```
REF 10MHZINT
REF?
> 10MHZINT
```

関連コマンド

FREQREF

備考

*RST コマンドで初期化されません。

SHUTDOWN

Shutdown

機能

システムをシャットダウンします。

コマンド

SHUTDOWN

詳細

本コマンドを送信すると，全モジュールのリモートの接続が切断されます。

使用例

SHUTDOWN

関連コマンド

REBOOT

SYSERR?

System Error

機能

エラー・イベントキューに存在するエラーメッセージを問い合わせます。

クエリ

SYSERR?

レスポンス

code,msg

パラメータ

code	エラーコード
0	"No Error"
–113	"Undefined header"
–200	"Execution error"
–220	"Parameter error"
–310	"Option license is required to execute the command"
–370	"Main frame upgrade is required to execute the command"
msg	エラー・イベント番号に対応したイベントメッセージ

詳細

エラー・イベントキューが空の場合のレスポンスは，"0,"No Error""です。

使用例

SYSERR?
> -220, "Parameter error"

SYSERRALL?

System All Errors

機能

エラー・イベントキューに存在するエラーメッセージを問い合わせます。

クエリ

```
SYSERRALL?
```

レスポンス

code,msg,code,msg,.....

パラメータ

code	エラーコード
0	"No Error"
–113	"Undefined header"
–200	"Execution error"
–220	"Parameter error"
–310	"Option license is required to execute the command"
–370	"Main frame upgrade is required to execute the command"
msg	エラー・イベント番号に対応したイベントメッセージ

詳細

エラー・イベントキューが空の場合のレスポンスは，"0,"No Error""です。
実行後，エラー・イベントキューは空になります。

使用例

SYSERRALL?
> -113, "Undefined header",-220,"Parameter error"

SYSERRCNT?

System Error Count

機能

エラー・イベントキューに存在するすべてのエラー・イベントの数を問い合わせます。

クエリ

SYSERRCNT?

レスポンス

num

パラメータ

num	エラー・イベント数

詳細

エラー・イベントキューは最大 4 個までキューイングします。
エラーが 5 個以上発生した場合は，最も古いエラーが削除されます。
SYSERR?，SYSERRCODE?コマンドで，最も古いエラーから順に取得し，エラー・イベントキューは1つ削除されます。
SYSERRALL?，SYSERRCODEALL?コマンドで，すべてのエラーを取得した後，エラー・イベントキューは空になります。

使用例

SYSERRCNT?
> 3

SYSERRCODE?

System Error Code

機能

エラー・イベントキューに存在するエラーコードを問い合わせます。

クエリ

SYSERRCODE?

レスポンス

code

パラメータ

code	エラーコード
0	"No Error"
–113	"Undefined header"
–200	"Execution error"
–220	"Parameter error"
–310	"Option license is required to execute the command"
–370	"Main frame upgrade is required to execute the command"

詳細

エラー・イベントキューが空の場合のレスポンスは，0 です。

使用例

SYSERRCODE?
> -220

SYSERRCODEALL?

System All Error Codes

機能

エラー・イベントキューに存在するすべてのエラーコードを問い合わせます。

クエリ

```
SYSERRCODEALL?
```

レスポンス

code,code,....

パラメータ

code	エラーコード
0	"No Error"
–113	"Undefined header"
–200	"Execution error"
–220	"Parameter error"
–310	"Option license is required to execute the command"
–370	"Main frame upgrade is required to execute the command"

詳細

エラー・イベントキューが空の場合のレスポンスは，0 です。

実行後，エラー・イベントキューは空になります。

使用例

SYSERRCODEALL?

> -113,-220

SYSINFO?

Module Information

機能

モジュールの情報を問い合わせます。

クエリ

```
SYSINFO? [index]
```

レスポンス

パラメータが 1 の場合

model name　品名（30 文字以内）

パラメータが 2 の場合

model number　形名（20 文字以内）

パラメータが 3 の場合

BIOS version　BIOS (IPL) version (15 文字以内)

パラメータが 4 の場合

OS version　OS version (255 文字以内)

パラメータが 5 の場合

number of HW,name/rev,name/rev,...,name/rev

number of HW:　ハードウェアの数

name:　ハードウェア名称（30 文字以内, ""でくくられる)

rev:　ハードウェアレビジョン（10 文字以内)

パラメータが 6 の場合

number of FPGA,name/ver,name/ver,...,name/ver

number of FPGA:　FPGA の数

name:　FPGA 名称（30 文字以内, ""でくくられる)

ver:　FPGA バージョン（"xx.xx.xx"で記述)

パラメータが 7 の場合

number of HW option,num/name/sw,num/name/sw,...,num/name/sw

number of HW option: ハードウェアオプションの数

num:　オプション番号（0～999)

name:　オプション名称（50 文字以内, ""でくくられる)

sw:　オプションスイッチ

ON:　オプション有り

OFF:　オプション無し

パラメータが 8 の場合

常に 0 です。

パラメータが 9 の場合

常に 0 です。

パラメータが 10 の場合

常に 0 です。

パラメータが 0, または, 省略した場合

上記のすべてのレスポンスをコンマで連結した文字列

パラメータ

index	取得情報のインデックス
範囲	0～10

使用例

SYSINFO? 1
> TRX Test Module
SYSINFO? 2
> MU887002A
SYSINFO? 3
> 3.01.10.5.11
SYSINFO? 4
> CentOS Linux release 7.4.1708 (Core) /Linux 3.10.0-693.el7.x86_64
SYSINFO? 5
> 5,"Carrier Board"/00, "BB Board"/01, "RF Board"/01,"Divider1 Board"/01,"Divider2 Board"/01
SYSINFO? 6
> 4,"BB FPGA"/00.00.92,"RF FPGA"/07.00.00,"Divider1 FPGA"/16.00.00,"Divider2 FPGA"/16.00.00
SYSINFO? 7
> 1, 007/"7GHz Frequency Extension"/ON
SYSINFO? 8
> 0
SYSINFO? 9
> 0
SYSINFO? 10
> 0

関連コマンド

MFINFO?

備考

インストールされている期限付きライセンスの期限を過ぎている場合は，本コマンドでライセンス情報を取得できません。

SYSLASTERROR?

System Last Error

機能

前回終了時の異常を問い合わせます。

クエリ

SYSLASTERROR?

レスポンス

msg

パラメータ

msg	異常内容
NO ERROR	異常なし
TemperatureAlarm	温度異常により強制シャットダウン

詳細

本コマンドを実行しても前回終了時の異常の情報はクリアされません。

使用例

SYSLASTERROR?
>NO ERROR

SYSMODEL?

Module Model Number

機能

モジュールのモデル番号を問い合わせます。

クエリ

SYSMODEL?

レスポンス

model

パラメータ

model	モデル番号

使用例

SYSMODEL?
> MU887002A

SYSPLUGCNT?

Count of Module Insertion

機能

モジュールの挿抜回数を問い合わせます。

クエリ

SYSPLUGCNT?

レスポンス

num

パラメータ

num	モジュールの挿抜回数

詳細

注:

モジュールを別のスロットに移動した場合に，挿抜回数がカウントされます。
同じ MT8870A の同じスロットにモジュールを抜き差しした場合は，挿抜回数がカウントされません。

使用例

SYSPLUGCNT?
> 40

SYSPOWERONCNT

Module Power ON Count

機能

モジュールの電源がオンになった回数を問い合わせます。

クエリ

SYSPOWERONCNT?

レスポンス

num

パラメータ

num	モジュールの電源投入回数

使用例

SYSPOWERONCNT?
> 9

SYSSERIAL?

Module Serial No.

機能

モジュールのシリアル番号を問い合わせます。

クエリ

SYSSERIAL?

レスポンス

serial

パラメータ

serial	モジュールのシリアル番号 10 桁の数字です。 例 6200103056

使用例

SYSSERIAL?
> 6123456789

備考

MU887002A の場合, TRX1, TRX2 ともモジュールのシリアル番号は同じ値が返ります。

SYSSLOT?

Module Slot Number

機能

モジュールが装着されているスロット番号を問い合わせます。

クエリ

SYSSLOT?

レスポンス

slot

パラメータ

slot	スロット番号
範囲	1～4

使用例

SYSSLOT?
> 1

SYSTEMDEFAULT

User Area Delete

機能

ユーザエリアを削除します。

コマンド

SYSTEMDEFAULT

詳細

FTP を使用してアクセスできるモジュール内部の記憶媒体上のデータを削除します。
波形ファイルなど /user 以下にあるファイルがすべて削除されます。

使用例

SYSTEMDEFAULT

SYSTEMP?

Module Temperature

機能

モジュールの温度を問い合わせます。

クエリ

SYSTEMP?

レスポンス

temp	モジュールの温度（℃）
分解能:	0.01

使用例

SYSTEMP?
> 47.12

SYSTEST?

System Test

機能

モジュールの自己診断結果と実行した日時を問い合わせます。

クエリ

```
SYSTEST?
```

レスポンス

status1,status2,status3,... ,year,month,day,hour,min
status の詳細は以下のようなフォーマットになります。

"Chassis EEPROM: xx","Module EEPROM: xx","RAM: xx","Correction ROM: xx","FPGA Configuration: xx"

xx には"Pass"または"Fail"が入ります。

パラメータ

status	最後実行したセルフテストの結果
year	年 (2001～)
month	月 (1～12)
day	日 (1～31)
hour	時 (0～23)
min	分 (0～59)

使用例

SYSTEST?
> "Chassis EEPROM: Pass","Module EEPROM: Pass","RAM: Pass","Correction ROM: Pass","FPGA Configuration:Pass",2012,4,5,12,34

SYSTESTERRCNT?

System Test Error Count

機能

モジュールの自己診断を失敗した回数を問い合わせます。

クエリ

```
SYSTESTERRCNT?
```

レスポンス

count

パラメータ

count	モジュールの自己診断が失敗した回数
範囲	0 以上

使用例

SYSTESTERRCNT?
> 0

SYSTESTERR_TM?

System Test Error Last Time

機能

最後にモジュールの自己診断を失敗した日時を問い合わせます。

クエリ

```
SYSTESTERR_TM?
```

レスポンス

year,month,day,hour,min

パラメータ

year	年 (2001～)
month	月 (1～12)
day	日 (1～31)
hour	時 (0～23)
min	分 (0～59)

使用例

SYSTESTERR_TM?
> 2012,1,2,3,45

備考

モジュールの自己診断を 1 度も失敗していない場合, レスポンスは"***"になります。

SYST:LANG

Language Selection of Remote Command

機能

リモート制御コマンドの言語モードを切り替えます。

コマンド

```
SYST:LANG mode
```

クエリ

```
SYST:LANG?
```

レスポンス

mode

パラメータ

mode	言語モード
NAT	Native
SCPI	SCPI

使用例

SYST:LANG NAT
SYST:LANG?
>NAT

SYSTMCNT?

Module Running Time Count Read

機能

モジュールの積算稼働時間を問い合わせます。

クエリ

```
SYSTMCNT?
```

レスポンス

time

パラメータ

time	積算稼働時間（分単位）

使用例

SYSTMCNT?
> 1234

SYSTRIGINSRC

Select Source of Trigger Input

機能

外部トリガとして使用する信号を設定，または問い合わせます。

コマンド

```
SYSTRIGINSRC source
```

クエリ

```
SYSTRIGINSRC?
```

レスポンス

source

パラメータ

source	外部トリガとして使用する信号
TRGIN	制御対象のモジュールの TRX が挿入されているスロット用トリガコネクタに入力されるトリガ
TRGIN1	トリガコネクタ 1 に入力されるトリガ
SLOT1	スロット 1 のモジュールの TRX から出力されるトリガ信号
初期値	TRGIN

詳細

TRGIN を選択する際は，Trigger コネクタの入出力設定を入力にしてください。
次のコマンドによって外部トリガとして使用する信号が変更されることがあります。

SOUR:GPRF:GEN:ARB:WAV:SYNC:STAR:PATT:SEL
SOUR:GPRF:GEN:SEQ:COMB:PATT:SYNC

使用例

SYSTRIGINSRC TRGIN1
SYSTRIGINSRC?
> TRGIN1

関連コマンド

MFTRIGPORT

SYSTRIGOUTSRC

Select Source of Trigger Output

機能

MT8870A 背面パネルのトリガコネクタに出力する信号を設定，または問い合わせます。

コマンド

```
SYSTRIGOUTSRC source
```

クエリ

```
SYSTRIGOUTSRC?
```

レスポンス

source

パラメータ

source	トリガコネクタに出力する信号
WFM1	SG 波形マーカ 1
WFM2	SG 波形マーカ 2
WFM3	SG 波形マーカ 3
SGSYNC	SG 同期開始信号
NONE	出力信号なし（初期値）
初期値	NONE

詳細

Trigger コネクタの入出力設定を出力設定にした際に,本コマンドで選択した信号が Trigger コネクタに出力されます。

次のコマンドによってトリガコネクタに出力する信号が変更されることがあります。

SOUR:GPRF:GEN:ARB:WAV:SYNC:STAR:PATT:SEL

SOUR:GPRF:GEN:SEQ:COMB:PATT:SYNC

使用例

SYSTRIGOUTSRC WFM1

SYSTRIGOUTSRC?

> WFM1

関連コマンド

MFTRIGPORT

SYSTRX?

TRX 1/2

機能

TRX 1/2 を問い合わせます。

クエリ

```
SYSTRX?
```

レスポンス

trx

パラメータ

trx	TRX 1/2
1	TRX 1
2	TRX 2

使用例

SYSTRX?
> 1

SYSVER?

Package Version

機能

パッケージバージョンを問い合わせます。

クエリ

```
SYSVER?
```

レスポンス

ver

パラメータ

ver	パッケージバージョン（"xx.xx.xx"で記述）

使用例

SYSVER?
> 01.00.00

TRM

Terminator of Message

機能

ターミネータ (メッセージ末尾に付加される文字) を設定, または問い合わせます。

コマンド

```
TRM code
```

クエリ

```
TRM?
```

レスポンス

code 終端コード

パラメータ

code	終端コード
LF	LF (Line Feed)
CRLF	CR/LF (Carriage Return and Line Feed)
NONE	None (EOI のみ)
初期値	CRLF

詳細

モジュールごとに設定できます。

MT8870A のパネルの IP Reset ボタンを押しながら電源を投入すると, 終端コードは初期値に変更されます。

使用例

TRM CRLF
TRM?
>CRLF

備考

DELM と同じ処理をします。

*RST, REBOOT, NETRESTART コマンドで初期化されません。

イーサネット接続でパラメータを NONE に設定した場合, 本器からのレスポンスメッセージはターミネータが CRLF に設定されています。

UPDATELOWERLIMIT?

Lower Limit of Firmware Update

機能

ファームウェアアップデートの下限バージョンを問い合わせます。

クエリ

```
UPDATELOWERLIMIT?
```

レスポンス

lowerlimit

パラメータ

lowerlimit	ファームウェアアップデートの下限バージョン（"xx.xx.xx"で記述）

詳細

MU887002Aは，V3.00.00以降のファームウェアパッケージでなければインストールできません。
MU887002A-007搭載時は，V4.00.00以降のファームウェアパッケージでなければインストールできません。

使用例

UPDATELOWERLIMIT?
> 04.00.00

この章では，MU887002A の性能試験に必要な測定機器，セットアップ，性能試験手順について説明します。
性能試験結果の記録には，「付録 C 性能試験記録表」を使用してください。

7.1 概要 7-2
7.2 性能試験用機器一覧表 7-3
7.3 各測定に対する性能試験詳細 7-8
7.3.1 VSWR 7-8
7.3.2 SGレベル確度 7-12
7.3.3 SGスプリアス 7-22
7.3.4 SAレベル確度 7-25
7.3.5 SAリニアリティ 7-33
7.3.6 パスロス測定確度 (MX887092A) 7-41
7.4 サービスについて 7-44

7.1 概要

性能試験は，MU887002A の性能劣化を未然に防止するため，保守の一環として行います。性能試験は，MU887002A の受け入れ検査，定期検査，修理後の性能確認などに利用してください。重要と判断される項目は，予防保守として定期的に行ってください。この節では，以下の性能試験について説明します。

- VSWR
- SG レベル確度
- SG スプリアス
- SA レベル確度
- SA リニアリティ
- パスロス測定確度（MX887092A）

定期試験の推奨繰り返し期間は，年に 1～2 回程度です。

性能試験で規格を満足しない項目を発見した場合，本書（紙版説明書では巻末，電子版説明書では別ファイル）に記載の「本製品についてのお問い合わせ窓口」へすみやかにご連絡ください。

⚠ 注意

被測定装置と測定器類は，特に指示する場合＊ を除き少なくとも 30 分間は予熱を行い，十分に安定してから性能試験を行ってください。最高の測定確度を発揮するには，上記のほかに室温下での実施，AC 電源電圧の変動が少ないこと，騒音・振動・ほこり・湿気などについても全く問題がないことが必要です。

性能試験の前に，MU887002A のレベル校正（フルバンド校正）を実行してください。レベル校正に関するコマンドは，「3.5.2 レベル校正」を参照してください。

＊：MU887002A を購入後，初めてお使いになるとき，および長期間使用されなかったときはレベル確度に影響を与えるおそれがあるため，十分に予熱してからお使いください。

7.2 性能試験用機器一覧表

次の表に性能試験で使用する測定機器一覧を示します。

表7.2-1　性能試験用機器一覧表 (MU887002A-007 非搭載時)

機器名	要求される性能*1	推奨機器
信号発生器	・周波数範囲: 400～6000 MHz ・周波数設定分解能: 100 kHz ・出力レベル範囲: –30～0 dBm ・外部基準入力: (10 MHz) ・コネクタ: N 型	ベクトル信号発生器 (MG3710E) 上限周波数 6 GHz (MG3710E-036)
パワーメータ	・本体確度: ±0.02 dB ・周波数範囲: 400～6000 MHz ・表示分解能: 0.01 dB	パワーメータ (ML2438A)
パワーセンサ	・周波数範囲: 400～6000 MHz ・測定電力範囲: –40～+20 dBm ・入力コネクタ: N 型	パワーセンサ (MA2442D)
固定減衰器 (6 dB)	・周波数範囲: 400～6000 MHz ・減衰量: 6 dB ・コネクタ: N 型	固定減衰器 (BW-N6W5+)
固定減衰器 (3 dB)	・周波数範囲: 400～6000 MHz ・減衰量: 3 dB ・コネクタ: N 型	固定減衰器 (BW-N3W5+)
中継アダプタ	・周波数範囲: 400～6000 MHz 100～8000 MHz*2 ・コネクタ: N-J・N-J	(34NFNF50)
パスロス測定用同軸ケーブル*2	・周波数範囲: 400～6000 MHz ・コネクタ: N-P・N-J ・ケーブル長: 1.5 m ・比誘電率: 2.03	
パスロス測定用 OPEN/SHORT*2	・周波数範囲: 400～6000 MHz ・コネクタ: N-P	(22N50)
シグナルアナライザ	・周波数範囲: 400～6000 MHz ・周波数設定分解能: 1 Hz ・測定電力範囲: –140～+20 dBm ・外部基準入力: (10 MHz)	シグナルアナライザ (MS269xA, または MS2830A-041)

＊1: 試験項目の測定範囲をカバーできる性能の一部を抜粋

＊2: パスロス測定確度の性能試験に必要

表7.2-1　性能試験用機器一覧表 (MU887002A-007 非搭載時) (続き)

機器名	要求される性能*1	推奨機器
ネットワーク アナライザ	・ 周波数範囲: 400～6000 MHz ・ 入力コネクタ: N 型	ネットワークアナライザ (MS2026C)
	・ 周波数範囲: 100～8000 MHz*2 ・ 入力コネクタ: N 型	ネットワークアナライザ (S820E) 8GHz サイトマスタ (S820E-0708) ベクトルネットワークアナライザ (S820E-0440)
アプリケーションソフトウェア		セルラ規格用シーケンス測定 (MX887010A)
		MU887002A パスロス測定機能*2 (MX887092A)

表7.2-2　性能試験用機器一覧表 (MU887002A-007 搭載時)

機器名	要求される性能*1	推奨機器
信号発生器	・周波数範囲: 400～7300 MHz ・周波数設定分解能: 100 kHz ・出力レベル範囲: –30～0 dBm ・外部基準入力: (10 MHz) ・コネクタ: K 型	信号発生器 (MG3692C) アナログダウンコンバータ (8 MHz～2 GHz) (MG3690C/5)
パワーメータ	・本体確度: ±0.02 dB ・周波数範囲: 400～7300 MHz ・表示分解能: 0.01 dB	パワーメータ (ML2438A)
パワーセンサ	・周波数範囲: 400～7300 MHz ・測定電力範囲: –40～+20 dBm ・入力コネクタ: N 型	パワーセンサ (MA2442D)
固定減衰器 (6 dB)	・周波数範囲: 400～7300 MHz ・減衰量: 6 dB ・コネクタ: N 型	固定減衰器 (BW-N6W5+)
固定減衰器 (3 dB)	・周波数範囲: 400～7300 MHz ・減衰量: 3 dB ・コネクタ: N 型	固定減衰器 (BW-N3W5+)
中継アダプタ	・周波数範囲: 400～7300 MHz 100～8000 MHz*2 ・コネクタ: N-J・N-J	(34NFNF50)
変換アダプタ	・周波数範囲: 400～7300 MHz ・コネクタ: K-P・N-J	(34RKNF50)
パスロス測定用同軸ケーブル*2	・周波数範囲: 400～6000 MHz ・コネクタ: N-P・N-J ・ケーブル長: 1.5 m ・比誘電率: 2.03	
パスロス測定用 OPEN/SHORT*2	・周波数範囲: 400～6000 MHz ・コネクタ: N-P	(22N50)
シグナルアナライザ	・周波数範囲: 400～7300 MHz ・周波数設定分解能: 1 Hz ・測定電力範囲: –140～+20 dBm ・外部基準入力: (10 MHz)	シグナルアナライザ (MS2830A-044) マイクロ波プリセレクタバイパス (MS2830A-067) マイクロ波帯プリアンプ (MS2830A-068)

＊1: 試験項目の測定範囲をカバーできる性能の一部を抜粋

＊2: パスロス測定確度の性能試験に必要

表7.2-2　性能試験用機器一覧表 (MU887002A-007 搭載時) (続き)

機器名	要求される性能*1	推奨機器
ネットワーク アナライザ	・ 周波数範囲: 400～7400 MHz 100～8000 MHz*2 ・ 入力コネクタ: N 型	ネットワークアナライザ (S820E) 8GHz サイトマスタ (S820E-0708) ベクトルネットワークアナライザ (S820E-0440)
アプリケーションソフトウェア		セルラ規格用シーケンス測定 (MX887010A)
		MU887002A パスロス測定機能*2 (MX887092A)

表7.2-3　性能試験項目と使用する機器

機器名	性能試験項目					
	VSWR	SG レベル確度	SG スプリアス	SA レベル確度	SA リニアリティ	パスロス 測定確度
信号発生器		✓		✓	✓	
パワーメータ		✓		✓	✓	
パワーセンサ		✓		✓	✓	
固定減衰器（6 dB）		✓				
固定減衰器（3 dB）				✓	✓	
中継アダプタ		✓		✓	✓	✓
変換アダプタ*		✓		✓	✓	
パスロス測定用同軸ケーブル						✓
パスロス測定用 OPEN/SHORT						✓
シグナルアナライザ		✓	✓	✓	✓	
ネットワークアナライザ	✓					✓
アプリケーションソフトウェア	✓	✓		✓	✓	✓

＊：　MU887002A-007 搭載時

7.3 各測定に対する性能試験詳細

7.3.1　VSWR

ここでは，MU887002A のコネクタに信号を入力した場合の VSWR（電圧定在波比）に関する試験をします。

(1)　試験対象規格

MU887002A-007 非搭載時 TRX 1/2，Port 1～Port 12，20～30°C にて
MU887002A-007 搭載時 TRX 1/2，Port 1～Port 4，20～30°C にて

周波数 f	規格
400 MHz≦f＜450 MHz	＜1.4
450 MHz≦f≦2700 MHz	＜1.3
2700 MHz＜f≦3800 MHz	＜1.4
3800 MHz＜f≦6000 MHz	＜1.4

MU887002A-007 搭載時 TRX 1/2，Port 5～Port 12，20～30°C にて

周波数 f	規格
400 MHz≦f＜450 MHz*1	＜1.4
450 MHz≦f≦2700 MHz*1	＜1.3
2700 MHz＜f≦3800 MHz*1	＜1.4
3800 MHz＜f＜6000 MHz*1	＜1.4
5800 MHz≦f≦7400 MHz*2	＜1.7

＊1: 設定周波数＜5900 MHz
＊2: 設定周波数≧5900 MHz

(2)　試験用測定器

・ ネットワークアナライザ:　MS2026C (MU887002A-007 非搭載時)
　　S820E (MU887002A-007 搭載時)

・ 校正キット

(3)　セットアップ

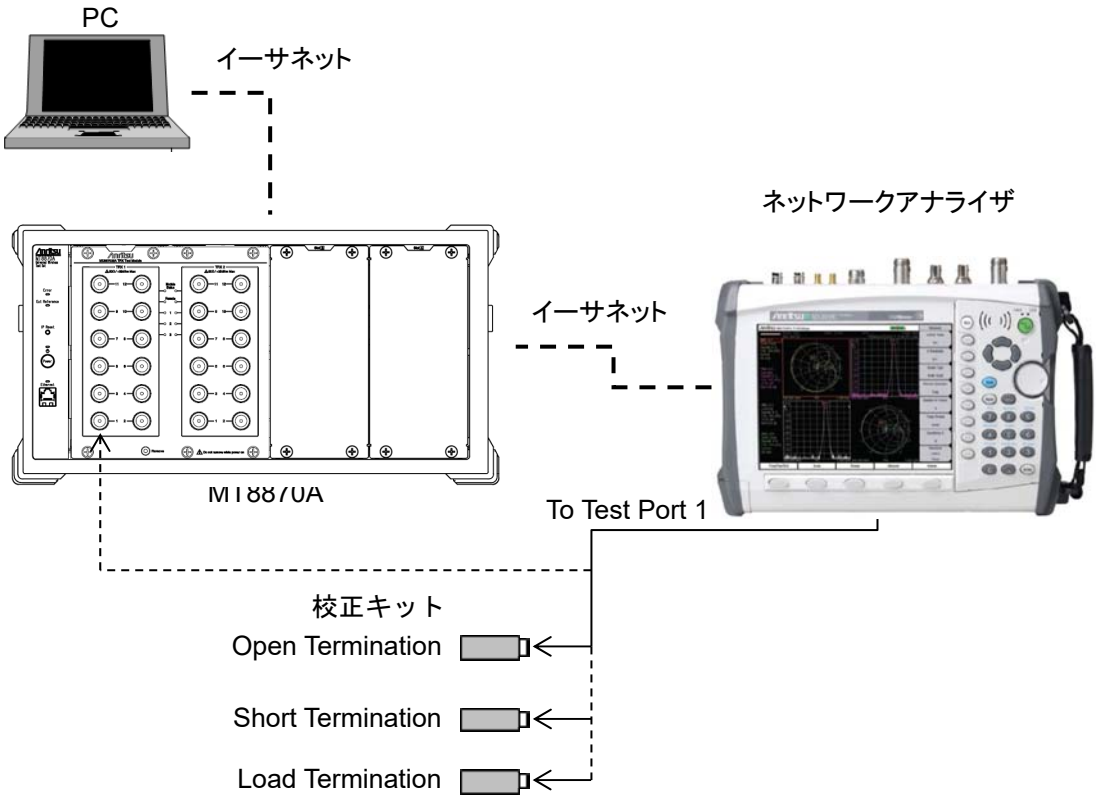


図7.3.1-1　VSWR の試験系

(4) a　試験手順（MU887002A-007 非搭載時　Port 1～12,
MU887002A-007 搭載時　Port 1～4,
または MU887002A-007 搭載時　Port 5～12,
設定周波数＜5900 MHz）

VSWR は，TRX 1/2，Port 1～Port 12 を試験します。
TRX 1 Port 1 以外を試験する場合は，
TRX 1 Port 1 の場合の "TRX 1 Port 1" を読み替えてください。

TRX 1 Port 1 の場合

1. ネットワークアナライザに，MU887002A へ接続するための N 型ケーブルを接続します。
2. ネットワークアナライザを，以下のように設定します。
   - Start Frequency:　400 MHz
   - Stop Frequency:　6000 MHz
3. N 型ケーブルの先端にて，Open，Short，Load ターミネーションを使用して S_{11} (VSWR) 測定の校正をします。
4. N 型ケーブルを MU887002A の TRX 1 Port 1 に接続します。
5. MU887002A の設定を初期化します。
   ```
   *RST
   ```
6. ゼロ出力波形（zerowave.wvd, zerowave.xml）をロードします。
   ゼロ出力波形はアンリツホームページのダウンロードセンターから入手できます。
7. MU887002A の入力ポート，出力ポートを TRX 1 Port 1 に設定します。
   ```
   :ROUTe:PORT:CONNect:DIRection PORT1,PORT1
   ```
8. MU887002A の入力レベルを–65 dBm に設定します。
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:LEVel -65
   ```
9. MU887002A の入力周波数を 5000 MHz に設定します。
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:FREQuency
   5000MHZ
   ```
10. VSG の動作モードを Normal に設定します。
    ```
    :SOURce:GPRF:GENerator:MODE NORMAL
    ```
11. VSG の変調を ARB に設定します。
    ```
    :SOURce:GPRF:GENerator:BBMode ARB
    ```
12. VSG の出力レベルを–10 dBm に設定します。
    ```
    :SOURce:GPRF:GENerator:RFSettings:LEVel -10
    ```
13. VSG の出力周波数を 5000 MHz に設定します。
    ```
    :SOURce:GPRF:GENerator:RFSettings:FREQuency
    5000MHZ
    ```
14. VSG の信号出力をオンにします。
    ```
    :SOURce:GPRF:GENerator:STATe ON
    ```
15. ネットワークアナライザで S_{11} (VSWR) を測定します。

(4) b　試験手順（MU887002A-007 搭載時，Port 5～Port 12
設定周波数≧5900 MHz）

VSWR は，TRX 1/2，Port 5～Port 12 を試験します。
TRX 1 Port 5 以外を試験する場合は，
TRX 1 Port 5 の場合の "TRX 1 Port 5" を読み替えてください。

TRX 1 Port 5 の場合

1. ネットワークアナライザに，MU887002A へ接続するための N 型ケーブルを接続します。
2. ネットワークアナライザを，以下のように設定します。
   - Start Frequency:　400 MHz
   - Stop Frequency:　7400 MHz
3. N 型ケーブルの先端にて，Open，Short，Load ターミネーションを使用して S_{11} (VSWR) 測定の校正をします。
4. N 型ケーブルを MU887002A の TRX 1 Port 5 に接続します。
5. MU887002A の設定を初期化します。
   ```
   *RST
   ```
6. ゼロ出力波形（zerowave.wvd，zerowave.xml）をロードします。
   ゼロ出力波形はアンリツホームページのダウンロードセンターから入手できます。
7. MU887002A の入力ポートを TRX 1 Port 5 に，出力ポートを TRX 1 Port 6 に設定します（入力ポートを TRX 1 Port 6 に設定するときは出力ポートを TRX1 Port 5 に設定します）。
   ```
   :ROUTe:PORT:CONNect:DIRection PORT5,PORT6
   ```
8. MU887002A の入力レベルを–4 dBm に設定します。
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:LEVel -4
   ```
9. MU887002A の入力周波数を 6000 MHz に設定します。
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:FREQuency
   6000MHZ
   ```
10. VSG の動作モードを Normal に設定します。
    ```
    :SOURce:GPRF:GENerator:MODE NORMAL
    ```
11. VSG の変調を ARB に設定します。
    ```
    :SOURce:GPRF:GENerator:BBMode ARB
    ```
12. VSG の出力レベルを–10 dBm に設定します。
    ```
    :SOURce:GPRF:GENerator:RFSettings:LEVel -10
    ```
13. VSG の出力周波数を 6000 MHz に設定します。
    ```
    :SOURce:GPRF:GENerator:RFSettings:FREQuency
    6000MHZ
    ```
14. VSG の信号出力をオフにします。
    ```
    :SOURce:GPRF:GENerator:STATe OFF
    ```
15. ネットワークアナライザで S_{11} (VSWR) を測定します。

7.3.2　SGレベル確度

ここでは，VSG 出力信号のレベル確度に関する試験をします。

(1)　試験対象規格

MU887002A-007 非搭載時 TRX 1/2，Port 1～Port 12，
MU887002A-007 搭載時 TRX 1/2，Port 1～Port 4
無変調信号（CW），校正実行後，20～30°C にて

周波数 f (MHz)	出力レベル (dBm)	規格
400≦f≦3800	–120≦出力レベル≦–5	±1.0 dB
3800＜f≦6000	–100≦出力レベル≦–8	±1.3 dB

MU887002A-007 搭載時 TRX 1/2，Port 5～Port 12，
無変調信号（CW），校正実行後，20～30°C にて

周波数 f (MHz)	出力レベル (dBm)	規格
400≦f≦3800	–120≦出力レベル≦–5	±1.0 dB
3800＜f＜5900	–100≦出力レベル≦–8	±1.3 dB
5900≦f≦7300	–100≦出力レベル≦–10	±1.3 dB

(2)　試験用測定器

- 信号発生器: MG3710E (MU887002A-007 非搭載時)
  MG3692C (MU887002A-007 搭載時)
- パワーメータ: ML2438A
- パワーセンサ: MA2442D
- シグナルアナライザ: MS269xA，または MS2830A-041
  (MU887002A-007 非搭載時)
  MS2830A-044 (MU887002A-007 搭載時)
- 変換アダプタ: 34RKNF50 (MU887002A-007 搭載時)

(3)　セットアップ

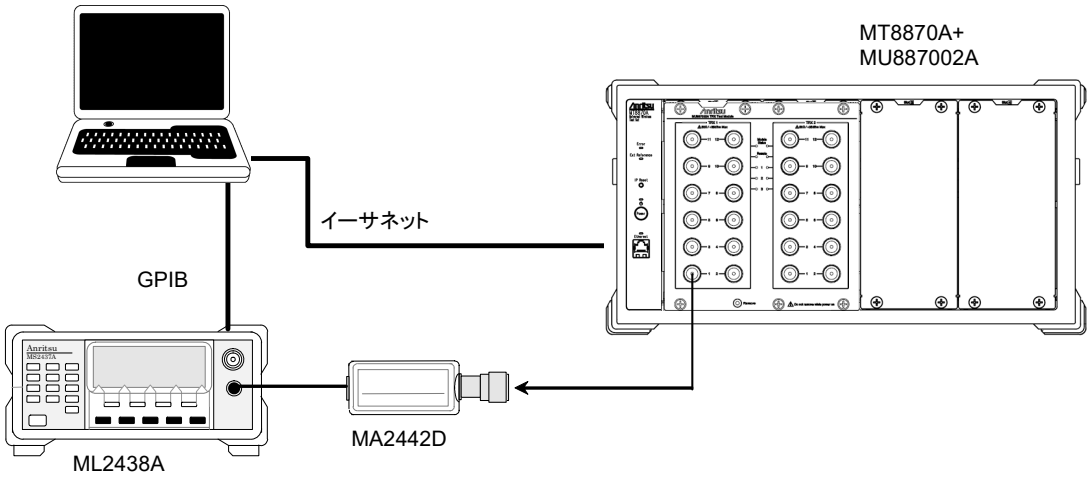


図7.3.2-1　SG レベル確度の試験系 (絶対レベル測定)

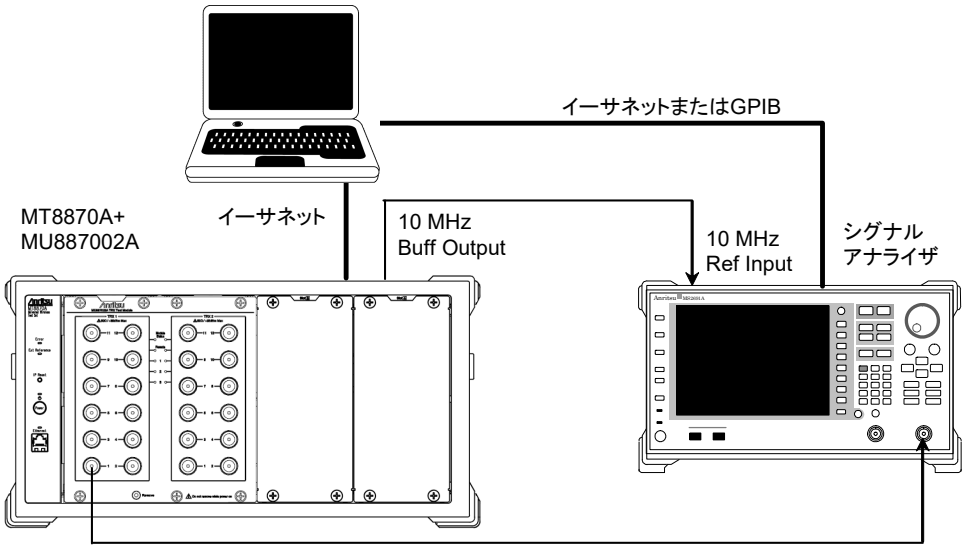


図7.3.2-2　SG レベル確度の試験系 (相対レベル測定)

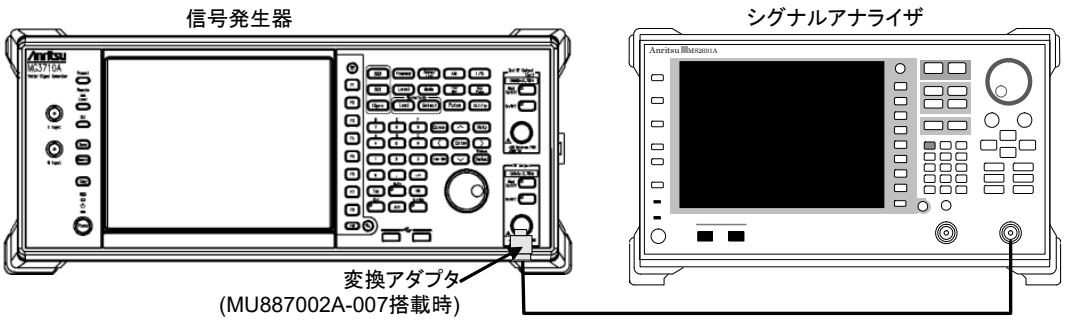


図7.3.2-3　シグナルアナライザの測定レベル校正

SG レベル確度測定の準備

1. パワーセンサのセンサを校正（ゼロ点，感度）します。
2. 「図7.3.2-3　シグナルアナライザの測定レベル校正」のとおり，信号発生器とシグナルアナライザを接続します。
3. 信号発生器の出力レベルを–50 dBm に設定します。
4. 信号発生器の出力周波数を 400.001 MHz に設定します。
5. 「表7.3.2-2　シグナルアナライザの設定」の番号#1の値をシグナルアナライザに設定します。
6. シグナルアナライザで測定したレベルを#1 測定値とします。
7. 「表7.3.2-2　シグナルアナライザの設定」の番号#2の値をシグナルアナライザに設定します。
8. #1 測定値と#2 測定値の差を計算します。この値をシグナルアナライザのレンジ間誤差（D）とします。

   D = (#1 測定値) – (#2 測定値)
9. 「表7.3.2-3　周波数リスト（シグナルアナライザ)」の各周波数を信号発生器に設定して，手順 4～8 を繰り返します。

表7.3.2-1　周波数リスト (パワーメータ)

周波数 (MHz)	周波数 (MHz)	周波数 (MHz)
400.001*1,*2	4072.001*1,*2	5900.001*3
1318.001*1,*2	4990.001*1,*2	6818.001*3
2236.001*1,*2	5908.001*1	7299.999*3
3154.001*1,*2	5999.999*1	–
3800.001*1,*2	–	–

＊1: MU887002A-007 非搭載時 TRX1/2, Port 1～12
　　または MU887002A-007 搭載時 TRX1/2, Port 1～4
＊2: MU887002A-007 搭載時 TRX1/2, Port 5～12
　　設定周波数＜5900 MHz
＊3: MU887002A-007 搭載時 TRX1/2, Port 5～12
　　設定周波数≧5900 MHz

表7.3.2-2　シグナルアナライザの設定

番号	MU887002A 出力レベル 設定値 (dBm)	シグナルアナライザの設定 (マイクロ波プリセレクタバイパス ON)							
		RBW (Hz)	Zone Width (Hz)	Span (kHz)	ATT (dB)	Preamp	Average	Average Count	Ref Lev (dBm)
#1	≦0 dBm ≧–49 dBm	100	781.3	25	20	OFF	OFF	2	0
#2	＜–49 dBm ≧–79 dBm	100	781.3	25	0	ON	OFF	2	–40
#3	＜–79 dBm ≧–100 dBm	10	156.3	2.5	0	ON	OFF	2	–40
#4	＜–100 dBm ≧–110 dBm	10	15.63	1	0	ON	ON	2	–70
#5	＜–110 dBm ≧–120 dBm	3	15.63	1	0	ON	ON	4	–70

表7.3.2-3　周波数リスト (シグナルアナライザ)

信号発生器 出力周波数 (MHz)	シグナルアナライザ 中心周波数 (MHz)	#1 測定値 (dBm)	#2 測定値 (dBm)	D = #1 – #2 (dB)
400.001*1,*2	400.001*1,*2			
1318.001*1,*2	1318.001*1,*2			
2236.001*1,*2	2236.001*1,*2			
3154.001*1,*2	3154.001*1,*2			
3800.001*1,*2	3800.001*1,*2			
4072.001*1,*2	4072.001*1,*2			
4990.001*1,*2	4990.001*1,*2			
5908.001*1	5908.001*1			
5999.999*1	5999.999*1			
5900.001*3	5900.001*3			
6818.001*3	6818.001*3			
7299.999*3	7299.999*3			

＊1: MU887002A-007 非搭載時 TRX1/2, Port 1～12,
または MU887002A-007 搭載時 TRX1/2, Port 1～4

＊2: MU887002A-007 搭載時 TRX1/2, Port 5～12 設定周波数＜5900 MHz

＊3: MU887002A-007 搭載時 TRX1/2, Port 5～12 設定周波数≧5900 MHz

(4) a 試験手順（MU887002A-007 非搭載時 Port 1～12,
MU887002A-007 搭載時 Port1～4,
または MU887002A-007 搭載時 Port 5～12 設定周波数＜5900 MHz）

SG レベル確度は，TRX 1/2，Port 1～Port 12 を試験します。
TRX 1 Port 1 以外を試験する場合は，
TRX 1 Port 1 の場合の "TRX 1 Port 1" を読み替えてください。

TRX 1 Port 1 の場合

絶対レベル測定

1. TRX 1 Port 1 を試験する場合，図7.3.2-1のとおり機器を接続します。
2. MU887002A の設定を初期化します。
   ```
   *RST
   ```
3. 入力ポート，出力ポートを TRX 1 Port 1 に設定します。
   ```
   :ROUTe:PORT:CONNect:DIRection PORT1,PORT1
   ```
4. 入力レベルを–30 dBm に設定します。
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:LEVel
   -30
   ```
5. 入力周波数を 400 MHz に設定します。
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:FREQ
   uency 400MHZ
   ```
6. VSG の動作モードを Normal に設定します。
   ```
   :SOURce:GPRF:GENerator:MODE NORMAL
   ```
7. VSG の変調を CW に設定します。
   ```
   :SOURce:GPRF:GENerator:BBMode CW
   ```
8. VSG の出力レベルを–22 dBm に設定します。
   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel -22
   ```
9. VSG の信号出力をオンにします。
   ```
   :SOURce:GPRF:GENerator:STATe ON
   ```
10. VSG の出力周波数を 400.001 MHz に設定します。
    ```
    :SOURce:GPRF:GENerator:RFSettings:FREQuency
    400.001MHZ
    ```
11. パワーメータで VSG の出力レベルを測定します。
    この測定値を TRX 1 Port 1 の 400.001 MHz の絶対レベル基準値 *C* とします。
12. VSG の出力周波数を「表7.3.2-1　周波数リスト（パワーメータ）」の各周波数に設定して，手順 10～11 を繰り返します。

相対レベル測定

13. 図7.3.2-2 のとおり機器を接続します。
14. VSG の出力周波数を 400.001 MHz に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:FREQuency
400.001MHZ
```

15. VSG の出力レベルを–22 dBm に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -22
```

16. シグナルアナライザでレベルを測定します。
この測定値を *A* dBm とします。
17. VSG の出力レベルを–5 dBm に設定します。
この出力レベル設定値を *P* とします。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -5
```

18. VSG の出力レベルに応じて，シグナルアナライザを「表7.3.2-2　シグナルアナライザの設定」のとおり設定します。
19. シグナルアナライザでレベルを測定します。
この測定値を *B* dBm とします。
20. 次の式でレベル確度を計算します。

レベル確度 = B – A + C – P

21. VSG の出力レベルを 5 dB ステップで–10 dBm, –15 dBm, ..., –115 dBm, –120 dBm と変えて，手順 17～20 を繰り返します。
ただし，表7.3.2-2 番号#2～#5 の条件で測定した場合は，次の式でレベル確度を計算します。*D* はシグナルアナライザのレンジ間誤差です。

レベル確度 = (B + D) – A + C – P

22. VSG の出力周波数を，「表7.3.2-3　周波数リスト（シグナルアナライザ)」の各周波数≦3800 MHz に設定して，手順 14～21 を繰り返します。
23. VSG の出力周波数を 3800.001 MHz に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:FREQuency
3800.001MHZ
```

24. VSG の出力レベルを–22 dBm に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -22
```

25. シグナルアナライザでレベルを測定します。
この測定値を *A* dBm とします。
26. VSG の出力レベルを–8 dBm に設定します。この出力レベル設定値を P とします。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -8
```

27. 設定したレベルに応じて，シグナルアナライザを表7.3.2-2のとおり設定します。

28. シグナルアナライザでレベルを測定します。
この測定値を *B* dBm とします。

29. 次の式でレベル確度を計算します。

レベル確度 = B – A + C – P

30. VSG の出力レベルを 5 dB ステップで–10 dBm, –15 dBm, ..., –95 dBm, –100 dBm と変えて, 手順 28～29 を繰り返します。ただし, 表 7.3.2-2 番号#2～#5 の条件で測定した場合は場合は, 次の式でレベル確度を計算します。D はシグナルアナライザのレンジ間誤差です。

レベル確度 = (B + D) – A + C – P

31. VSG の出力周波数を,「表7.3.2-3　周波数リスト (シグナルアナライザ)」の各周波数に設定して, 手順 23～30 を繰り返します。

(4) b 試験手順（MU887002A-007 搭載時 Port 5～12
設定周波数≧5900 MHz）

SG レベル確度は，TRX 1/2，Port 5～Port 12 を試験します。
TRX 1 Port 5 以外を試験する場合は，
TRX 1 Port 5 の場合の "TRX 1 Port 5" を読み替えてください。

TRX 1 Port 5 の場合

絶対レベル測定

1. TRX 1 Port 5 を試験する場合，図7.3.2-1のとおり機器を接続します。
2. MU887002A の設定を初期化します。
   ```
   *RST
   ```
3. 入力ポートを TRX 1 Port 6 に，出力ポートを TRX 1 Port 5 に設定します（入力ポートを TRX 1 Port 5 に設定するときは出力ポートを TRX 1 Port 6 に設定します)。
   ```
   :ROUTe:PORT:CONNect:DIRection PORT6,PORT5
   ```
4. 入力レベルを–30 dBm に設定します。
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:LEVel -30
   ```
5. 入力周波数を 6000 MHz に設定します。
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:FREQ
   uency 6000MHZ
   ```
6. VSG の動作モードを Normal に設定します。
   ```
   :SOURce:GPRF:GENerator:MODE NORMAL
   ```
7. VSG の変調を CW に設定します。
   ```
   :SOURce:GPRF:GENerator:BBMode CW
   ```
8. VSG の出力レベルを–22 dBm に設定します。
   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel -22
   ```
9. VSG の信号出力をオンにします。
   ```
   :SOURce:GPRF:GENerator:STATe ON
   ```
10. VSG の出力周波数を 5900.001 MHz に設定します。
    ```
    :SOURce:GPRF:GENerator:RFSettings:FREQuency
    5900.001MHZ
    ```
11. パワーメータで VSG の出力レベルを測定します。
    この測定値を TRX 1 Port 5 の 5900.001 MHz の絶対レベル基準値 *C* とします。
12. VSG の出力周波数を「表7.3.2-1　周波数リスト（パワーメータ)」の各周波数に設定して，手順 10～11 を繰り返します。

相対レベル測定

13. 図7.3.2-2 のとおり機器を接続します。
14. VSG の出力周波数を 5900.001 MHz に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:FREQuency
5900.001MHZ
```

15. VSG の出力レベルを–22 dBm に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -22
```

16. シグナルアナライザでレベルを測定します。
この測定値を *A* dBm とします。
17. VSG の出力レベルを–10 dBm に設定します。
この出力レベル設定値を *P* とします。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -10
```

18. VSG の出力レベルに応じて，シグナルアナライザを「表7.3.2-2　シグナルアナライザの設定」のとおり設定します。
19. シグナルアナライザでレベルを測定します。
この測定値を *B* dBm とします。
20. 次の式でレベル確度を計算します。

レベル確度 = B – A + C – P

21. VSG の出力レベルを 5 dB ステップで–15 dBm, ..., –95 dBm, –100 dBm と変えて，手順 17～20 を繰り返します。
ただし，表7.3.2-2 番号#2～#5 の条件で測定した場合は，次の式でレベル確度を計算します。*D* はシグナルアナライザのレンジ間誤差です。

レベル確度 = (B + D) – A + C – P

22. VSG の出力周波数を，「表7.3.2-3　周波数リスト（シグナルアナライザ)」の各周波数に設定して，手順 14～21 を繰り返します。

7.3.3　SGスプリアス

ここでは，VSG 信号のスプリアスレベルに関する以下の試験をします。

(1)　試験対象規格

MU887002A-007 非搭載時 TRX 1/2，Port 1～Port 12，
MU887002A-007 搭載時 TRX 1/2，Port 1～Port 4，
無変調信号（CW），–120 dBm≦出力レベル≦–5 dBm，20～30°C にて

高調波周波数 f (MHz)	規格 (dBc)
800≦f≦6000	高調波ひずみ＜–25

MU887002A-007 搭載時 TRX 1/2，Port 5～Port 12，
無変調信号（CW），–120 dBm≦出力レベル≦–5 dBm，20～30°C にて

高調波周波数 f (MHz)	規格 (dBc)
800≦f≦7300	高調波ひずみ＜–25

(2)　試験用測定器

・ シグナルアナライザ:　MS269xA または MS2830A-041
(MU887002A-007 非搭載時)
MS2830A-044 (MU887002A-007 搭載時)

(3)　セットアップ

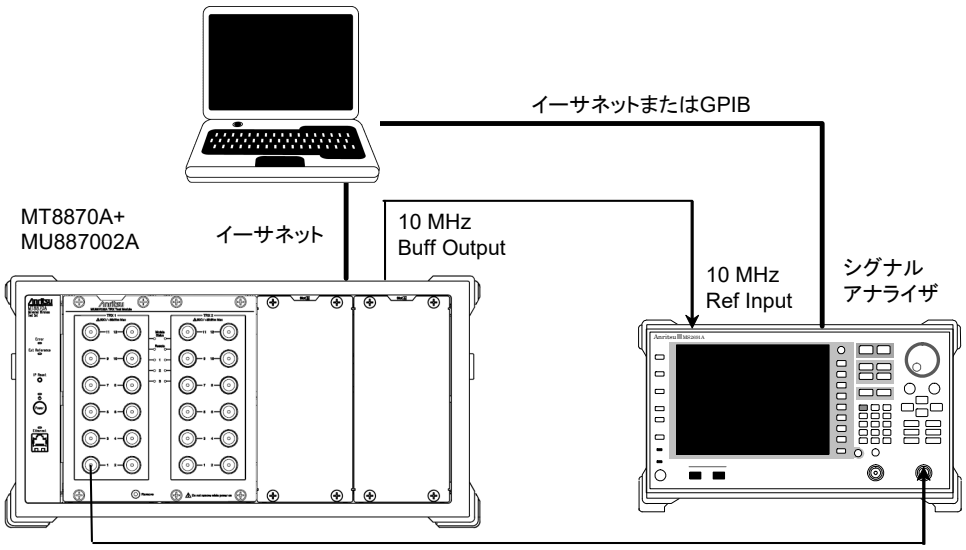


図7.3.3-1　SG スプリアスの試験系

(4)　試験手順

スプリアスは TRX 1/2，Port 1～Port 12 を試験します。
TRX 1 Port 1 以外を試験する場合は，
TRX 1 Port 1 の場合の "TRX 1 Port 1" を読み替えてください。

TRX 1 Port 1 の場合

1. TRX 1 Port 1 を試験する場合，図 7.3.3-1 のように機器を接続します。
2. MU887002A の設定を初期化します。
   ```
   *RST
   ```
3. 入力ポート，出力ポートを TRX 1 Port 1 に設定します。
   ```
   :ROUTe:PORT:CONNect:DIRection PORT1,PORT1
   ```
4. VSG の動作モードを Normal に設定します。
   ```
   :SOURce:GPRF:GENerator:MODE NORMAL
   ```
5. VSG の変調を CW に設定します。
   ```
   :SOURce:GPRF:GENerator:BBMode CW
   ```
6. VSG の出力レベルを–5 dBm に設定します。
   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel -5
   ```
7. VSG の信号出力をオンにします。
   ```
   :SOURce:GPRF:GENerator:STATe ON
   ```
8. VSG の出力周波数を 400 MHz に設定します。
   ```
   :SOURce:GPRF:GENerator:RFSettings:FREQuency 400MHZ
   ```
9. シグナルアナライザで 400 MHz（基本波）の信号のレベルを測定します。この測定値を *A* dBm とします。
10. シグナルアナライザで 800 MHz（2 次高調波）の信号のレベルを測定します。この測定値を *B* dBm とします。ただし，2 次高調波の周波数が 6 GHz を超える場合は測定しません。
    MU887002A-007 搭載時，Port 5～Port 12 については高調波の周波数が 7.3 GHz を超える場合は測定しません。
11. シグナルアナライザで 1200 MHz（3 次高調波）の信号のレベルを測定します。この測定値を *C* dBm とします。ただし，3 次高調波の周波数が 6 GHz を超える場合は測定しません。
    MU887002A-007 搭載時，Port 5～Port 12 について高調波の周波数が 7.3 GHz を超える場合は測定しません。
12. 次の式で 2 次高調波ひずみ，3 次高調波ひずみを計算します。

    2 次高調波ひずみ ＝ B – A (dBc)
    3 次高調波ひずみ ＝ C – A (dBc)
13. VSG の出力周波数を表7.3.3-1に従い周波数を変えて手順 8～12 を繰り返します。

表7.3.3-1　周波数ポイント

番号	MU887002A-007 非搭載時 TRX 1/2, Port 1～Port 12 MU887002A-007 搭載時 TRX 1/2, Port 1～Port 4	MU887002A-007 搭載時 TRX 1/2, Port 5～Port 12
	出力周波数 (MHz)	出力周波数 (MHz)
1	400	400
2	600	600
3	850	850
4	900	900
5	1000	1000
6	1500	1500
7	1800	1800
8	1950	1950
9	2200	2200
10	2450	2450
11	2650	2650
12	3000	3000
13	–	3300
14	–	3650

7.3.4　SAレベル確度

ここでは，MU887002A の SA レベル確度に関する試験をします。

(1)　試験対象規格

MU887002A-007 非搭載時 TRX 1/2，Port 1～Port 12，
MU887002A-007 搭載時 TRX 1/2，Port 1～Port 4，
無変調信号（CW），校正実行後，20～30°C，
測定帯域幅 = 300 kHz，RBW = 100 kHz，
設定周波数に等しい周波数の被測定信号，
設定レベルに等しいレベルの被測定信号にて，

周波数 (MHz) / レベル (dBm)	400≦f≦3800	3800＜f≦6000
–30≦レベル≦+35	±0.5 dB	±0.7 dB
–55≦レベル＜–30	±0.7 dB	±0.9 dB
–65≦レベル＜–55	±0.9 dB	±1.1 dB

MU887002A-007 搭載時 TRX 1/2，Port 5～Port 12，
無変調信号（CW），校正実行後，20～30°C，
測定帯域幅 = 300 kHz，RBW = 100 kHz，
設定周波数に等しい周波数の被測定信号，
設定レベルに等しいレベルの被測定信号にて，

周波数 (MHz) / レベル (dBm)	400≦f≦3800	3800＜f＜5900	5900≦f≦7300
–30≦レベル≦+35	±0.5 dB	±0.7 dB	–
–30≦レベル≦+30	–	–	±1.0 dB
–55≦レベル＜–30	±0.7 dB	±0.9 dB	±1.2 dB
–65≦レベル＜–55	±0.9 dB	±1.1 dB	±1.8 dB

(2)　試験用測定器

- 信号発生器: MG3710E (MU887002A-007 非搭載時)
  MG3692C (MU887002A-007 搭載時)
- パワーメータ: ML2438A
- パワーセンサ: MA2442D
- シグナルアナライザ: MS269xA または MS2830A-041
  (MU887002A-007 非搭載時)
  MS2830A-044 (MU887002A-007 搭載時)
- N 型アッテネータ（6 dB）: BW-N6W5+　1 個
- N 型アッテネータ（3 dB）: BW-N3W5+　1 個
- 中継アダプタ: 34NFNF50　1 個
- 変換アダプタ: 34RKNF50 (MU887002A-007 搭載時)

(3)　セットアップ

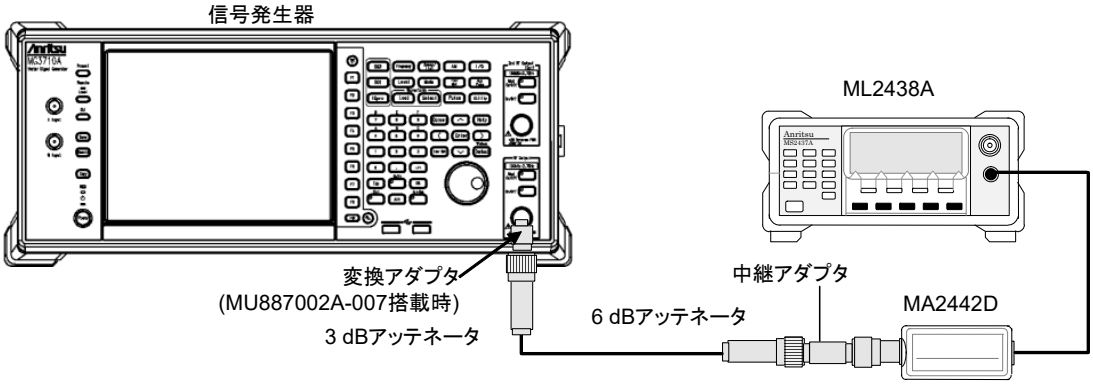


図7.3.4-1　出力レベルの測定 (0, –25, –30 dBm)

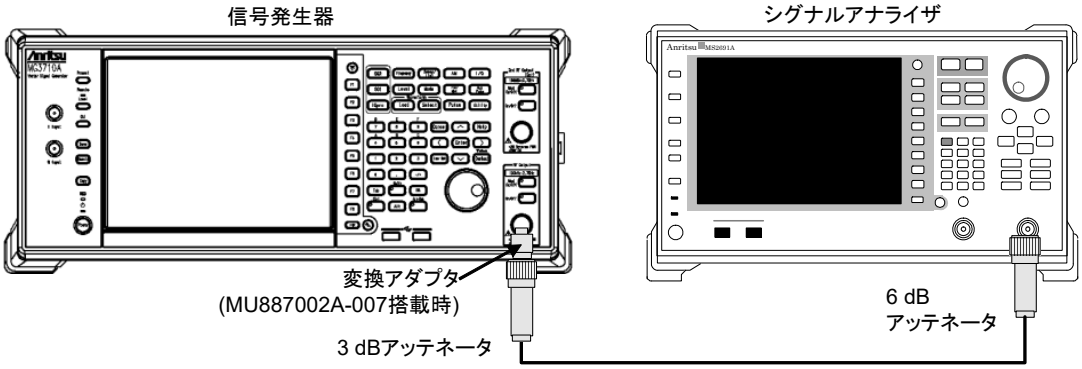


図7.3.4-2　出力レベルの測定 (–35 dBm)

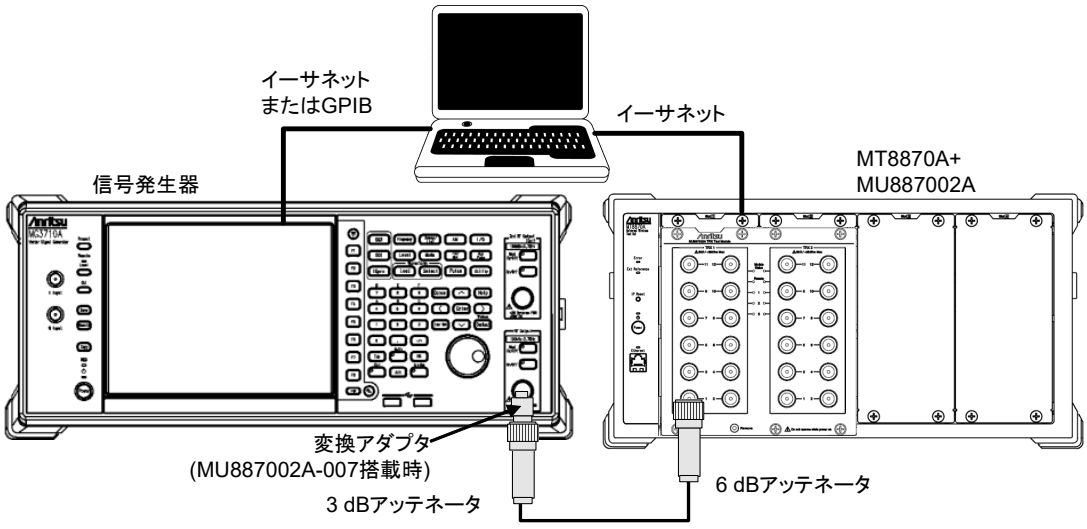


図7.3.4-3　SA レベル確度の試験系 (TRX 1 Port 1)

SA レベル確度測定の準備

1. パワーセンサのセンサを校正（ゼロ点，感度）します。
2. 両端に 3 dB，6 dB アッテネータを付けたケーブルで，信号発生器とパワーセンサを接続します。
3. 信号発生器の周波数を 1317.9 MHz に設定します。
4. パワーメータの表示が 0 dBm に最も近くなるように，信号発生器の出力レベルを調整します。
   このときの信号発生器の設定値を P0，パワーメータ測定値を C0 とします。
5. パワーメータの表示が–25 dBm に最も近くなるように，信号発生器の出力レベルを調整します。
   このときの信号発生器の設定値を P25，パワーメータ測定値を C25 とします。
6. パワーメータの表示が–30 dBm に最も近くなるように，信号発生器の出力レベルを調整します。
   このときの信号発生器の設定値を P30，パワーメータ測定値を C30 とします。
7. 信号発生器の周波数を「表7.3.4-1　周波数リスト」の各周波数に設定し，手順 3～6 を繰り返します。
8. 「図7.3.4-2　出力レベルの測定（–35 dBm）」のとおり，両端に 3 dB，6 dB アッテネータを付けたケーブルで，信号発生器とシグナルアナライザを接続します。
9. 信号発生器の周波数を 1317.9 MHz に設定します。
10. 手順 5 で測定した，出力レベルが–25 dBm となる値（P25）を信号発生器に設定します。
11. シグナルアナライザでレベルを測定します。この測定値を S とします。
12. 信号発生器の出力レベルに，手順 10 で設定した値から 10 dB 低い値（P25–10）を設定します。
13. シグナルアナライザでレベルを測定します。この測定値を S1 とします。
14. 信号発生器の周波数を「表7.3.4-1　周波数リスト」の各周波数に設定し，手順 9～13 を繰り返します。

表7.3.4-1　周波数リスト

番号	MU887002A 受信周波数 (MHz)	信号発生器 設定周波数 (MHz)
1	1318*1,*2	1317.9*1,*2
2	2236*1,*2	2235.9*1,*2
3	3154*1,*2	3153.9*1,*2
4	3800*1,*2	3799.9*1,*2
5	4072*1,*2	4071.9*1,*2
6	4990*1,*2	4989.9*1,*2
7	6000*1	5999.9*1
8	5900*3	5900.1*3
9	6818*3	6817.9*3
10	7300*3	7299.9*3

＊1: MU887002A-007 非搭載時 TRX1/2, Port 1～12
　　または MU887002A-007 搭載時 TRX1/2, Port 1～4
＊2: MU887002A-007 搭載時 TRX1/2, Port 5～12
　　設定周波数＜5900 MHz
＊3: MU887002A-007 搭載時 TRX1/2, Port 5～12
　　設定周波数≧5900 MHz

(4) a 試験手順（MU887002A-007 非搭載時 Port 1～12,
MU887002A-007 搭載時 Port 1～4,
または MU887002A-007 搭載時 Port 5～12 設定周波数＜5900 MHz）

SA レベル確度は，TRX 1/2，Port 1～Port 12 を試験します。
TRX 1 Port 1 以外を試験する場合は，
TRX 1 Port 1 の場合の "TRX 1 Port 1" を読み替えてください。

TRX 1 Port 1 の場合

1. TRX 1 Port 1 を試験する場合，図7.3.4-3のとおり機器を接続します。
2. MU887002A の設定を初期化します。

```
*RST
```

3. 入力ポート，出力ポートを TRX 1 Port 1 に設定します。

```
:ROUTe:PORT:CONNect:DIRection PORT1,PORT1
```

4. VSG の動作モードを Normal に設定します。

```
:SOURce:GPRF:GENerator:MODE NORMAL
```

5. VSG の変調を CW に設定します。

```
:SOURce:GPRF:GENerator:BBMode CW
```

6. VSG の出力レベルを–120 dBm に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -120
```

7. VSG の信号出力をオフにします。

```
:SOURce:GPRF:GENerator:STATe OFF
```

8. MU887002A の測定項目をスペクトラムモニタにします。

```
:INSTrument[:SELect] CELLULAR
:CONFigure:CELLular:MEASurement:STANdard COMMON
:CONFigure:CELLular:MEASurement:SELect SPMON
```

9. スペクトラムモニタを次のとおり設定します。

入力周波数:	「表7.3.4-1　周波数リスト」の番号 1 の周波数
スパン:	50 MHz
パワー測定帯域幅:	300 kHz
RBW:	100 kHz
検波方式:	rms

```
:CONFigure:CELLular:MEASurement:RFSettings:FREQ
uency 1318MHZ
:CONFigure:CELLular:COMMon:SPMonitor:SPAN 50MHZ
:CONFigure:CELLular:COMMon:SPMonitor:PMBW 300KHZ
:CONFigure:CELLular:COMMon:SPMonitor:RBW 100KHZ
:CONFigure:CELLular:COMMon:SPMonitor:DETect RMS
```

10. 信号発生器の出力周波数を 1317.9 MHz に設定します（「表7.3.4-1 周波数リスト」の番号 1)。
11. 信号発生器の出力レベルに「SA レベル確度測定の準備」で求めた P0

を設定します。

12. MU887002A の入力レベルを 0 dBm に設定します。

```
:CONFigure:CELLular:MEASurement:RFSettings:LEVel 0
```

13. MU887002A でレベルを測定します。
この測定値を *A* dBm とします。

```
:INITiate:CELLular:MEASurement:SINGle
*WAI
:FETCh:CELLular:COMMon:SPMonitor:POWer
```

14. 次の式でレベル確度を計算します。C0 は「SA レベル確度測定の準備」で求めた値です。

レベル確度 = A – C0

15. MU887002A の入力レベルを+5 dBm, +10 dBm, +35 dBm と 5 dB ステップで変えて手順 12～14 を繰り返します。

16. 信号発生器の出力レベルに「SA レベル確度測定の準備」で求めた P30 を設定します。

17. MU887002A の入力レベルを–30 dBm に設定します。

```
:CONFigure:CELLular:MEASurement:RFSettings:LEVel -30
```

18. MU887002A でレベルを測定します。

19. 次の式でレベル確度を計算します。C30 は「SA レベル確度測定の準備」で求めた値です。

レベル確度 = A – C30

20. MU887002A の入力レベルを–25 dBm, –20 dBm, …, –5 dBm と 5 dB ステップで変えて手順 17～19 を繰り返します。

21. 信号発生器の出力レベルを P25–10 に設定します。P25 は「SA レベル確度測定の準備」で求めた値です。

22. MU887002A の入力レベルを –35 dBm に設定します。

```
:CONFigure:CELLular:MEASurement:RFSettings:LEVel -35
```

23. MU887002A でレベルを測定します。
この測定値を *A* dBm とします。

24. 次の式でレベル確度を計算します。C25, S, S1 は「SA レベル確度測定の準備」で求めた値です。

レベル確度 = A – C25 – (S1 – S)

25. MU887002A の入力レベルを–40 dBm, –45 dBm, …, –65 dBm と 5 dB ステップで変えて手順 22～24 を繰り返します。

26. 「表7.3.4-1　周波数リスト」の番号 2 以降について, 手順 9～25 を繰り返します。

(4) b 試験手順 (MU887002A-007 搭載時 Port 5～12
設定周波数≧5900 MHz)

SA レベル確度は, TRX 1/2, Port 5～Port 12 を試験します。
TRX 1 Port 5 以外を試験する場合は,
TRX 1 Port 5 の場合の "TRX 1 Port 5" を読み替えてください。

TRX 1 Port 5 の場合

1. TRX 1 Port 5 を試験する場合, 図7.3.4-3のとおり機器を接続します。
2. MU887002A の設定を初期化します。
   ```
   *RST
   ```
3. 入力ポートを TRX 1 Port 5 に, 出力ポートを TRX 1 Port 6 に設定します (入力ポートを TRX 1 Port 6 に設定するときは出力ポートを TRX 1 Port 5 に設定します)。
   ```
   :ROUTe:PORT:CONNect:DIRection PORT5,PORT6
   ```
4. VSG の動作モードを Normal に設定します。
   ```
   :SOURce:GPRF:GENerator:MODE NORMAL
   ```
5. VSG の変調を CW に設定します。
   ```
   :SOURce:GPRF:GENerator:BBMode CW
   ```
6. VSG の出力レベルを–120 dBm に設定します。
   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel -120
   ```
7. VSG の信号出力をオフにします。
   ```
   :SOURce:GPRF:GENerator:STATe OFF
   ```
8. MU887002A の測定項目をスペクトラムモニタにします。
   ```
   :INSTrument[:SELect] CELLULAR
   :CONFigure:CELLular:MEASurement:STANdard COMMON
   :CONFigure:CELLular:MEASurement:SELect SPMON
   ```
9. スペクトラムモニタを次のとおり設定します。
   入力周波数: 「表7.3.4-1　周波数リスト」の番号 8 の周波数
   スパン: 50 MHz
   パワー測定帯域幅: 300 kHz
   RBW: 100 kHz
   検波方式: rms
   ```
   :CONFigure:CELLular:MEASurement:RFSettings:FREQuency 5900MHZ
   :CONFigure:CELLular:COMMon:SPMonitor:SPAN 50MHZ
   :CONFigure:CELLular:COMMon:SPMonitor:PMBW 300KHZ
   :CONFigure:CELLular:COMMon:SPMonitor:RBW 100KHZ
   :CONFigure:CELLular:COMMon:SPMonitor:DETect RMS
   ```
10. 信号発生器の出力周波数を 5900.1 MHz に設定します (「表7.3.4-1 周波数リスト」の番号 8)。

11. 信号発生器の出力レベルに「SA レベル確度測定の準備」で求めた P0 を設定します。

12. MU887002A の入力レベルを 0 dBm に設定します。

    ```
    :CONFigure:CELLular:MEASurement:RFSettings:LEVel 0
    ```

13. MU887002A でレベルを測定します。
    この測定値を *A* dBm とします。

    ```
    :INITiate:CELLular:MEASurement:SINGle
    *WAI
    :FETCh:CELLular:COMMon:SPMonitor:POWer
    ```

14. 次の式でレベル確度を計算します。C0 は「SA レベル確度測定の準備」で求めた値です。

    レベル確度 = A – C0

15. MU887002A の入力レベルを+5 dBm, +10 dBm, +30 dBm と 5 dB ステップで変えて手順 12～14 を繰り返します。

16. 信号発生器の出力レベルに「SA レベル確度測定の準備」で求めた P30 を設定します。

17. MU887002A の入力レベルを–30 dBm に設定します。

    ```
    :CONFigure:CELLular:MEASurement:RFSettings:LEVel -30
    ```

18. MU887002A でレベルを測定します。

19. 次の式でレベル確度を計算します。C30 は「SA レベル確度測定の準備」で求めた値です。

    レベル確度 = A – C30

20. MU887002A の入力レベルを–25 dBm, –20 dBm, …, –5 dBm と 5 dB ステップで変えて手順 17～19 を繰り返します。

21. 信号発生器の出力レベルを P25–10 に設定します。P25 は「SA レベル確度測定の準備」で求めた値です。

22. MU887002A の入力レベルを –35 dBm に設定します。

    ```
    :CONFigure:CELLular:MEASurement:RFSettings:LEVel -35
    ```

23. MU887002A でレベルを測定します。
    この測定値を *A* dBm とします。

24. 次の式でレベル確度を計算します。C25, S, S1 は「SA レベル確度測定の準備」で求めた値です。

    レベル確度 = A – C25 – (S1 – S)

25. MU887002A の入力レベルを–40 dBm, –45 dBm, …, –65 dBm と 5 dB ステップで変えて手順 22～24 を繰り返します。

26. 「表7.3.4-1　周波数リスト」の番号 9 以降について, 手順 9～25 を繰り返します。

7.3.5　SAリニアリティ

ここでは，MU887002A の SA リニアリティに関する試験をします。

(1)　試験対象規格

TRX 1/2，Port 1～Port 12，20～30°C，無変調信号（CW），
測定帯域幅 = 300 kHz，RBW = 100 kHz にて，

入力レベル	入力レベル減衰量	規格
≧–55 dBm	0～–40 dB	±0.2 dB
≧–65 dBm	0～–40 dB	±0.4 dB

(2)　試験用測定器

- 信号発生器: MG3710E (MU887002A-007 非搭載時)
  MG3692C (MU887002A-007 搭載時)
- パワーメータ: ML2438A
- パワーセンサ: MA2442D
- シグナルアナライザ: MS269xA または MS2830A-041
  (MU887002A-007 非搭載時)
  MS2830A-044 (MU887002A-007 搭載時)
- N 型アッテネータ（6 dB）: BW-N6W5+　1 個
- N 型アッテネータ（3 dB）: BW-N3W5+　1 個
- 中継アダプタ: 34NFNF50　1 個
- 変換アダプタ: 34RKNF50 (MU887002A-007 搭載時)

(3)　セットアップ

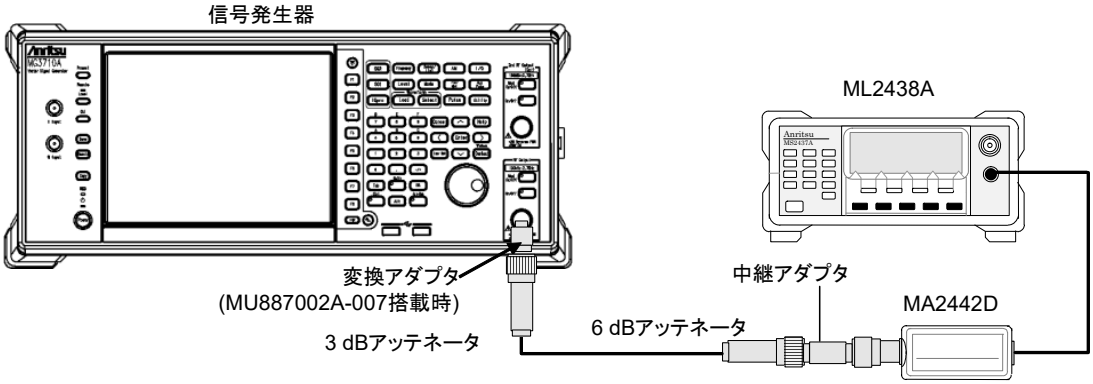


図7.3.5-1　出力レベルの測定 (0, –25 dBm)

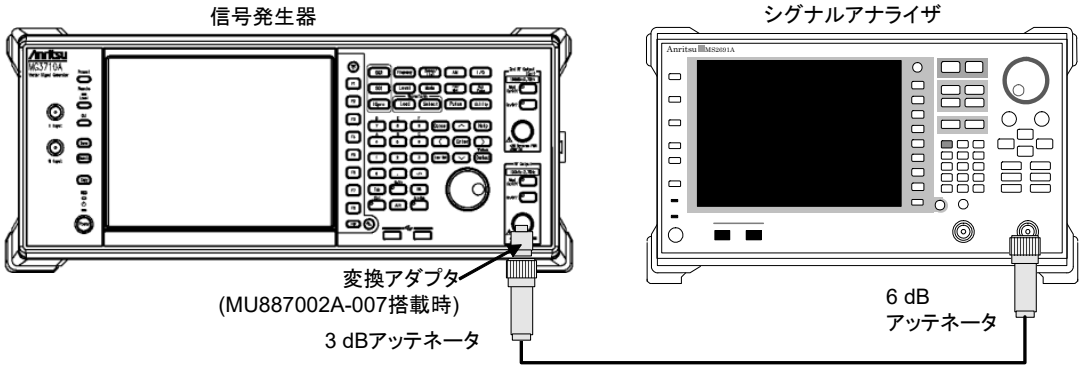


図7.3.5-2　出力レベルの測定 (0～–65 dBm)

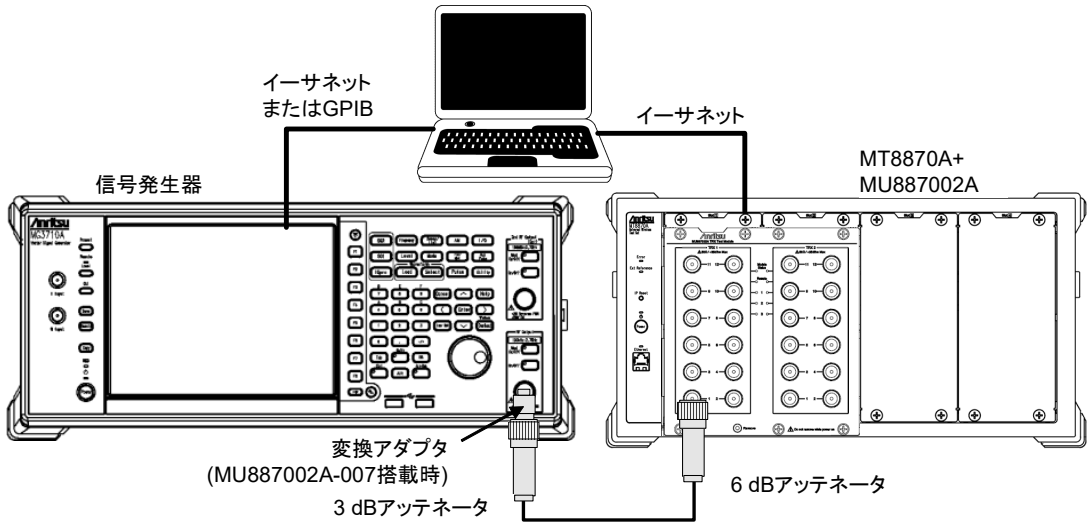


図7.3.5-3　SA リニアリティの試験系 (TRX 1 Port 1)

SA リニアリティ測定の準備

1. パワーセンサのセンサを校正（ゼロ点，感度）します。
2. 両端に 3 dB，6 dB アッテネータを付けたケーブルで，信号発生器とパワーセンサを接続します。
3. 信号発生器の周波数を 1317.9 MHz に設定します。
4. パワーメータの表示が 0 dBm に最も近くなるように，信号発生器の出力レベルを調整します。
   このときの信号発生器の設定値を P0，パワーメータ測定値を C0 とします。
5. パワーメータの表示が–25 dBm に最も近くなるように，信号発生器の出力レベルを調整します。
   このときの信号発生器の設定値を P25，パワーメータ測定値を C25 とします。
6. 信号発生器の周波数を「表7.3.5-1　周波数リスト」の各周波数に設定し，手順 3～5 を繰り返します。
7. 「図7.3.5-2　出力レベルの測定（0～–65 dBm)」のとおり，両端に 3 dB，6 dB アッテネータを付けたケーブルで，信号発生器とシグナルアナライザを接続します。
8. 信号発生器の周波数を 1317.9 MHz に設定します。
9. 手順 4 で測定した，出力レベルが 0 dBm となる値（P0）を信号発生器に設定します。
10. シグナルアナライザでレベルを測定します。この測定値を S とします。
11. 信号発生器の出力レベルに，手順 9 で設定した値から 10 dB 低い値（P0–10）を設定します。
12. シグナルアナライザでレベルを測定します。この測定値を S1 とします。
13. 信号発生器の出力レベルに，手順 9 で設定した値から 20 dB 低い値（P0–20）を設定します。
14. シグナルアナライザでレベルを測定します。この測定値を S2 とします。
15. 信号発生器の出力レベルに，手順 9 で設定した値から 30 dB 低い値（P0–30）を設定します。
16. シグナルアナライザでレベルを測定します。この測定値を S3 とします。
17. 信号発生器の出力レベルに，手順 9 で設定した値から 40 dB 低い値（P0–40）を設定します。
18. シグナルアナライザでレベルを測定します。この測定値を S4 とします。
19. 手順 5 で測定した，出力レベルが–25 dBm となる値（P25）を信号発生器に設定して，手順 9～18 を繰り返します。
20. 信号発生器の周波数を「表7.3.5-1　周波数リスト」の各周波数に設定し，手順 8～19 を繰り返します。

表7.3.5-1　周波数リスト

番号	MU887002A 受信周波数 (MHz)	信号発生器 設定周波数 (MHz)
1	1318*1,*2	1317.9*1,*2
2	2236*1,*2	2235.9*1,*2
3	3154*1,*2	3153.9*1,*2
4	4072*1,*2	4071.9*1,*2
5	4990*1,*2	4989.9*1,*2
6	6000*1	5999.9*1
7	5900*3	5900.1*3
8	6818*3	6817.9*3
9	7300*3	7299.9*3

＊1: MU887002A-007 非搭載時 TRX1/2, Port 1～12
　　または MU887002A-007 搭載時 TRX1/2, Port 1～4
＊2: MU887002A-007 搭載時 TRX1/2, Port 5～12
　　設定周波数＜5900 MHz
＊3: MU887002A-007 搭載時 TRX1/2, Port 5～12
　　設定周波数≧5900 MHz

(4) a 試験手順（MU887002A-007 非搭載時 Port 1～12,
MU887002A-007 搭載時 Port 1～4,
または MU887002A-007 搭載時 Port 5～12 設定周波数＜5900 MHz）

SA リニアリティは，TRX 1/2，Port 1～Port 12 を試験します。
TRX 1 Port 1 以外を試験する場合は，
TRX 1 Port 1 の場合の "TRX 1 Port 1" を読み替えてください。

TRX 1 Port 1 の場合

1. TRX 1 Port 1 を試験する場合，図7.3.5-3のとおり機器を接続します。
2. MU887002A の設定を初期化します。

```
*RST
```

3. 入力ポート，出力ポートを TRX 1 Port 1 に設定します。

```
:ROUTe:PORT:CONNect:DIRection PORT1,PORT1
```

4. VSG の動作モードを Normal に設定します。

```
:SOURce:GPRF:GENerator:MODE NORMAL
```

5. VSG の変調を CW に設定します。

```
:SOURce:GPRF:GENerator:BBMode CW
```

6. VSG の出力レベルを–120 dBm に設定します。

```
:SOURce:GPRF:GENerator:RFSettings:LEVel -120
```

7. VSG の信号出力をオフにします。

```
:SOURce:GPRF:GENerator:STATe OFF
```

8. MU887002A の測定項目をスペクトラムモニタにします。

```
:INSTrument[:SELect] CELLULAR
:CONFigure:CELLular:MEASurement:STANdard COMMON
:CONFigure:CELLular:MEASurement:SELect SPMON
```

9. スペクトラムモニタを次のとおり設定します。

入力周波数:	「表7.3.5-1　周波数リスト」の番号 1 の周波数
スパン:	50 MHz
パワー測定帯域幅:	300 kHz
RBW:	100 kHz
検波方式:	rms

```
:CONFigure:CELLular:MEASurement:RFSettings:FREQuency
1318MHZ
:CONFigure:CELLular:COMMon:SPMonitor:SPAN 50MHZ
:CONFigure:CELLular:COMMon:SPMonitor:PMBW 300KHZ
:CONFigure:CELLular:COMMon:SPMonitor:RBW 100KHZ
:CONFigure:CELLular:COMMon:SPMonitor:DETect RMS
```

10. 信号発生器の出力周波数を 1317.9 MHz に設定します（「表7.3.5-1 周波数リスト」の番号 1)。

11. 信号発生器の出力レベルに「SA リニアリティ測定の準備」で求めた P0 を設定します。
12. MU887002A の入力レベルを 0 dBm に設定します。

    ```
    :CONFigure:CELLular:MEASurement:RFSettings:LEVel 0
    ```

13. MU887002A でレベルを測定します。この測定値を *A* dBm とします。

    ```
    :INITiate:CELLular:MEASurement:SINGle
    *WAI
    :FETCh:CELLular:COMMon:SPMonitor:POWer
    ```

14. 信号発生器の出力レベルを，手順 11 で設定した値から 10 dB 低い値に設定します。
15. MU887002A でレベルを測定します。この測定値を *B1* dBm とします。
16. 次の式でリニアリティを計算します。

    以下 S, S1, S2, S3, S4 は「SA リニアリティ測定の準備」で取得した値です。

    リニアリティ = (B1 – A) – (S1 – S)
17. 信号発生器の出力レベルを，手順 11 で設定した値から 20 dB 低い値に設定します。
18. MU887002A でレベルを測定します。この測定値を *B2* dBm とします。
19. 次の式でリニアリティを計算します。

    リニアリティ = (B2 – A) – (S2 – S)
20. 信号発生器の出力レベルを，手順 11 で設定した値から 30 dB 低い値に設定します。
21. MU887002A でレベルを測定します。この測定値を *B3* dBm とします。
22. 次の式でリニアリティを計算します。

    リニアリティ = (B3 – A) – (S3 – S)
23. 信号発生器の出力レベルを，手順 11 で設定した値から 40 dB 低い値に設定します。
24. MU887002A でレベルを測定します。この測定値を *B4* dBm とします。
25. 次の式でリニアリティを計算します。

    リニアリティ = (B4 – A) – (S4 – S)
26. MU887002A に入力するレベルが–25 dBm になるように，信号発生器の出力レベルを設定して，手順 12～25 を繰り返します。
27. 「表7.3.5-1　周波数リスト」の各周波数について，手順 9～26 を繰り返します。

(4) b 試験手順（MU887002A-007 搭載時 Port 5～12
設定周波数≧5900 MHz）

SA リニアリティは，TRX 1/2，Port 5～Port 12 を試験します。
TRX 1 Port 5 以外を試験する場合は，
TRX 1 Port 5 の場合の "TRX 1 Port 5" を読み替えてください。

TRX 1 Port 5 の場合

1. TRX 1 Port 5 を試験する場合，図7.3.5-3のとおり機器を接続します。
2. MU887002A の設定を初期化します。
   ```
   *RST
   ```
3. 入力ポートを TRX 1 Port 5 に，出力ポートを TRX 1 Port 6 に設定します（入力ポートを TRX 1 Port 6 に設定するときは出力ポートを TRX 1 Port 5 に設定します）。
   ```
   :ROUTe:PORT:CONNect:DIRection PORT5,PORT6
   ```
4. VSG の動作モードを Normal に設定します。
   ```
   :SOURce:GPRF:GENerator:MODE NORMAL
   ```
5. VSG の変調を CW に設定します。
   ```
   :SOURce:GPRF:GENerator:BBMode CW
   ```
6. VSG の出力レベルを–120 dBm に設定します。
   ```
   :SOURce:GPRF:GENerator:RFSettings:LEVel -120
   ```
7. VSG の信号出力をオフにします。
   ```
   :SOURce:GPRF:GENerator:STATe OFF
   ```
8. MU887002A の測定項目をスペクトラムモニタにします。
   ```
   :INSTrument[:SELect] CELLULAR
   :CONFigure:CELLular:MEASurement:STANdard COMMON
   :CONFigure:CELLular:MEASurement:SELect SPMON
   ```
9. スペクトラムモニタを次のとおり設定します。

入力周波数:	「表7.3.5-1　周波数リスト」の番号 7 の周波数
スパン:	50 MHz
パワー測定帯域幅:	300 kHz
RBW:	100 kHz
検波方式:	rms

   ```
   :CONFigure:CELLular:MEASurement:RFSettings:FREQuency
   5900MHZ
   :CONFigure:CELLular:COMMon:SPMonitor:SPAN 50MHZ
   :CONFigure:CELLular:COMMon:SPMonitor:PMBW 300KHZ
   :CONFigure:CELLular:COMMon:SPMonitor:RBW 100KHZ
   :CONFigure:CELLular:COMMon:SPMonitor:DETect RMS
   ```
10. 信号発生器の出力周波数を 5900.1 MHz に設定します（「表7.3.5-1 周波数リスト」の番号 7）。

11. 信号発生器の出力レベルに「SA リニアリティ測定の準備」で求めた P0 を設定します。
12. MU887002A の入力レベルを 0 dBm に設定します。

    ```
    :CONFigure:CELLular:MEASurement:RFSettings:LEVel 0
    ```

13. MU887002A でレベルを測定します。この測定値を *A* dBm とします。

    ```
    :INITiate:CELLular:MEASurement:SINGle
    *WAI
    :FETCh:CELLular:COMMon:SPMonitor:POWer
    ```

14. 信号発生器の出力レベルを，手順 11 で設定した値から 10 dB 低い値に設定します。
15. MU887002A でレベルを測定します。この測定値を *B1* dBm とします。
16. 次の式でリニアリティを計算します。

    以下 S, S1, S2, S3, S4 は「SA リニアリティ測定の準備」で取得した値です。

    リニアリティ = (B1 – A) – (S1 – S)
17. 信号発生器の出力レベルを，手順 11 で設定した値から 20 dB 低い値に設定します。
18. MU887002A でレベルを測定します。この測定値を *B2* dBm とします。
19. 次の式でリニアリティを計算します。

    リニアリティ = (B2 – A) – (S2 – S)
20. 信号発生器の出力レベルを，手順 11 で設定した値から 30 dB 低い値に設定します。
21. MU887002A でレベルを測定します。この測定値を *B3* dBm とします。
22. 次の式でリニアリティを計算します。

    リニアリティ = (B3 – A) – (S3 – S)
23. 信号発生器の出力レベルを，手順 11 で設定した値から 40 dB 低い値に設定します。
24. MU887002A でレベルを測定します。この測定値を *B4* dBm とします。
25. 次の式でリニアリティを計算します。

    リニアリティ = (B4 – A) – (S4 – S)
26. MU887002A に入力するレベルが–25 dBm になるように，信号発生器の出力レベルを設定して，手順 12～25 を繰り返します。
27. 「表7.3.5-1　周波数リスト」の各周波数について，手順 9～26 を繰り返します。

7.3.6　パスロス測定確度 (MX887092A)

ここでは，MX887092A パスロス測定機能の測定確度に関する試験をします。

(1)　試験対象規格

TRX 1/2，Port 1～Port 12，20～30°C，移動平均の区間数 11 にて
経路長 1.5 m，比誘電率 2.03 の同軸ケーブルにて

TRX 1/2，Port 11 が規格を満足するとき，同じ TRX の他の Port の規格は設計保証されます。

周波数 f (MHz)	規格
400≦f≦2700	±0.5 dB
3300≦f≦6000	±0.9 dB

(2)　試験用測定器

- ネットワークアナライザ:　S820E
- 校正キット
- パスロス測定用同軸ケーブル:　1 個
- パスロス測定用 OPEN/SHORT:　22N50　1 個
- 中継アダプタ:　34NFNF50　1 個

(3)　セットアップ

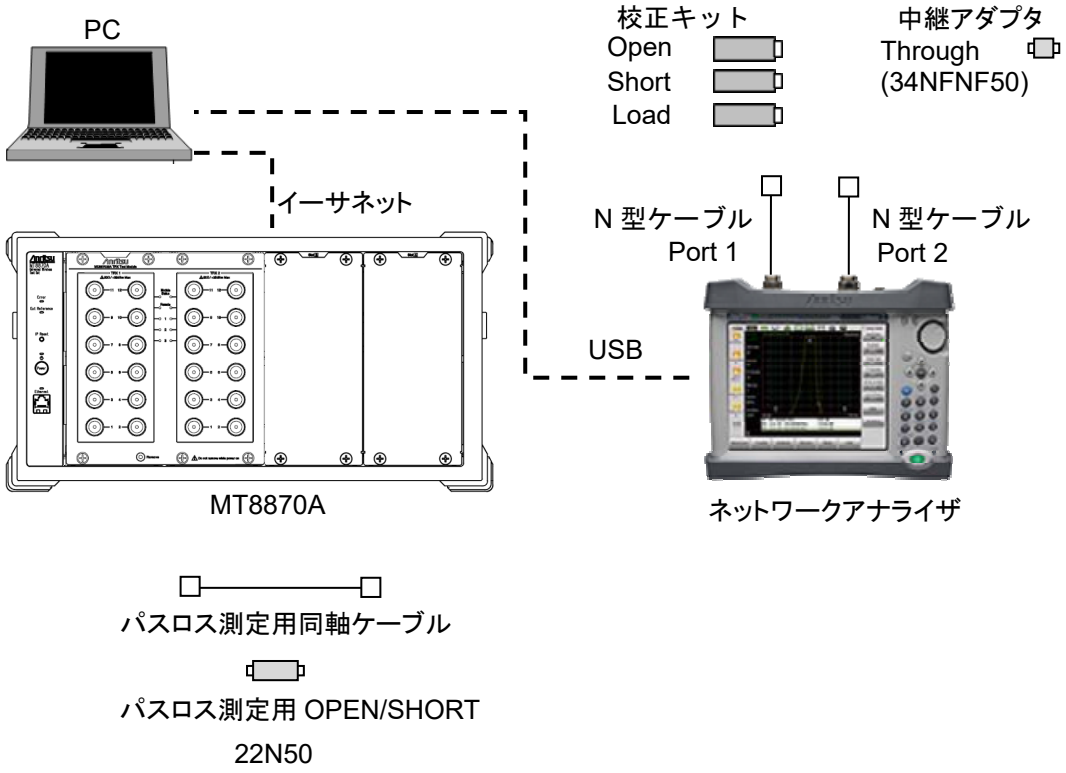


図7.3.6-1　パスロス測定確度の試験系

(4)　試験手順

パスロス測定確度は TRX 1/2，Port 11 を試験します。
TRX 1/2，Port 11 が規格を満足するとき，同じ TRX の他の Port の規格は設計保証されます。

1. ネットワークアナライザに N 型ケーブルを接続します。
2. ネットワークアナライザを，以下のように設定します。
   - Start Frequency:　100 MHz
   - Stop Frequency:　8000 MHz
   - Smoothing:　5%
3. N 型ケーブルの先端にて，Open，Short，Load，Through を使用して Full 2 Port の校正をします。
4. パスロス測定用同軸ケーブルの S_{21} を測定します。
5. パスロス測定用同軸ケーブルを TRX 1 の Port 11 に接続します。
6. パスロス計算に使用する移動平均の区間数を 11 とし，付録 E の手順に従って，「表7.3.6-1　周波数リスト」の周波数でパスロスを測定します。
7. 手順 6 で測定したパスロス値と手順 4 で測定した S_{21} からパスロス測定確度を計算します。

   パスロス測定確度 = (手順 6 のパスロス値) + (手順 4 の S_{21})
8. パスロス測定用同軸ケーブルを TRX 2 の Port 11 に接続して，手順 6～7 を繰り返します。

表7.3.6-1　周波数リスト

番号	周波数 (MHz)
1	400
2	1852.5
3	2700
4	3300
5	4920
6	6000

7.4 サービスについて

万一，破損または仕様どおりに動作しない場合は，本書（紙版説明書では巻末，電子版説明書では別ファイル）に記載の「本製品についてのお問い合わせ窓口」へすみやかにご連絡ください。

修理の依頼をするときは，次の内容をご連絡ください。

(a)　機器名と背面パネルに記入されている機械番号

(b)　故障状況

(c)　故障内容について確認や，修理完了時に連絡をとることができる担当者名

(d)　ソフトウェアバージョン

第8章　ユーティリティツール

この章では，MX887900A ユーティリティツールの使用方法について説明します。

8.1　ユーティリティツールの概要 8-2
8.1.1 機能 8-2
8.1.2 動作環境 8-7
8.2　インストール 8-8
8.2.1 .NET Framework 8-8
8.2.2 NI-VISA 8-10
8.2.3 ユーティリティツール 8-12
8.3　操作方法 8-15
8.3.1 起動と終了 8-15
8.3.2 画面の名称 8-16
8.3.3 機器の探索 8-18
8.3.4 機器の情報表示 8-19
8.3.5 ネットワーク設定 8-22
8.3.6 モジュール設定 8-24
8.3.7 ログの操作 8-26
8.3.8 アップグレード 8-27
8.3.9 ファイル転送 8-30
8.3.10 一括アップグレード 8-32
8.3.11 ライセンス登録 8-34
8.3.12 波形ファイル転送 8-35
8.3.13 直接ファイル転送 8-37

8.1　ユーティリティツールの概要

8.1.1　機能

ユーティリティツールの機能は次のとおりです。

MT8870A 探索

イーサネットで接続している MT8870A を探索して，結果を表示します。

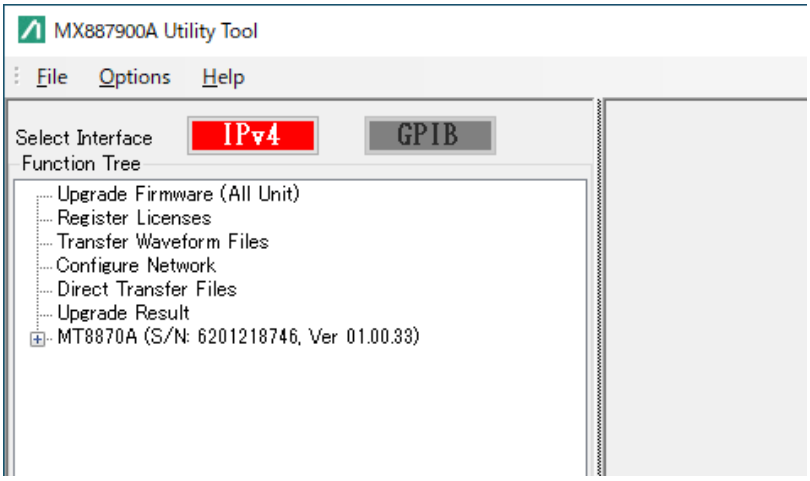


図8.1.1-1　探索結果の表示

ネットワーク設定

モジュールの IP アドレス, サブネットマスクなどのネットワーク設定をします。
Module 1/2/3/4 は, MU887002A 2 台に対応できます。

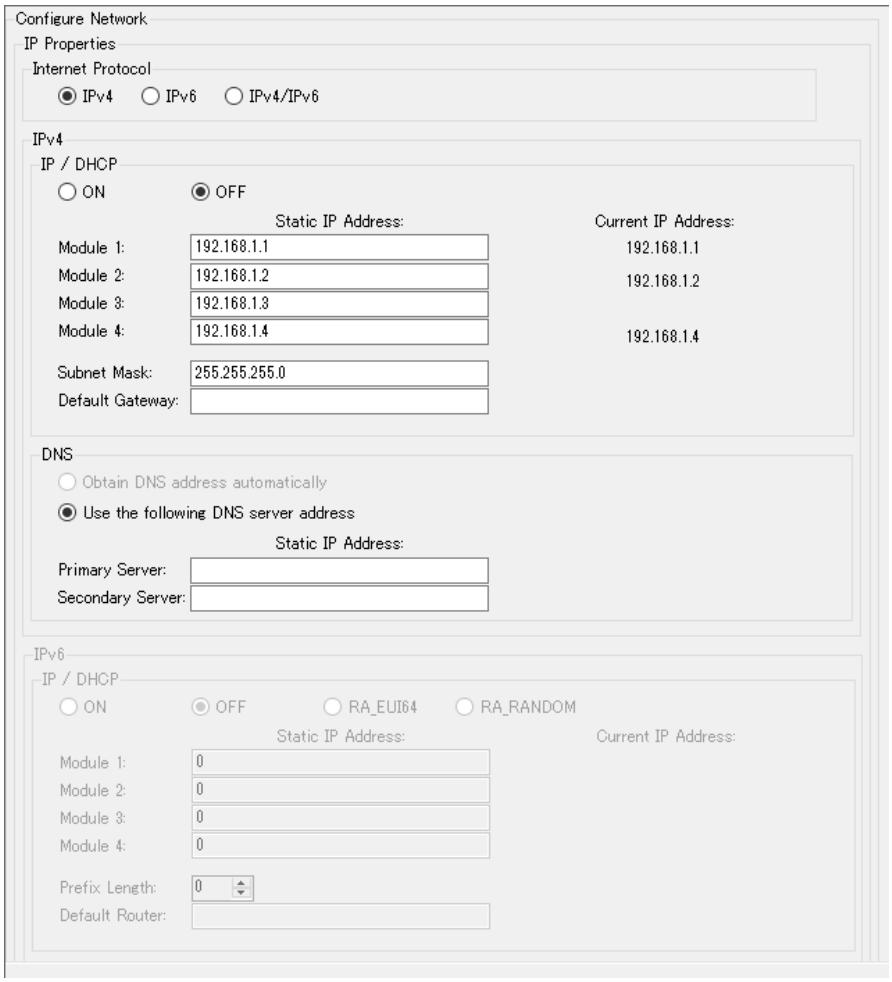


図8.1.1-2　ネットワーク設定画面

メンテナンス情報表示

MT8870A，およびモジュールの自己診断結果，稼働時間，電源投入回数，校正結果，通信設定などの情報を確認できます。



図8.1.1-3　メンテナンス 情報画面 (自己診断結果，稼働時間)



図8.1.1-4　メンテナンス 情報画面 (ハードウェア情報)

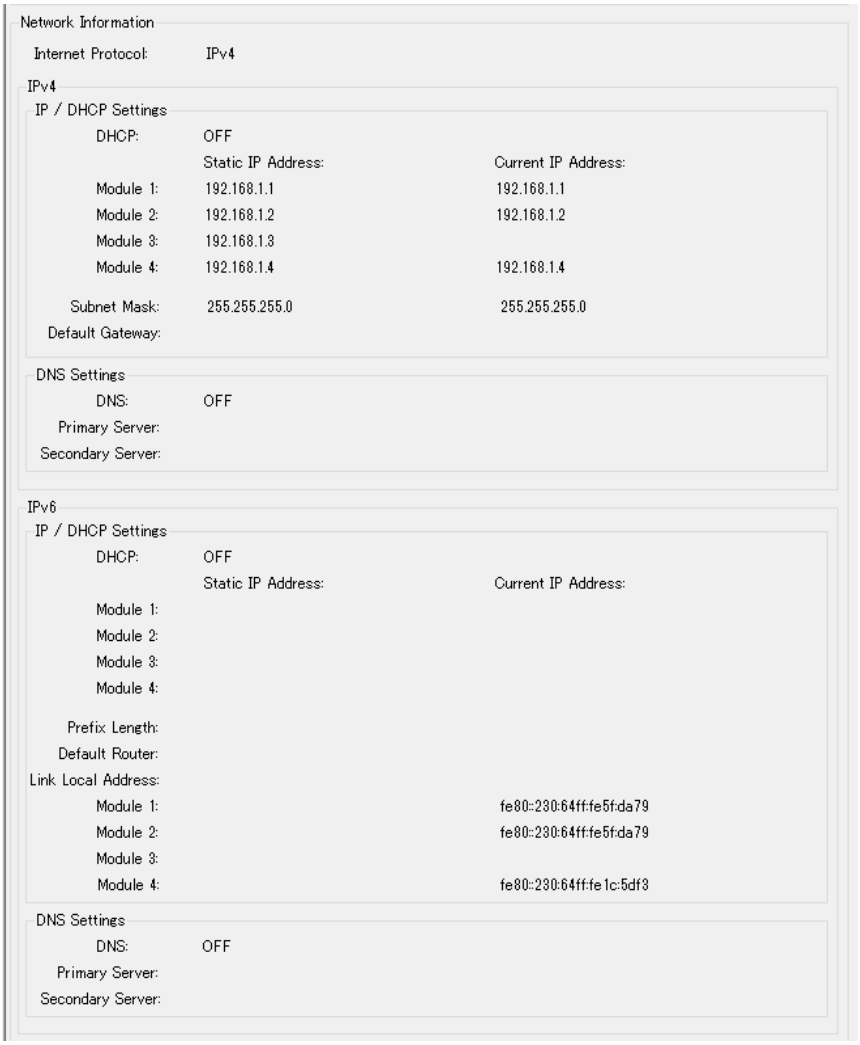


図8.1.1-5　メンテナンス 情報画面 (通信設定 1)

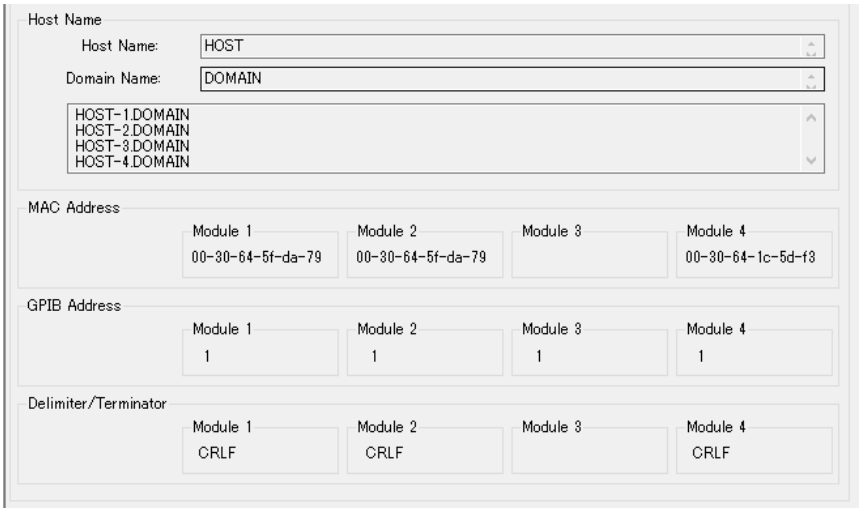


図8.1.1-6　メンテナンス 情報画面 (通信設定 2)

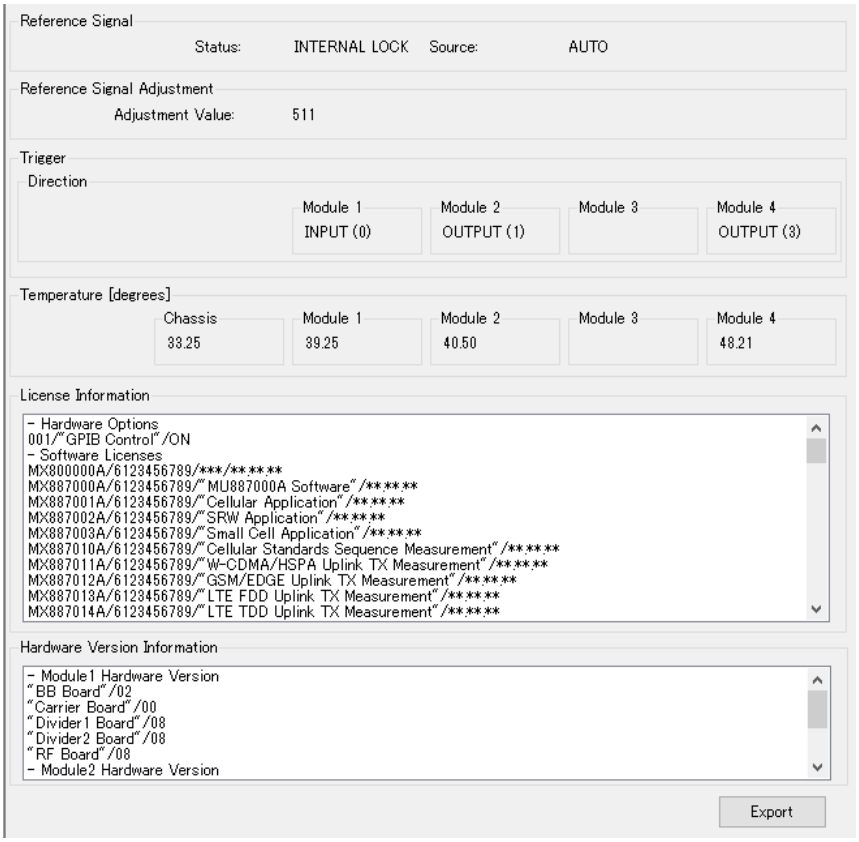


図8.1.1-7　メンテナンス 情報画面
(トリガ設定，ライセンス情報，ハードウェアバージョン情報)

ファイル転送

FTP を使用して，コンピュータとモジュール間でファイルを転送します。

アップグレード

MT8870A，およびモジュールの内蔵プログラムを更新します。

ライセンス登録

アプリケーションソフトウェアのライセンス情報を MT8870A に登録します。

8.1.2 動作環境

ユーティリティツールは Ver 2.0.0 以降のバージョンが必要です。
ユーティリティツールの動作環境は次のとおりです。

表8.1.2-1　ユーティリティツールの動作環境

項目	仕様
OS	Windows 7 Enterprise Service Pack 1 日本語/英語 Windows 10　日本語/英語
ディスプレイ	1024 × 768 以上
メモリ	1 GB 以上
ハードディスク 空き容量	200 MB 以上
VISA	NI-VISA*
インタフェース	イーサネット

*:　.NET Framework と NI-VISA のバージョンの対応は表8.1.2-2を参照してください。

表8.1.2-2　.NET Framework と NI-VISA の対応表

NET Framework	NI-VISA 5.0.3～5.2	NI-VISA 5.3 以降
.NET Framework 4.0	○	○
.NET Framework 4.5	×	○

○:　対応　×:　非対応

表8.1.2-3　NI-VISA のバージョン

ユーティリティツールのバージョン	NI-VISA のバージョン
Ver 2.0.0 以降	Version 5.03 以降, Version 5.4.1 以前, Version 14.0 以降

ユーティリティツールのバージョンとモジュール内蔵プログラムのバージョンの関係を次の表に示します。モジュールのバージョンは, 図8.3.2-1のツリービューに表示されます。

表8.1.2-4　対応するバージョン

ユーティリティツールのバージョン	モジュール内蔵プログラムのバージョン
Ver 2.0.0 以降	Ver 3.0.0 以降

8.2　インストール

8.2.1　.NET Framework

Windows 7 を使用する場合は，NET Framework 4.5 をインストールします。

8.2.1.1　.NET Framework 4.5

インストール手順

1. Microsoft 社のホームページから，.NET Framework 4.5 をダウンロードします。

   日本語：
   https://www.microsoft.com/ja-jp/download/details.aspx?id=30653

   英語：
   https://www.microsoft.com/en-us/download/details.aspx?id=30653

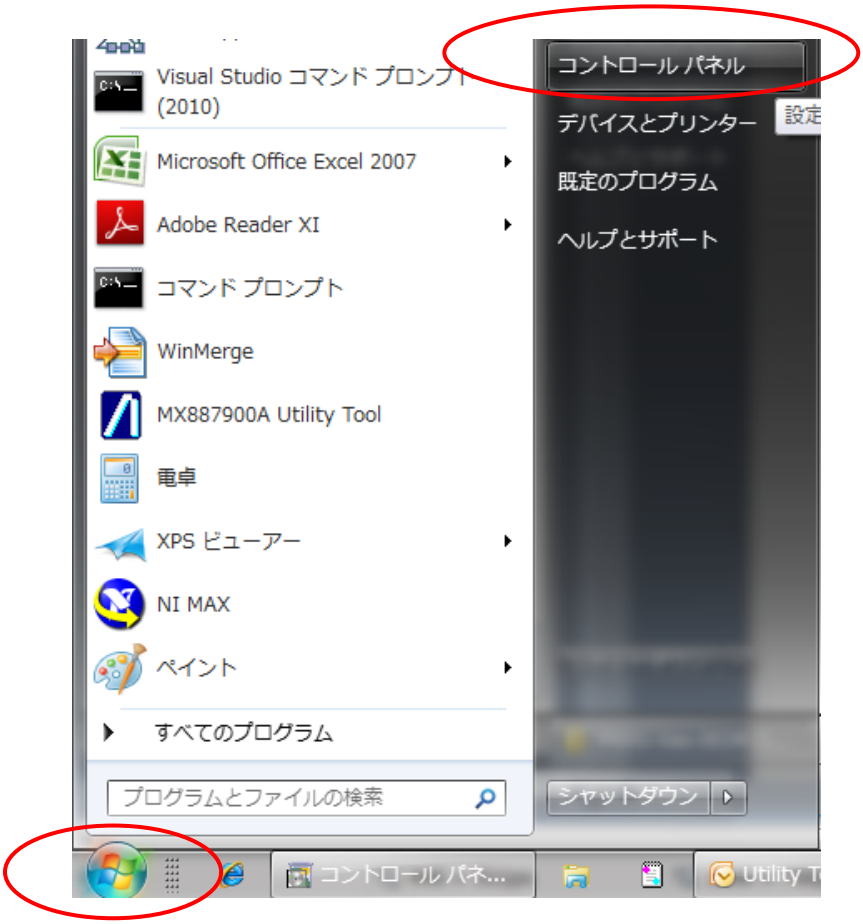
2. ダウンロードしたインストール実行ファイルをダブルクリックします。画面の指示に従ってインストールを開始します。

注：

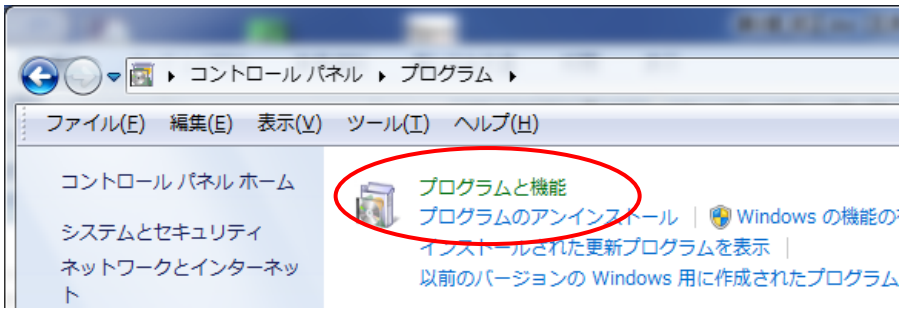
.NET Framework 4.5 をインストールすると，.NET Framework 4.0 は削除されます。

.NET Framework 4.5 のインストールの確認

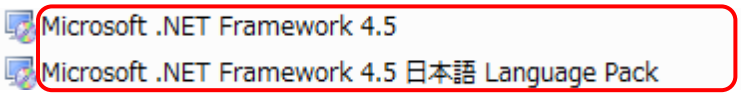
1. [スタート] → [コントロールパネル] の順でクリックします。



2. [プログラムのアンインストール], または [プログラムと機能] をクリックします。



3. Microsoft .NET Framework 4.5 が一覧に表示されていることを確認します。



8.2.2 NI-VISA

ユーティリティツールを使用する場合，制御用 PC などに VISA ドライバをインストールする必要があります。VISAドライバとしてNational Instruments™社（以下NI™社）の NI-VISA™を推奨します。
NI-VISA™の利用には適切な NI-VISA™ライセンスが必要です。
その利用および許諾範囲に関しては NI™社の規約を順守してください。

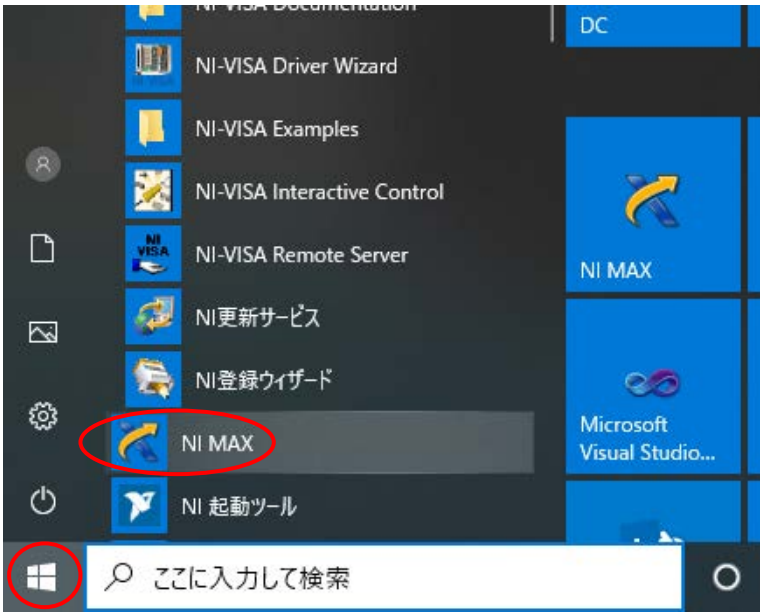
- NI-VISA™ドライバ自体は無料でウェブからダウンロードできますが，要件を満たしていない場合のみ，法的な理由で実装ライセンスが必要になります。(要件の詳細については NI™社の web ページでご確認ください。)
- NI™社のハードウェアやソフトウェアが使用されていない場合には NI-VISA™実装ライセンスを購入する必要があります。

インストール手順

1. National Instruments 社のホームページから，NI-VISA をダウンロードします。
2. ダウンロードしたファイルをダブルクリックします。画面の指示に従ってインストールを開始します。
3. インストールが終わったらコンピュータを再起動します。

NI-VISA のソフトウェアバージョンの確認方法

1. [スタート]→インデックス N の [NI MAX] の順にクリックします。



2. 「マイシステム」→「ソフトウェア」の順でクリックします。ソフトウェアの下にツリー構造で各ソフトウェアのバージョンが表示されます。



3. NI-VISA のバージョンが 5.0.3 以降であることを確認します。

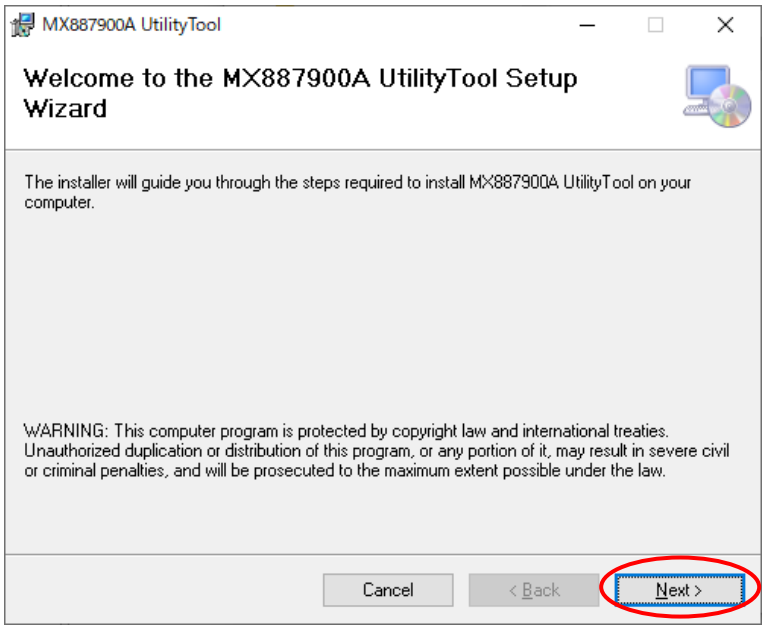
8.2.3 ユーティリティツール

インストール手順

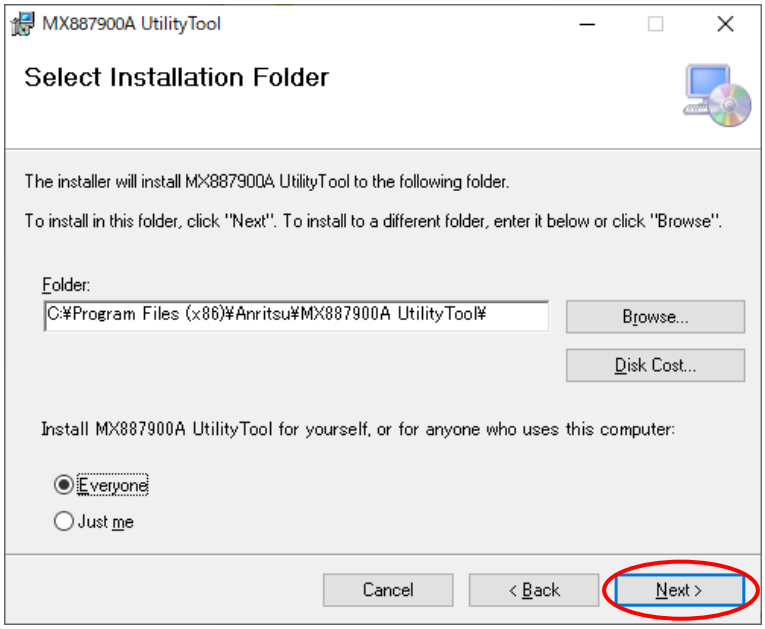
1. MT8870A に添付されている記憶媒体のセットアッププログラム（setup.exe）をダブルクリックします。

   セットアッププログラムは，製品に添付されている記憶媒体の \Installer\MT8870A_Utility Tool にあります。

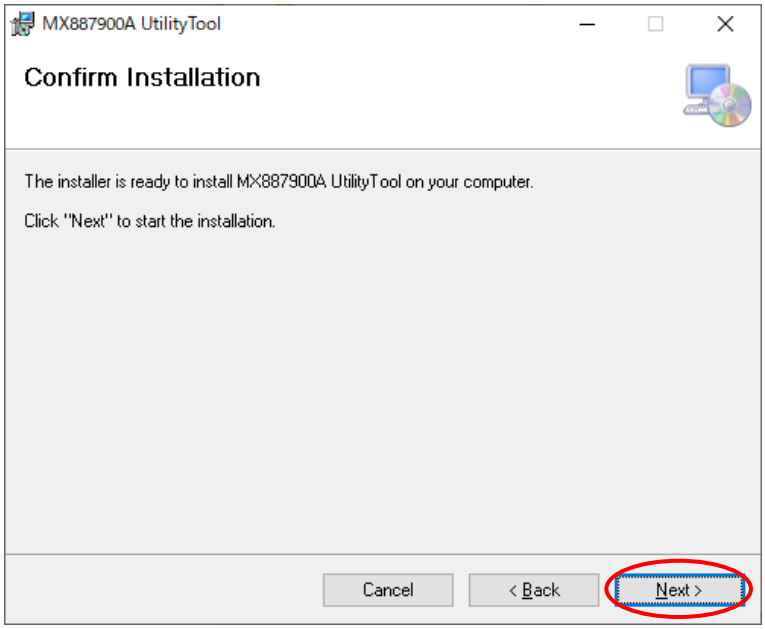
2. セットアップ画面の［Next］をクリックします。



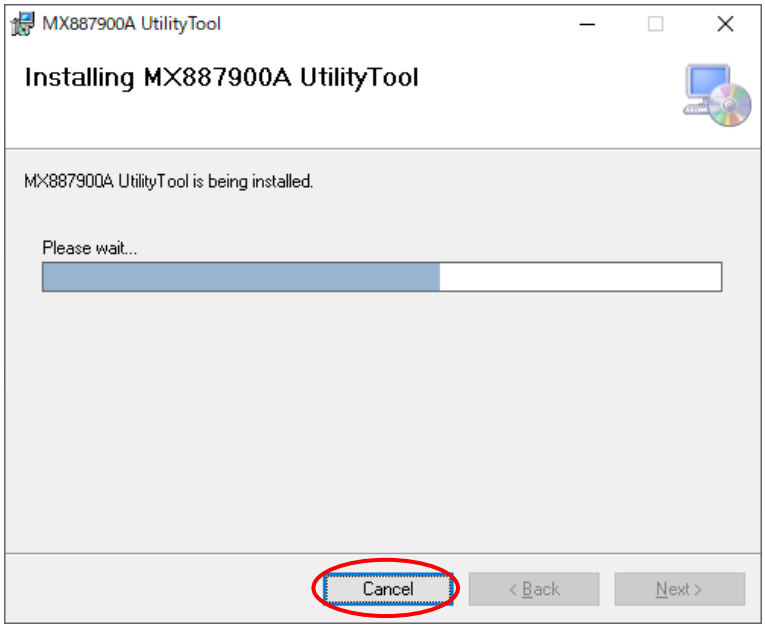
3. インストール先のフォルダを指定して，[Next] をクリックします。



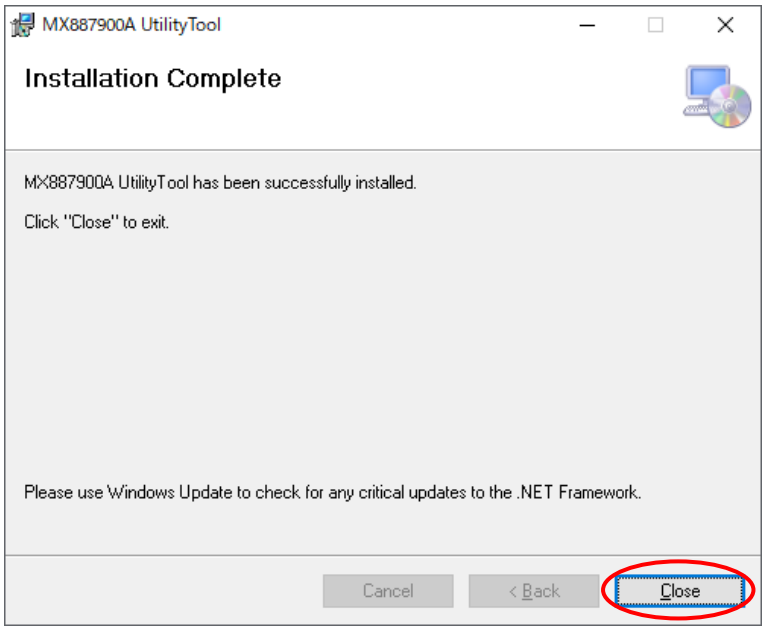
4. 次の画面が表示されたら，[Next] をクリックします。



5. インストールが始まるとプログレスバー付きのダイアログボックスが表示されます。インストールを中止する場合は，[Cancel] をクリックします。



6. インストールが完了したら，[Close] をクリックして，ダイアログボックスを閉じます。



ユーティリティツールのアンインストール

1. コントロールパネルから [プログラムと機能] を起動します。
2. プログラムの一覧から MX887900A Utility Tool を選択し，ダブルクリックするとアンインストールを開始します。
   画面の指示に従ってください。

8.3 操作方法

この節では，ユーティリティツール画面の操作箇所の名称に［ ］を付けて記載します。

たとえば図8.3.1-1の File の操作を説明する場合，［File］と記載します。

8.3.1 起動と終了

起動

1. スタートメニューのすべてのプログラムから，Anritsu を開き MX887900A Utility Tool を選択します。
2. MT8870A と通信するインタフェースのボタンをクリックします。モジュール検索のメッセージが表示されます。
3. モジュールを検出できた場合，ツリービューの表示が更新されます。

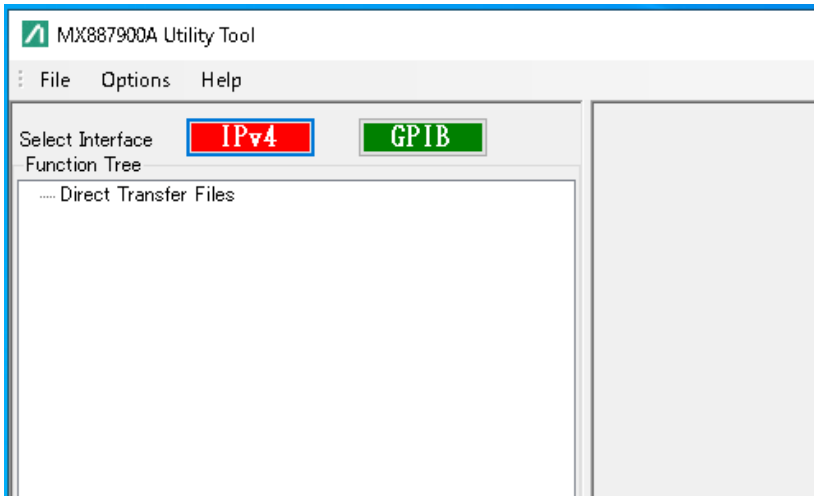


図8.3.1-1 起動時の画面

終了

1. メニューの［File］–［Exit］をクリックします。
2. 終了を確認するメッセージが表示されます。[はい］をクリックします。

8.3.2 画面の名称

画面の名称を次の図に示します。

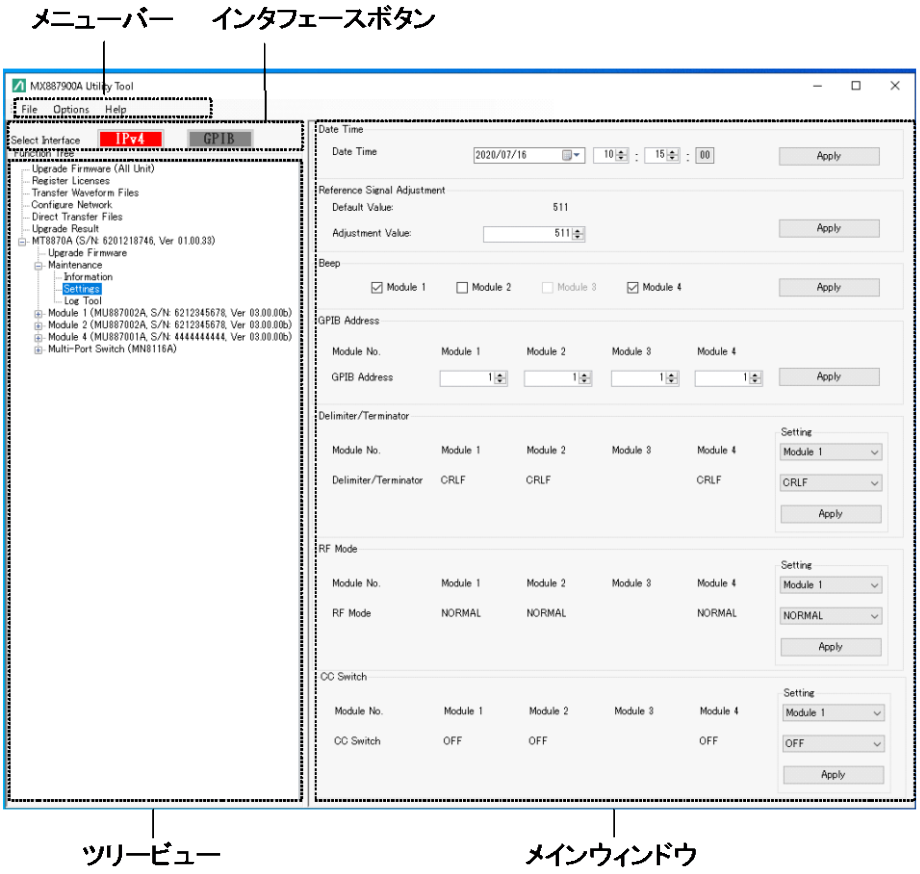


図8.3.2-1　画面の名称

表8.3.2-1 画面の説明

名称	説明	
メニュー	File Exit ユーティリティツールを終了します。	
	Options FIRMUPDATE option DIFF 内蔵プログラムと読み込むプログラムのバージョンが違う場合だけアップグレードします。 FORCE 内蔵プログラムと読み込むプログラムのバージョンが同じでも，強制的にアップグレードします。	
	Help バージョン情報を表示します。	
インタフェースボタン	IPv4	MT8870A と通信するインタフェースを選択します。 MU887002A の場合，GPIB は選択できません。
	GPIB	
ツリービュー	探索した MT8870A が表示されます。 MT8870A のスロットにモジュールが装着されている場合，Module 1～Module 4 が表示されます。 Module 1～Module 4 の番号は，モジュールが装着されている MT8870A のスロット番号です。Module 1～Module 4 には，形名・製造番号・内蔵プログラムのバージョンが表示されます。 ツリービューの項目をクリックすると，メインウィンドウの表示が切り替わります。	
メインウィンドウ	MT8870A またはモジュールの情報，操作ボタン，実行状況などが表示されます。	

8.3.3 機器の探索

MT8870A を探索します。

1. インタフェースボタンの [IPv4] をクリックして，インタフェースを選択します。

   [IPv4]をクリックした場合は，下記の画面で探索方法を指定します。

   - “Manually enter instrument IP address”のチェック無しの場合

     選択されたネットワーク上に存在するすべての MT8870A を探索します。

   - “Manually enter instrument IP address”のチェック有りの場合

     入力された IP アドレスのモジュールが装着された MT8870A を探索します。

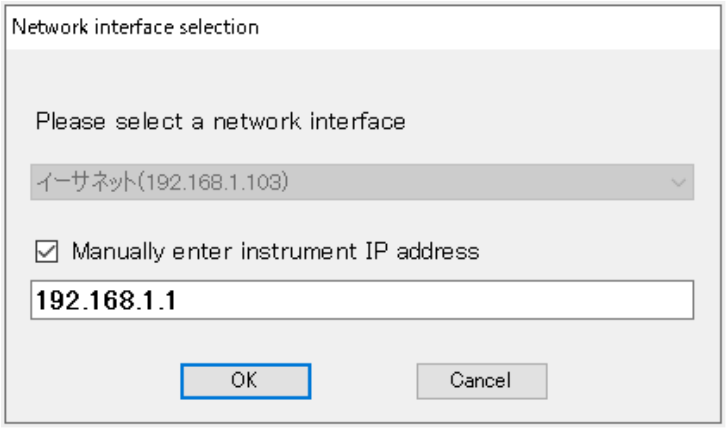


図8.3.3-1　ネットワークインタフェース選択画面

2. MT8870A が存在する場合は，ツリービューに形名と製造番号が表示されます。

   インタフェースが IPv4 の場合，ツリービューに以下が表示されます。

   Update Firmware (All Unit)
   Register Licenses
   Transfer Waveform Files
   Configure Network
   Direct Transfer files
   Update Result

注:

- 機器の探索は，数分かかることがあります。
- MT8870A にモジュールが装着されていないと，探索してもツリービューに結果が表示されません。

8.3.4 機器の情報表示

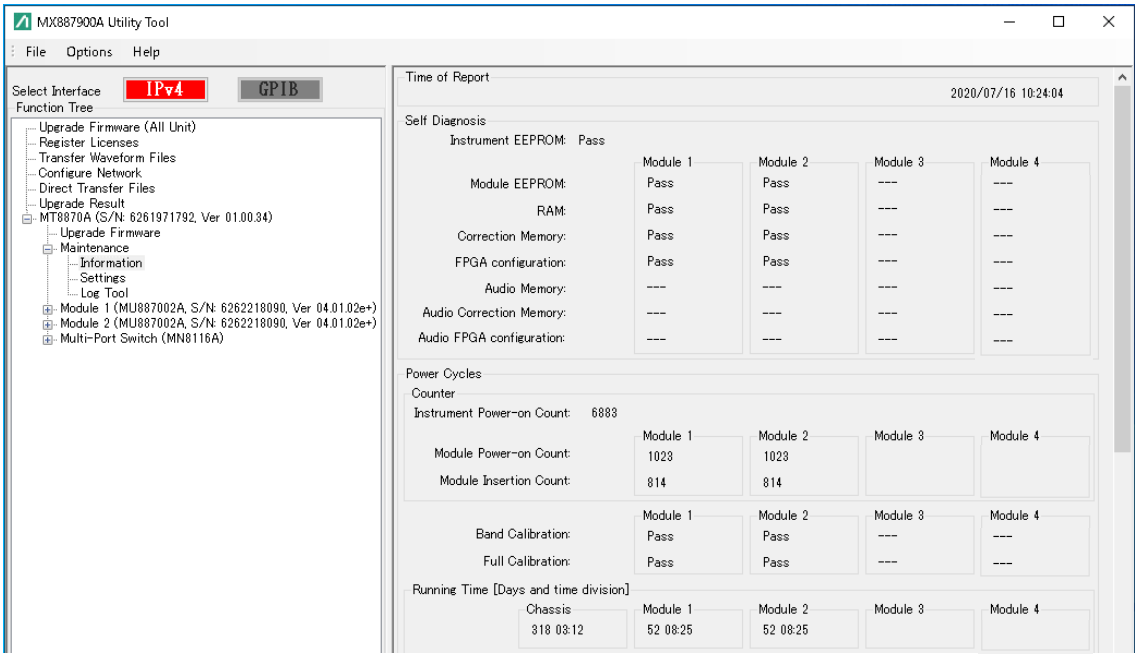
MT8870A, およびモジュールの情報を表示します。

1. ツリービューの MT8870A の [+] をクリックします。
2. [Maintenance] の [+] をクリックします。
3. [Information] をクリックすると, MT8870A, およびモジュールの情報が表示されます。

表示情報の保存

1. 画面の一番下にある [Export] をクリックすると, ファイル保存画面 (Export Log File) が表示されます。
2. フォルダ, ファイル名を入力して, [OK] をクリックすると, 表示されている情報がファイルに保存されます。

Maintenance Information画面の先頭



Maintenance Information画面の最下行

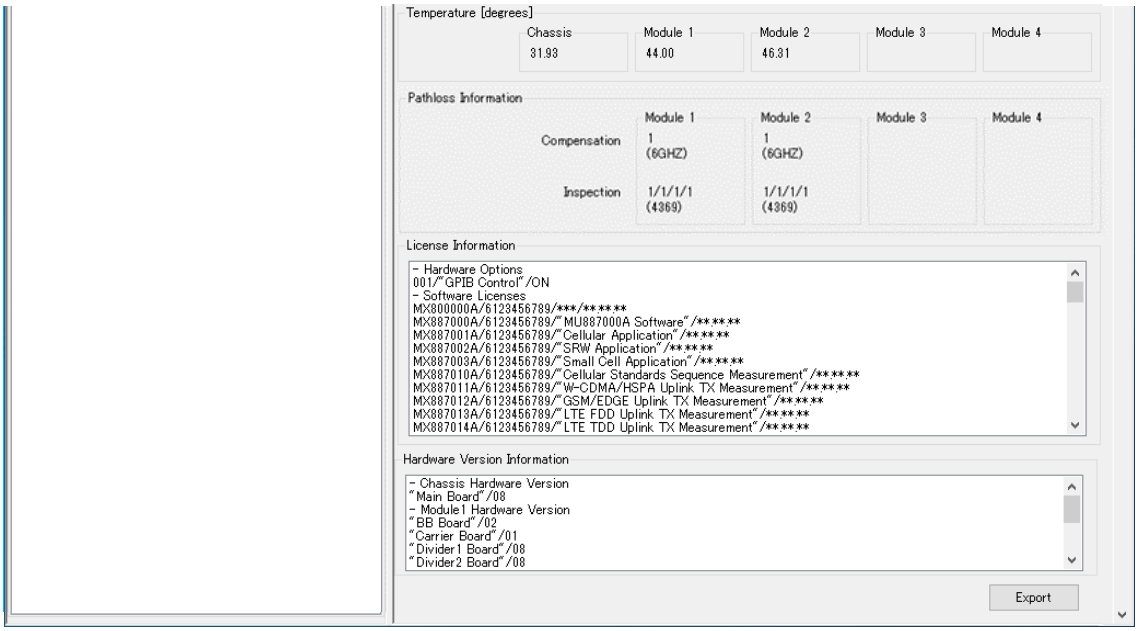


図8.3.4-1　Maintenance Information 画面

表8.3.4-1 メンテナンス情報の項目

表示項目	説明
Time of Report	メンテナンス情報を取得した日時（日, 時, 分)
Self-Diagnosis	自己診断結果
Power Cycles Counter	Instrument Power-on Count: MT8870A の電源投入回数 Module Power-on Count: モジュールの電源投入回数 Module Insertion Count: モジュールの挿抜回数
Band Calibration	バンド校正の実行結果
Full Calibration	全バンド校正の実行結果
Running Time	稼働時間（日, 時, 分)
Information	Chassis: 形名, 製造番号, バージョン Module 1～Module 4: スロット 1～スロット 4 のモジュールの形名, 製造番号, バージョン, アプリケーションソフトウェア, RF Mode (ユーティリティツールの Ver. 1.3.1 以降)
Date Time	情報取得時の MT8870A の日時（日, 時, 分)
Network Information	
Internet Protocol	IP のバージョン
IPv4	DHCP, DNS の設定
IPv6	DHCP, DNS の設定
Host Name	ホスト名, ドメイン名
MAC アドレス	スロット 1～スロット 4 のモジュールの MAC アドレス
GPIB アドレス	スロット 1～スロット 4 のモジュールの GPIB アドレス
Delimiter/Terminator	スロット 1～スロット 4 のモジュールの終端文字
Reference Signal	MT8870A の基準信号設定
Reference Signal Adjustment	MT8870A 基準信号のレベル調整値
Trigger Settings	MT8870A のトリガコネクタの入出力
Temperature [degrees]	MT8870A, およびモジュールの温度（°C)
Pathloss Information	スロット 1～スロット 4 のモジュールごとの 補正（対応周波数範囲) 検査
License Information	ライセンス情報
Hardware Version Information	ハードウェアバージョン情報

8.3.5 ネットワーク設定

ネットワークの通信設定をします。

設定項目の編集

1. ツリービューの [Configure Network] をクリックすると，ネットワーク設定画面(Configure Network)が表示されます。
2. All Unit に表示されている形名をクリックして，設定するモジュールを選択します。
3. メインウィンドウの項目を設定します。[Reload] をクリックすると，編集中のモジュールの設定値を，変更前の値に戻します。[Reload] は，メインウィンドウを下にスクロールすると一番下にあります。
4. 複数の MT8870A のネットワーク情報を設定する場合は，手順 1 から 3 を繰り返します。
5. [Apply] をクリックすると，すべてのモジュールにネットワーク情報が設定され，ネットワーク設定画面が閉じます。
   [Cancel] をクリックすると，すべての項目が変更前の設定に戻ります。
6. モジュールにネットワーク情報が設定されると，MT8870A が再起動します。MT8870A が再起動した後，インタフェースボタンをクリックしてツリービューの表示を更新します。

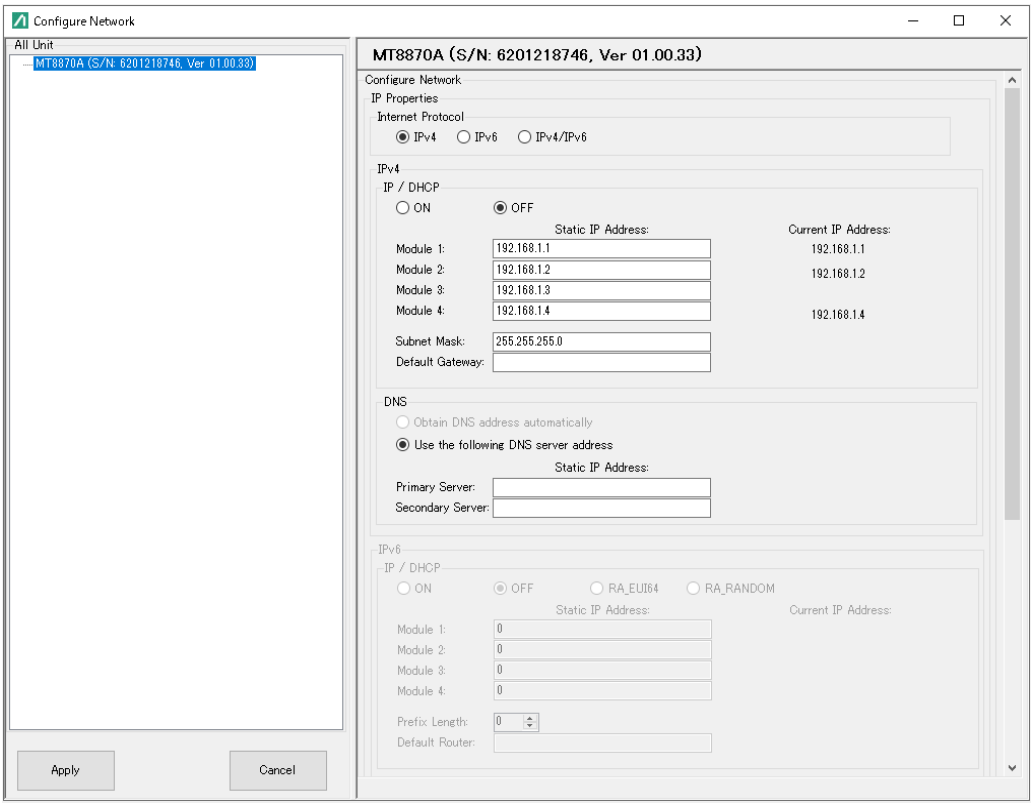


図8.3.5-1　ネットワーク設定画面

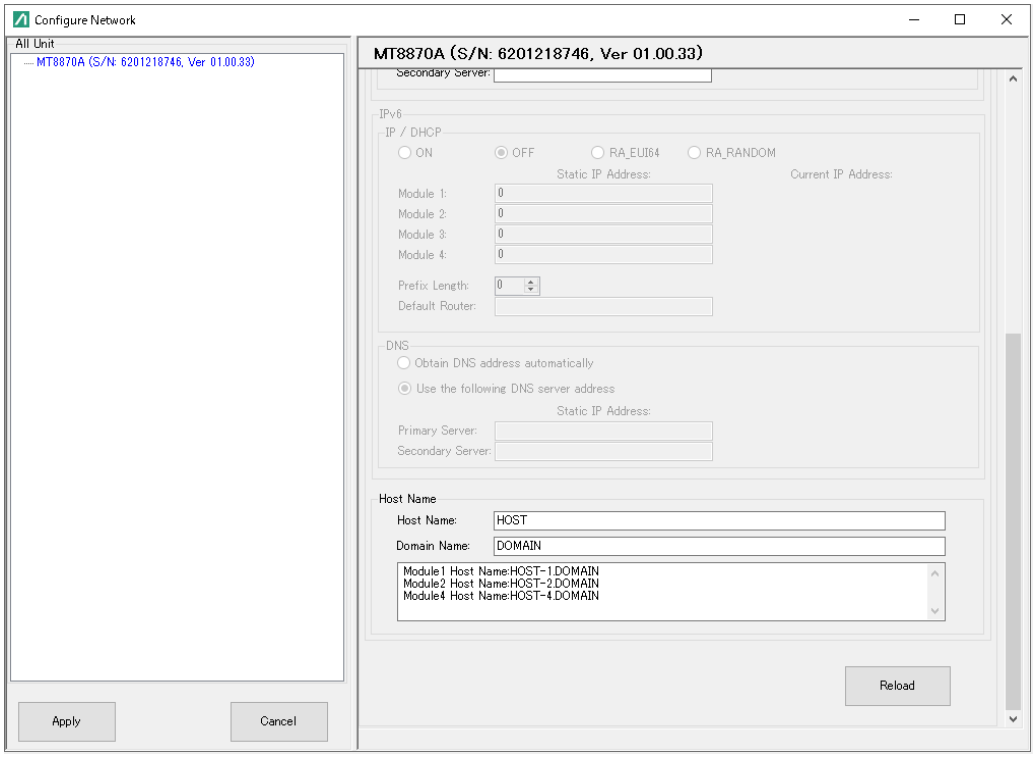


図8.3.5-2　ネットワーク設定画面 2

表8.3.5-1　ネットワーク設定の項目

表示項目	説明
IP Properties	
Internet Protocol	設定する IP のバージョン
IPv4 IP / DHCP DNS	 IP アドレス自動取得のオン/オフ OFF に設定すると，Static IP address の設定値を使用します。 DNS サーバアドレスの取得方法 Use the following DNS server address に設定すると，Static IP address の設定値を使用します。
IPv6 IP / DHCP DNS	 IP アドレス自動取得のオン/オフ OFF に設定すると，Static IP address の設定値を使用します。 DNS サーバアドレスの取得方法 Use the following DNS server address に設定すると，Static IP address の設定値を使用します。
Host Name	ホスト名，ドメイン名 スロット 1～スロット 4 のモジュールの設定が表示されます。

8.3.6 モジュール設定

モジュールの時刻，内部基準信号の周波数調整値，警告音を設定します。

1. ツリービューの MT8870A の ［＋］ をクリックします。
2. [Maintenance] の ［＋］ をクリックします。
3. [Settings] をクリックすると，モジュール設定画面が表示されます。
4. 日付と時刻を設定するには，Date Time の値を入力して [Apply] をクリックします。
5. 内部基準信号の周波数調整値を設定するには，Reference Signal Adjustment の Adjustment value の値を入力して [Apply] をクリックします。
6. 警告音を設定するには，Beep のチェックボックスを設定して，[Apply] をクリックします。
7. RF Mode を設定するには，Module を選択し，NORMAL/RFSEMAPHORE を設定してから，[Apply] をクリックします。一度に複数のモジュールを設定することはできません。
8. MU887002A は GPIB に対応していません。GPIB Address の設定は不要です。
9. Delimiter/Terminator を設定するには，Module を選択し，CRLF/LF/NONE を設定してから，[Apply] をクリックします。一度に複数のモジュールを設定することはできません。
10. CC Switch (Command Converter：コマンド変換機能) を設定するには，Module を選択し，ON/OFF を設定してから，[Apply] をクリックします。一度に複数のモジュールを設定することはできません。

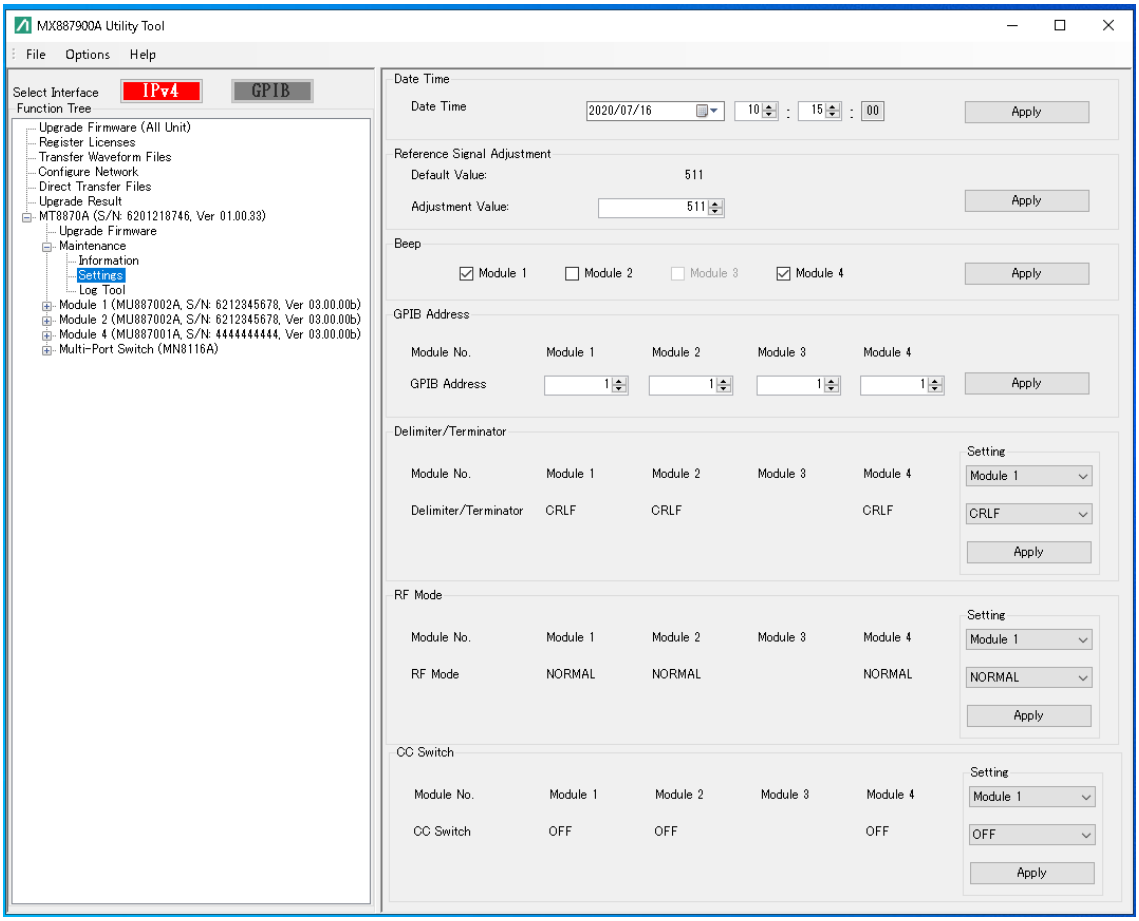


図8.3.6-1　モジュール設定画面

8.3.7 ログの操作

各モジュールのログを取得，クリアをします。

1. ツリービューの MT8870A の ＋ をクリックします。
2. [Maintenance] の ＋ をクリックします。
3. [Log Tool] をクリックすると，ログ操作画面が表示されます。

ログの取得

1. [Export] をクリックすると，ログファイル保存画面が表示されます。
2. フォルダ，ファイル名を入力して，[OK] をクリックすると，ログファイルが保存されます。

ログのクリア

1. [Clear] をクリックすると，確認画面が表示されます。
2. [OK] をクリックするとログがクリアされます。

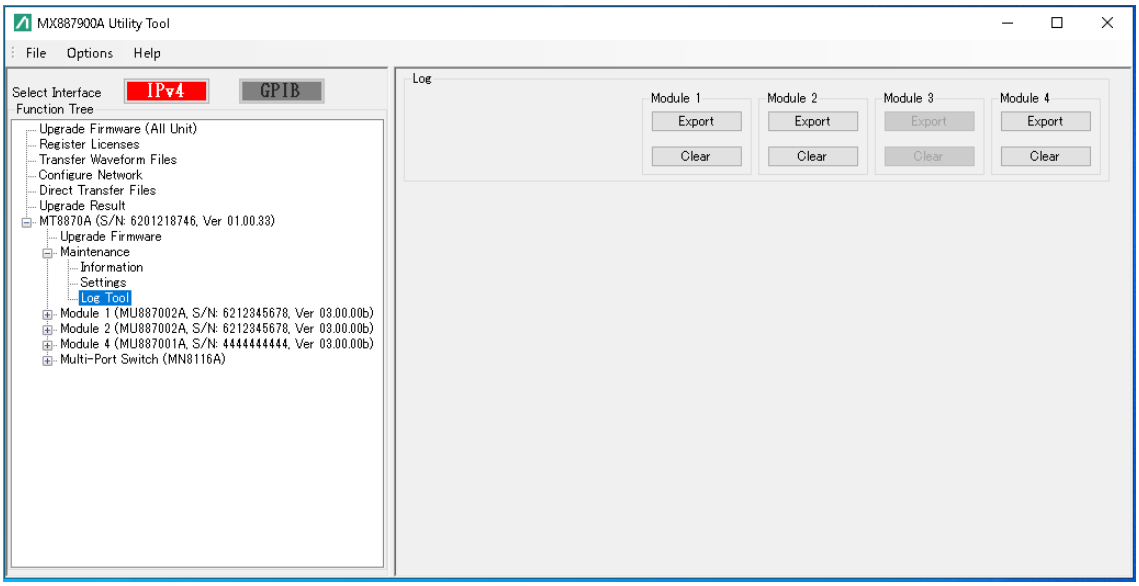


図8.3.7-1　ログ操作画面

8.3.8 アップグレード

IPv4 で制御している場合，MT8870A の Firmware をバージョンアップできます。

内蔵プログラムのバージョンを確認するには，IPv4 で制御して，ツリービューの [Upgrade Result] をクリックします。以前にアップグレードした結果が表示され，バージョンを確認できます。

1. ツリービューの MT8870A の [+] をクリックします。
2. [Upgrade Firmware] をクリックすると，アップグレード画面が表示されます。
3. [Browse] をクリックすると，フォルダ選択画面が表示されます。
4. アップグレードするファイルが入っているフォルダを選択して，[OK] をクリックします。選択したファイルがリストに追加されます。
5. Target Instrument に表示されているモジュールのうち，アップグレードするモジュールのチェックボックスにチェックを付けます。
6. リストからアップグレードするファイルをクリックしてから，[Upgrade] をクリックします。

   注:

   - 図8.3.8-4 のメッセージが表示された場合
     アップグレードが中止されます。この場合は，モジュール記憶媒体の不要なファイルを削除してください。
     モジュール記憶媒体の空き容量をチェックするコマンドは，モジュール内蔵プログラムのバージョンが 01.07.38 以降の場合に使用できます。
   - 図 8.3.8-5 のメッセージが表示された場合
     アップグレードが中止されます。
     MU887002A は，V3.00.00 以降のファームウェアパッケージでなければインストールできません。
     MU887002A-007 搭載時は，V4.00.00 以降のファームウェアパッケージでなければインストールできません。

   Progress 画面にアップグレードの実施状況が表示されます。

   アップグレード実行中は，MU887002A の状態表示ランプ 1～3 が点滅します。更新中のファイルの種類によって，点滅するランプは異なります。ランプの説明は，「付録 D ランプの状態表示」を参照してください。
7. アップグレードが完了すると，MT8870A が再起動します。再起動するまで時間がかかります。
8. MT8870A の再起動が完了すると，Progress 画面に完了結果が表示されます。[OK] をクリックして，Progress 画面を閉じます。

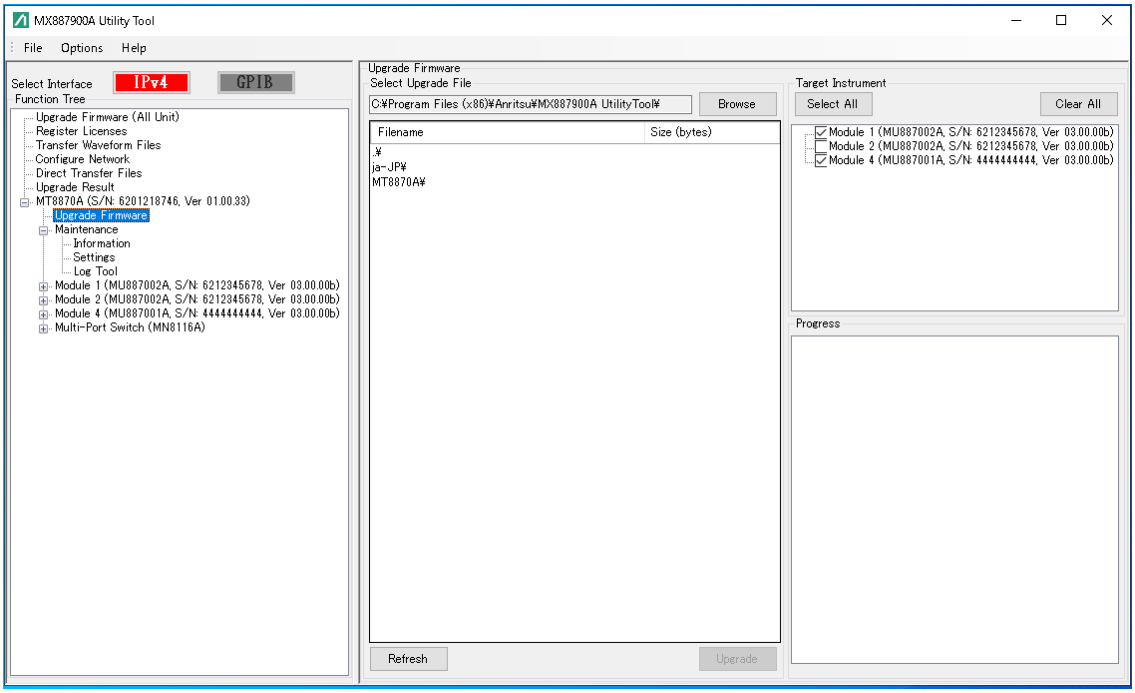


図8.3.8-1　アップグレード画面

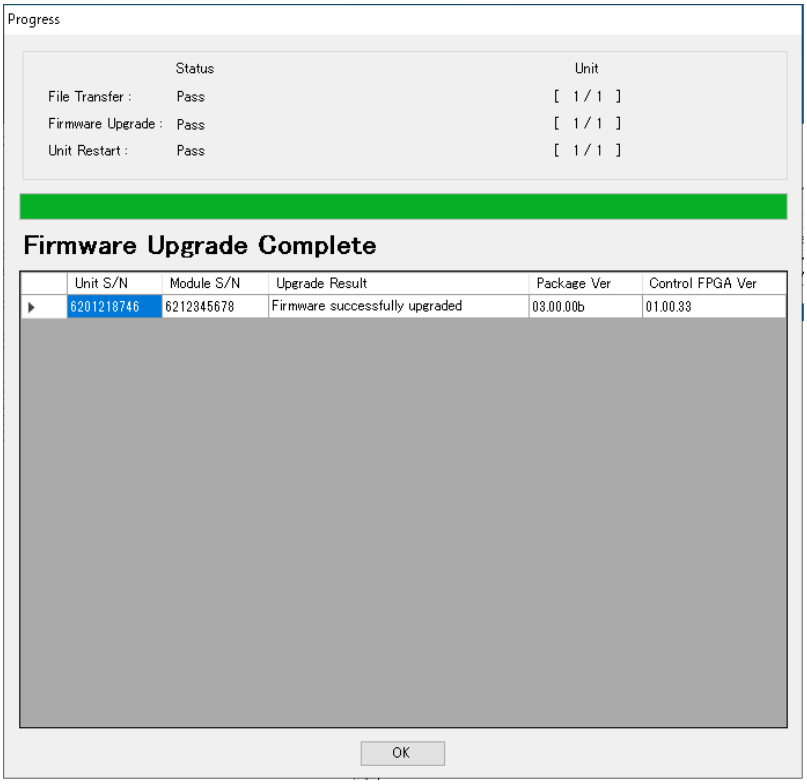


図8.3.8-2　アップグレード完了後の Progress 画面

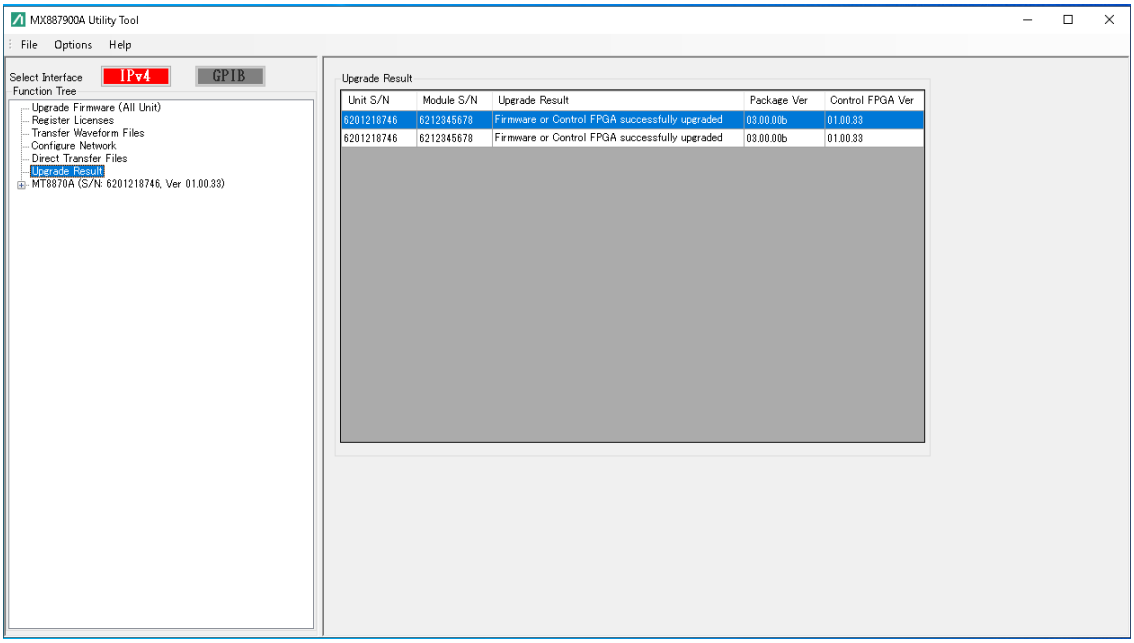


図8.3.8-3　アップグレード結果画面

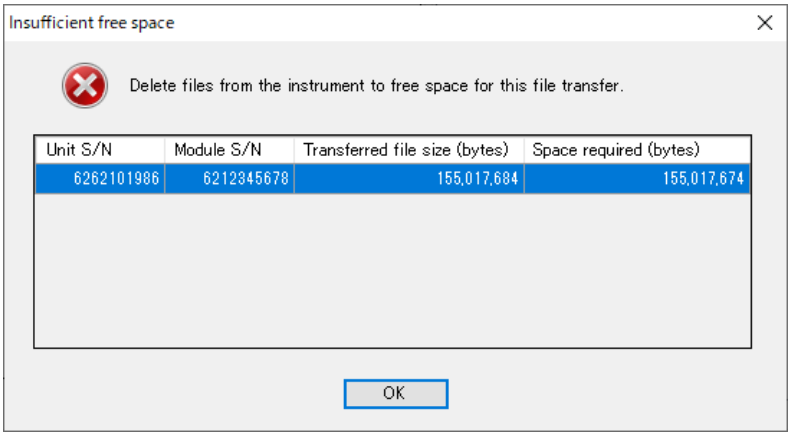


図8.3.8-4　空き容量不足のエラーメッセージ

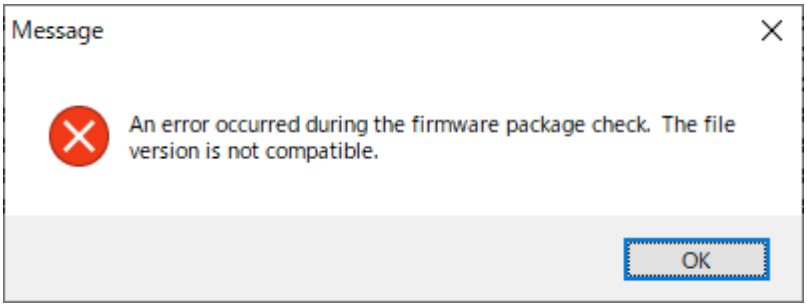


図8.3.8-5　ファームウェア下限バージョン不適合のエラーメッセージ

8.3.9 ファイル転送

IPv4 で制御している場合，モジュールの記憶媒体にファイルを転送できます。

1. ツリービューの MT8870A の ［＋］ をクリックします。
2. Module1～Module4 の ［＋］ をクリックします。
3. [Transfer files] をクリックします。ファイル転送画面が表示されます。

   PC:　　コンピュータ内のファイルが表示されます。

   Instrument:　　モジュール記憶媒体のファイルが表示されます。
4. PC の [Browse] をクリックすると，フォルダ選択画面が表示されます。
5. フォルダを選択すると，ファイルの一覧が表示されます。
6. 転送するファイルを選択して，[OK] をクリックします。

   選択したファイルが PC のリストに追加されます
7. コンピュータからモジュールにファイルを転送するには，[->] をクリックします。

   注:

   図8.3.8-4のメッセージが表示された場合は，ファイルが転送されません。この場合は，モジュール記憶媒体の不要なファイルを削除してください。

   モジュールからコンピュータにファイルを転送するには，[<-] をクリックします。

   転送先に同じ名称のファイルが存在する場合は，ファイルを上書きするか確認するメッセージが表示されます。
8. モジュール記憶媒体のファイルを削除するには，Instrument のリストでファイルを選択して，[Delete] をクリックします。

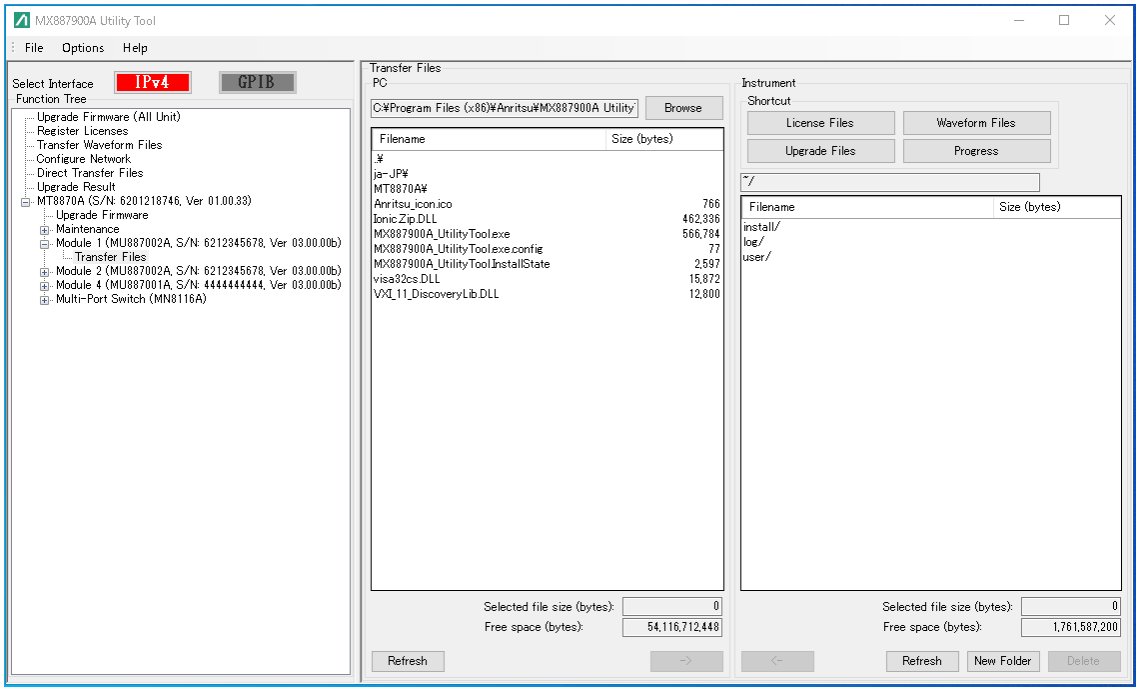


図8.3.9-1　ファイル転送画面

表8.3.9-1　画面の説明

名称	説明
PC	コンピュータのファイルが表示されます。
Browse	ファイルを表示するフォルダを選択します。
Refresh	表示されている内容を更新します。
->	コンピュータからモジュールにファイルを転送します。
Instrument	モジュールのファイルが表示されます。
License files	ライセンス情報のフォルダを表示します。
Upgrade files	内蔵プログラム，データのフォルダを表示します。
Waveform Files	波形ファイルのフォルダを表示します。
Progress	ログファイルのフォルダを表示します。
New Folder	新しいフォルダを作ります。
<-	モジュールからコンピュータにファイルを転送します。
Delete	選択したモジュールのファイル，フォルダを削除します。
Refresh	表示されている内容を更新します。

8.3.10 一括アップグレード

複数の MT8870A の内蔵プログラムをバージョンアップします。
IPv4 で制御している場合に操作できます。内蔵プログラムのバージョンを確認するには，ツリービューの［Upgrade Result］をクリックします。以前にアップグレードした結果が表示され，バージョンを確認できます。

1. ツリービューの［Update Firmware (All Unit)］をクリックします。一括アップグレード画面が表示されます。
2. [Browse] をクリックすると，フォルダ選択画面が表示されます。
3. フォルダを選択して［OK］をクリックします。ファイルの一覧が表示されます。
4. Select Update File のファイル名をクリックして，アップグレードするファイルを選択します。
5. アップグレードする MT8870A のチェックボックスにチェックを付けます。
   Select All:　すべての MT8870A にチェックを付けます。
   Clear All:　すべての MT8870A のチェックを外します。
6. [Upgrade] をクリックします。Progress 画面にアップグレードの実施状況が表示されます。

   注:

   図8.3.8-4 のメッセージが表示された場合，一括アップグレードは中止されます。この場合，モジュール記憶媒体の空き容量が不足しているMT8870Aがありますので，該当するMT8870Aのモジュール記憶媒体の不要なファイルを削除してください。

   アップグレード実行中は，MU887002A の状態表示ランプ 1～3 が点滅します。更新中のファイルの種類によって，点滅するランプは異なります。ランプの説明は，「付録 D ランプの状態表示」を参照してください。
7. アップグレードが完了すると，MT8870A が再起動します。再起動するまで時間がかかります。
8. MT8870A の再起動が完了すると，Progress 画面に完了結果が表示されます。[OK] をクリックして，Progress 画面を閉じます。

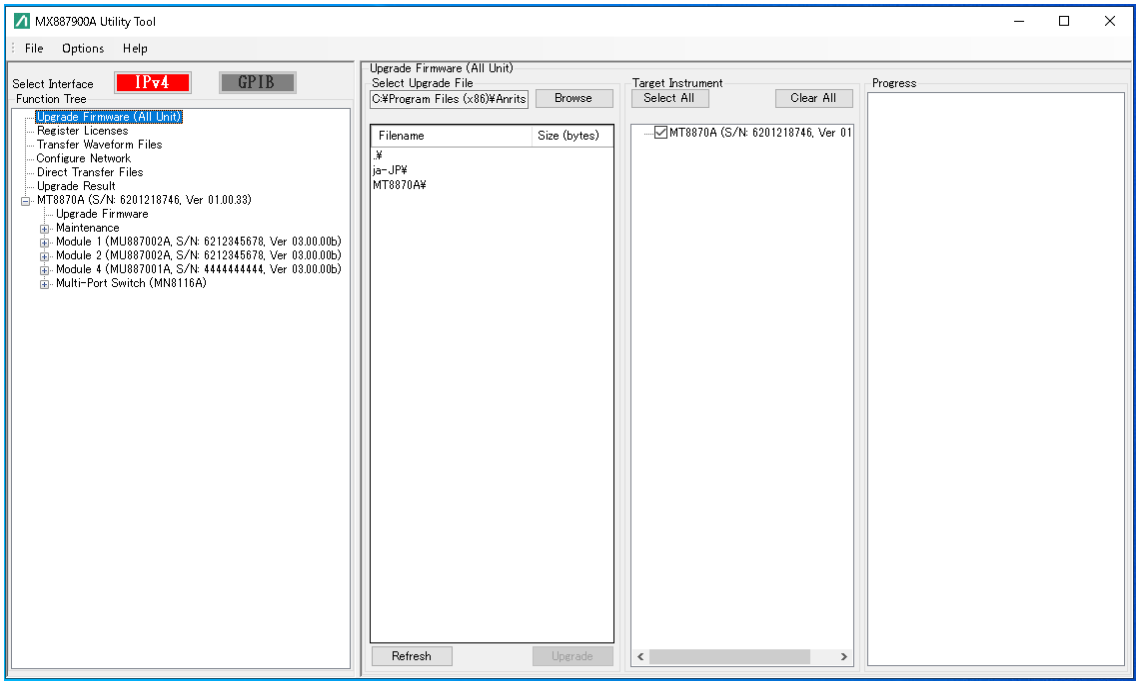


図8.3.10-1　一括アップグレード画面

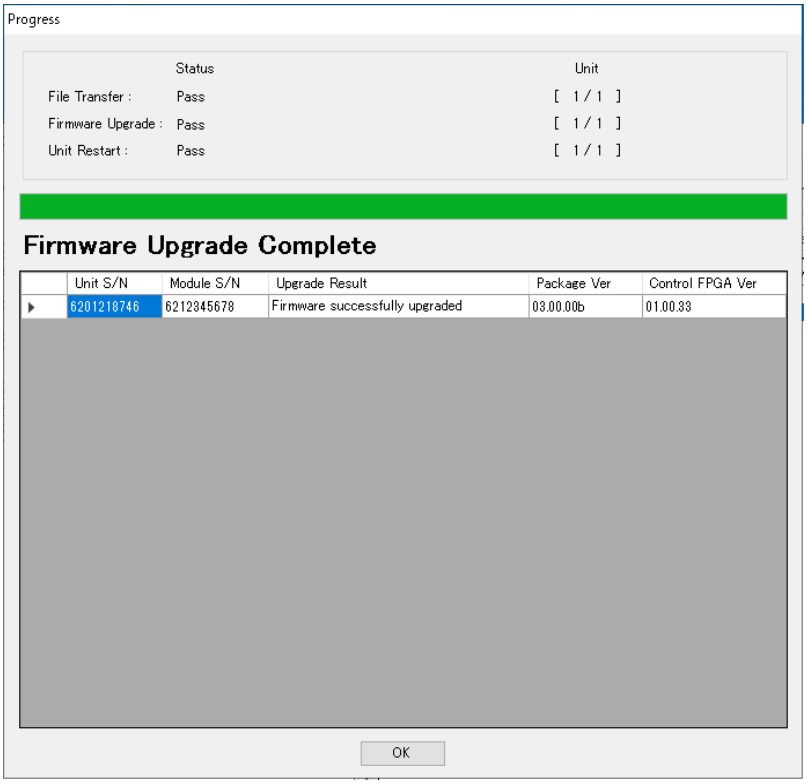


図8.3.10-2　一括アップグレード完了後の Progress 画面

8.3.11ライセンス登録

アプリケーションソフトウェアライセンス，および波形ライセンスを本体に登録します。ライセンスキーは，購入した記憶媒体に格納されています。

IPv4 で制御している場合に操作できます。

1. ツリービューの［Register Licenses］をクリックします。ライセンス登録画面が表示されます。
2. ［Browse］をクリックすると，フォルダ選択画面が表示されます。
3. フォルダを選択して［OK］をクリックします。ファイルの一覧が表示されます。
4. ファイル名をクリックして，ライセンスファイルを選択します。
5. ［Register］をクリックします。

   注：

   図8.3.8-4 のメッセージが表示された場合，ライセンス登録は中止されます。この場合，ライセンス追加対象でモジュール記憶媒体の空き容量が不足しているMT8870A がありますので，該当するMT8870A のモジュール記憶媒体の不要なファイルを削除してください。

   ライセンス登録処理中はメッセージが表示されます。
   ［Stop］をクリックすると，ライセンス登録を中止します。
6. ライセンス登録が終了したら，インタフェースボタンをクリックして，MT8870A との通信を終了します。
7. インタフェースボタンをクリックして，MT8870A との通信を開始します。
8. ツリービューの表示が更新されたら，ツリービューで各 MT8870A の［Information］をクリックします。アプリケーションソフトウェアライセンス，および波形ライセンスが変更されていることを確認します。

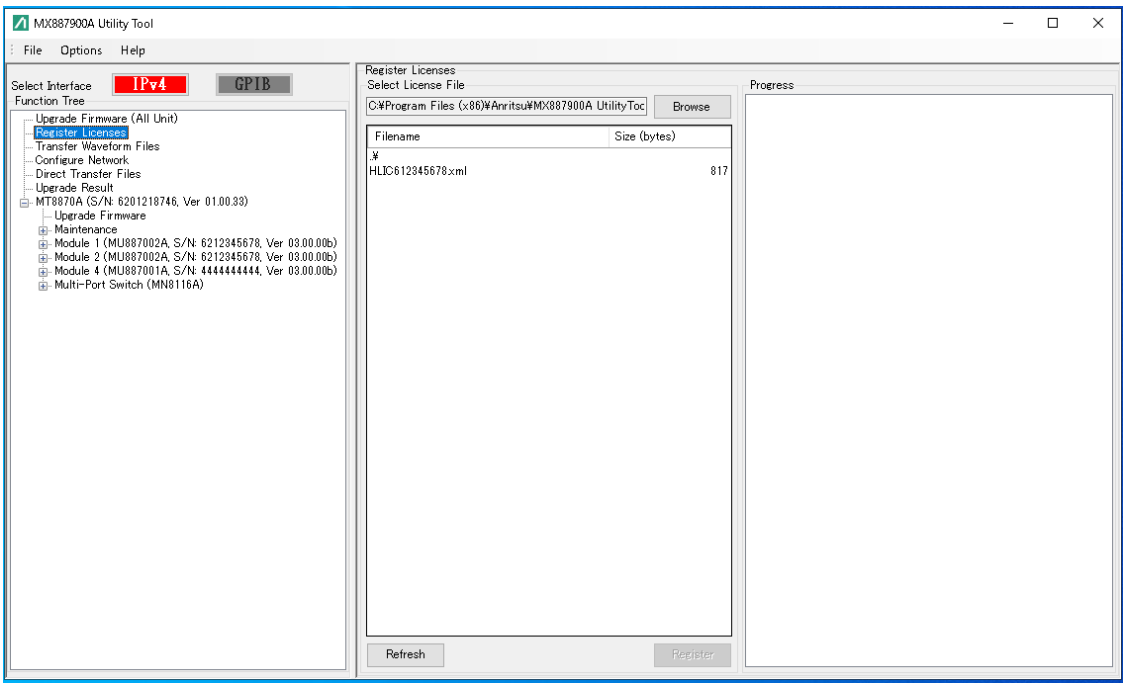


図8.3.11-1　ライセンス登録画面

8.3.12 波形ファイル転送

VSG の波形ファイルをモジュールに転送します。
IPv4 で制御している場合に操作できます。

1. ツリービューの［Transfer Waveform Files］をクリックします。波形ファイル転送画面が表示されます。
2. Select Waveform File の［Browse］をクリックします。フォルダ選択画面が表示されます。
3. フォルダを選択して［OK］をクリックします。ファイルの一覧が表示されます。
4. ファイル名をクリックして，波形ファイルを選択します。

   波形ファイルは，拡張子 wvd のファイルと xml のファイルが両方とも存在する場合のみ表示されます。フォルダ選択画面には，波形ファイルの拡張子は表示されません。
5. Overwrite Processing で，転送先に同じ名称のファイルが存在する場合の処理を選択します。

   Don't Confirm: 確認メッセージを表示しない。
   Always Confirm: 確認メッセージを表示する。
6. Target Instrument のリストで，波形ファイル転送先モジュールのチェックボックスにチェックを付けます。
7. ［Transfer］をクリックすると，波形ファイル転送を開始します。

   注:

   図8.3.8-4のメッセージが表示された場合，波形ファイル転送は中止されます。この場合，波形ファイル転送先となるモジュール記憶媒体の空き容量が不足している MT8870A がありますので，該当する MT8870A のモジュール記憶媒体の不要なファイルを削除してください。

   波形ファイル転送はメッセージが表示されます。［Stop］をクリックすると，波形ファイル転送を中止します。

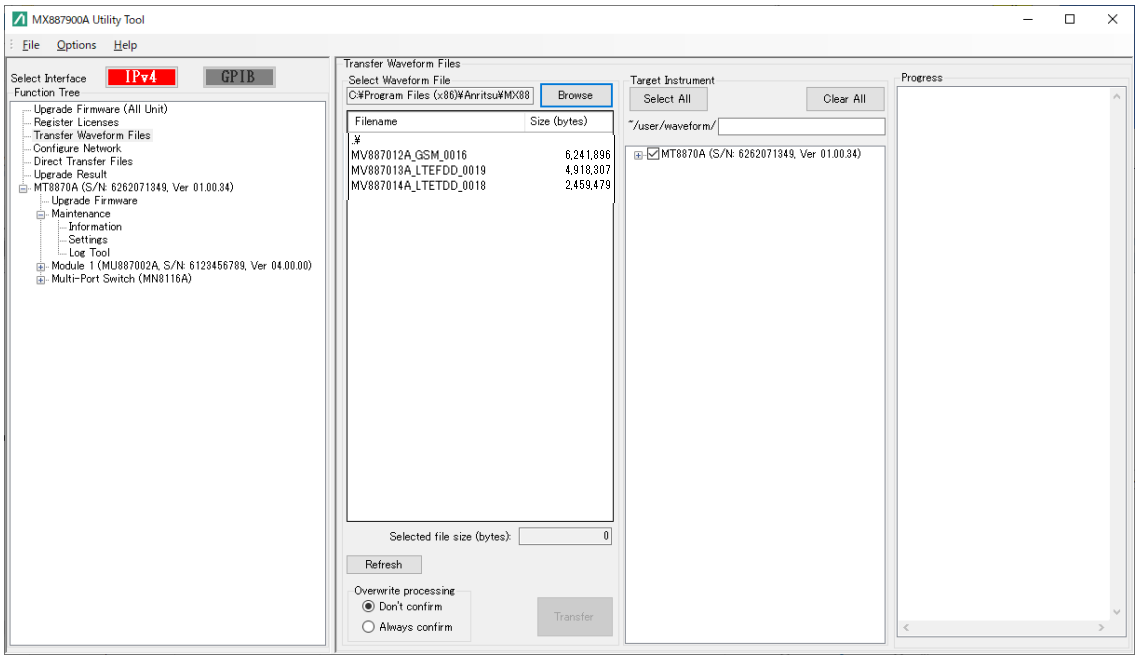


図8.3.12-1　波形ファイル転送画面

8.3.13 直接ファイル転送

直接ファイル転送は，ユーティリティツールが MT8870A を検出できない場合，IP アドレスを入力してファイル転送をする機能です。

IPv4 で制御している場合に操作できます。

1. ツリービューの［Direct Transfer Files］をクリックします。

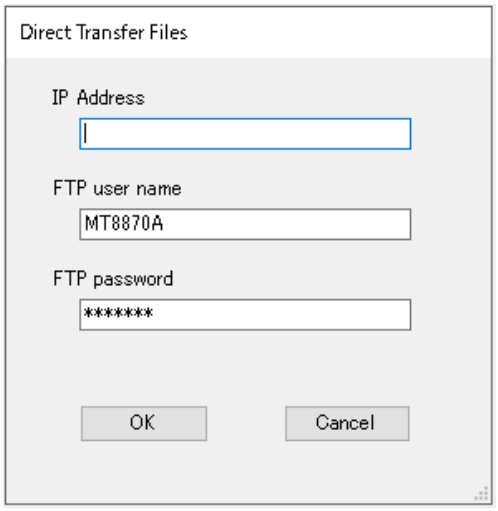


図8.3.13-1 波形ファイル転送画面

2. IP アドレス，FTP ユーザ，パスワードを入力して［OK］をクリックします。
3. 「図8.3.9-1 ファイル転送画面」が表示されます。
「8.3.9 ファイル転送」の説明を参照してください。

ここでは，規格について説明します。製品構成，オプション，応用部品については「1.3 製品構成」を参照してください。

(Typ.)：　Typical は 20～30°C における参考データであり，規格値としては保証していません。

(Nominal)：設計値であり，規格値としては保証していません。

A.1　MU887002A..... A-2
A.2　MU887002A-007..... A-6

A.1　MU887002A

MU887002A の規格について示します。

規格は，一定の周囲温度でウォームアップ 30 分後の値です。

表A.1-1　入出力端子

項目	規格
RF テストポート	TRX 1: Test Port 1～12 TRX 2: Test Port 1～12
コネクタ	N (female)
インピーダンス	50 Ω (Nominal)
VSWR	＜1.4　(20～30°C)　400 MHz≦周波数＜450 MHz, ＜1.3　(20～30°C)　450 MHz≦周波数≦2700 MHz ＜1.4　(20～30°C)　2700 MHz＜周波数≦3800 MHz ＜1.4　(20～30°C)　3800 MHz＜周波数≦6000 MHz
最大入力レベル	+35 dBm

表A.1-2 信号発生器

項目	規格
TRX 1/2, Test Port 1～12 共通	
周波数	
周波数設定範囲	400～6000 MHz
周波数分解能	1 Hz
周波数確度	MT8870A ユニバーサルワイヤレステストセットの基準周波数確度による
振幅	
レベル設定範囲	–130～0 dBm
性能の保証範囲上限	–5 dBm　400 MHz≦周波数≦3800 MHz –8 dBm　3800 MHz＜周波数≦6000 MHz ただし, 変調波の場合は RMS 値
レベル設定分解能	0.1 dB
レベル確度	CW, 校正実行後にて 400 MHz≦周波数≦3800 MHz, –120 dBm≦出力レベル≦–5 dBm ±0.7 dB (Typ.) ±1.0 dB (20～30°C) 3800 MHz＜周波数≦6000 MHz, –100 dBm≦出力レベル≦–8 dBm ±1.0 dB (Typ.) ±1.3 dB (20～30°C)
ケーブルロス設定時のレベル直線性	Broadcast 時, かつ Test Port ごとの CableLoss 値が異なるとき, ケーブルロス 0 dB 基準にて, ±0.2 dB (Typ.)　ただし, テストポート間の損失補正テーブル値の差≦8 dB
出力レベル偏差	Broadcast 時 ≦0.6 dB (Nominal)
スプリアス応答	
高調波ひずみ	800 MHz≦高調波周波数≦6000 MHz, –120 dBm≦出力レベル≦–5 dBm, CW にて, ＜–25 dBc (20～30°C)
ベクトル変調	
変調帯域幅	最大 200 MHz

表A.1-3　信号解析

項目	規格
TRX 1/2, Test Port 1～12 共通	
周波数	
周波数設定範囲	400～6000 MHz
周波数分解能	1 Hz
振幅	
レベル設定範囲	–65～+35 dBm (CW にて)
レベル設定分解能	0.1 dB
レベル確度	CW, 測定帯域幅 = 300 kHz, RBW = 100 kHz, 校正実行後, 設定周波数に等しい周波数の被測定信号, 設定レベルに等しいレベルの被測定信号にて, 400 MHz≦周波数≦3800 MHz ±0.3 dB (Typ.)　–30 dBm≦レベル≦+35 dBm ±0.5 dB (20～30°C)　–30 dBm≦レベル≦+35 dBm ±0.7 dB (20～30°C)　–55 dBm≦レベル＜–30 dBm ±0.9 dB (20～30°C)　–65 dBm≦レベル＜–55 dBm 3800 MHz＜周波数≦6000 MHz ±0.7 dB (20～30°C)　–30 dBm≦レベル≦+35 dBm ±0.9 dB (20～30°C)　–55 dBm≦レベル＜–30 dBm ±1.1 dB (20～30°C)　–65 dBm≦レベル＜–55 dBm
レベル直線性	CW, 測定帯域幅 = 300 kHz, RBW = 100 kHz にて, ±0.2 dB (20～30°C)　0～–40 dB, 入力レベル≧–55 dBm ±0.4 dB (20～30°C)　0～–40 dB, 入力レベル≧–65 dBm
変調解析	
最大解析帯域幅	25 MHz　400 MHz≦設定周波数＜500 MHz 80 MHz　500 MHz≦設定周波数＜600 MHz 200 MHz　600 MHz≦設定周波数≦6000 MHz ただし, (設定周波数 – 解析帯域幅 / 2)＜400 MHz, 6000 MHz＜(設定周波数 + 解析帯域幅 / 2) の場合の測定結果は保証範囲外とする。

表A.1-4 一般性能

項目	規格
インタフェース トリガ機能 リモート制御	 MT8870A 背面パネルのトリガコネクタを経由して，トリガ信号の入出力が可能 イーサネット：MT8870A のインタフェース経由でリモート制御可能
寸法	181 (W) × 193.6 (H) × 325 (D) mm (突起物を除く)
質量	≦12.5 kg
動作温度範囲	+5～+40°C
保管温度範囲	–20～+60°C

A.2　MU887002A-007

7GHz 拡張機能オプション MU887002A-007 搭載時の規格について示します。

規格は，一定の周囲温度でウォームアップ 30 分後の値です。

表A.2-1　入出力端子

項目	規格
RF テストポート	TRX 1: Test Port 1～12 TRX 2: Test Port 1～12
コネクタ	N (female)
インピーダンス	50 Ω (Nominal)
VSWR	TRX 1/2, Test Port 1～4 ＜1.4 (20～30°C)　400 MHz≦周波数＜450 MHz ＜1.3 (20～30°C)　450 MHz≦周波数≦2700 MHz ＜1.4 (20～30°C)2700 MHz＜周波数≦3800 MHz ＜1.4 (20～30°C)3800 MHz＜周波数≦6000 MHz TRX 1/2, Test Port 5～12 設定周波数＜5900 MHz ＜1.4 (20～30°C)　400 MHz≦周波数＜450 MHz ＜1.3 (20～30°C)　450 MHz≦周波数≦2700 MHz ＜1.4 (20～30°C)2700 MHz＜周波数≦3800 MHz ＜1.4 (20～30°C)3800 MHz＜周波数＜6000 MHz 5900 MHz≦設定周波数 ＜1.7 (20～30°C)5800 MHz≦周波数≦7400 MHz
最大入力レベル	TRX 1/2, Test Port 1～4 +35 dBm　　400 MHz≦設定周波数≦6000 MHz TRX 1/2, Test Port 5～12 +35 dBm　　400 MHz≦設定周波数＜5900 MHz +30 dBm　　5900 MHz≦設定周波数≦7300 MHz

表A.2-2 信号発生器

項目	規格
TRX 1/2, Test Port 1～12 共通	
周波数	
周波数設定範囲	TRX 1/2, Test Port 1～4 400～6000 MHz TRX 1/2, Test Port 5～12 400～7300 MHz
周波数分解能	1 Hz
周波数確度	MT8870A ユニバーサルワイヤレステストセットの基準周波数確度による
振幅	
レベル設定範囲	–130～0 dBm
性能の 保証範囲上限	TRX 1/2, Test Port 1～4 –5 dBm 400 MHz≦周波数≦3800 MHz –8 dBm 3800 MHz＜周波数≦6000 MHz ただし, 変調波の場合は RMS 値 TRX 1/2, Test Port 5～12 –5 dBm 400 MHz≦周波数≦3800 MHz –8 dBm 3800 MHz＜周波数＜5900 MHz –10 dBm 5900 MHz≦周波数≦7300 MHz ただし, 変調波の場合は RMS 値
レベル設定分解能	0.1 dB
レベル確度	CW, 校正実行後にて TRX 1/2, Test Port 1～4 400 MHz≦周波数≦3800 MHz, –120 dBm≦出力レベル≦–5 dBm ±0.7 dB (Typ.) ±1.0 dB (20～30°C) –5 dBm＜出力レベル≦0 dBm ±0.7 dB (Typ.) 3800 MHz＜周波数≦6000 MHz, –100 dBm≦出力レベル≦–8 dBm ±1.0 dB (Typ.) ±1.3 dB (20～30°C) –8 dBm＜出力レベル≦0 dBm ±1.0 dB (Typ.)

表A.2-2　信号発生器 (続き)

項目	規格
振幅 　レベル確度 (続き)	TRX 1/2, Test Port 5～12 400 MHz≦周波数≦3800 MHz, 　–120 dBm≦出力レベル≦–5 dBm 　　±0.7 dB (Typ.) 　　±1.0 dB (20～30°C) 　–5 dBm＜出力レベル≦0 dBm 　　±0.7 dB (Typ.) 3800 MHz＜周波数＜5900 MHz, 　–100 dBm≦出力レベル≦–8 dBm 　　±1.0 dB (Typ.) 　　±1.3 dB (20～30°C) 　–8 dBm＜出力レベル≦0 dBm 　　±1.0 dB (Typ.) 5900 MHz≦周波数≦7300 MHz, 　–100 dBm≦出力レベル≦–10 dBm 　　±1.0 dB (Typ.) 　　±1.3 dB (20～30°C) 　–10 dBm＜出力レベル≦0 dBm 　　±1.0 dB (Typ.)
ケーブルロス設定時のレベル直線性	TRX 1/2, Test Port 1～4, 400 MHz≦周波数≦6000 MHz, Broadcast 時, かつ Test Port ごとの CableLoss 値が異なるとき, ケーブルロス 0 dB 基準にて, 　　±0.2 dB (Typ.)　ただし, テストポート間の損失補正テーブル値の差≦8 dB TRX 1/2, Test Port 5～12, 400 MHz≦周波数≦7300 MHz, Broadcast 時, かつ Test Port ごとの CableLoss 値が異なるとき, ケーブルロス 0 dB 基準にて, 　　±0.2 dB (Typ.)　ただし, テストポート間の損失補正テーブル値の差≦8 dB
出力レベル偏差	TRX 1/2, Test Port 1～12 Broadcast 時 　　≦0.6 dB (Nominal)
スプリアス応答 　高調波ひずみ	TRX 1/2, Test Port 1～4, 800 MHz≦高調波周波数≦6000 MHz, –120 dBm≦出力レベル≦–5 dBm, CW にて, 　　＜–25 dBc (20～30°C) TRX 1/2, Test Port 5～12, 800 MHz≦高調波周波数≦7300 MHz, –120 dBm≦出力レベル≦–5 dBm, CW にて, 　　＜–25 dBc (20～30°C)
ベクトル変調 　変調帯域幅	最大 200 MHz

表A.2-3 信号解析

項目	規格
TRX 1/2, Test Port 1～12 共通	
周波数	
周波数設定範囲	TRX 1/2, Test Port 1～4 400～6000 MHz TRX 1/2, Test Port 5～12 400～7300 MHz
周波数分解能	1 Hz
振幅	
レベル設定範囲	TRX 1/2, Test Port 1～4 –65～+35 dBm (CW にて)　400 MHz≦設定周波数≦6000 MHz TRX 1/2, Test Port 5～12 –65～+35 dBm (CW にて)　400 MHz≦設定周波数＜5900 MHz –65～+30 dBm (CW にて)　5900 MHz≦設定周波数≦7300 MHz
レベル設定分解能	0.1 dB
レベル確度	CW, 測定帯域幅 = 300 kHz, RBW = 100 kHz, 校正実行後, 設定周波数に等しい周波数の被測定信号, 設定レベルに等しいレベルの被測定信号にて, TRX 1/2, Test Port 1～4 400 MHz≦周波数≦3800 MHz ±0.3 dB (Typ.)　–30 dBm≦レベル≦+35 dBm ±0.5 dB (20～30°C)　–30 dBm≦レベル≦+35 dBm ±0.7 dB (20～30°C)　–55 dBm≦レベル＜–30 dBm ±0.9 dB (20～30°C)　–65 dBm≦レベル＜–55 dBm 3800 MHz＜周波数≦6000 MHz ±0.7 dB (20～30°C)　–30 dBm≦レベル≦+35 dBm ±0.9 dB (20～30°C)　–55 dBm≦レベル＜–30 dBm ±1.1 dB (20～30°C)　–65 dBm≦レベル＜–55 dBm TRX 1/2, Test Port 5～12 400 MHz≦周波数≦3800 MHz ±0.3 dB (Typ.)　–30 dBm≦レベル≦+35 dBm ±0.5 dB (20～30°C)　–30 dBm≦レベル≦+35 dBm ±0.7 dB (20～30°C)　–55 dBm≦レベル＜–30 dBm ±0.9 dB (20～30°C)　–65 dBm≦レベル＜–55 dBm 3800 MHz＜周波数＜5900 MHz ±0.7 dB (20～30°C)　–30 dBm≦レベル≦+35 dBm ±0.9 dB (20～30°C)　–55 dBm≦レベル＜–30 dBm ±1.1 dB (20～30°C)　–65 dBm≦レベル＜–55 dBm 5900 MHz≦周波数≦7300 MHz ±1.0 dB (20～30°C)　–30 dBm≦レベル≦+30 dBm ±1.2 dB (20～30°C)　–55 dBm≦レベル＜–30 dBm ±1.8 dB (20～30°C)　–65 dBm≦レベル＜–55 dBm

表A.2-3　信号解析 (続き)

項目	規格
振幅 　レベル直線性	CW, 測定帯域幅 = 300 kHz, RBW = 100 kHz にて, TRX 1/2, Test Port 1～4 400 MHz≦周波数≦6000 MHz ±0.2 dB (20～30°C)　0～–40 dB, 入力レベル≧–55 dBm ±0.4 dB (20～30°C)　0～–40 dB, 入力レベル≧–65 dBm TRX 1/2, Test Port 5～12 400 MHz≦周波数≦7300 MHz ±0.2 dB (20～30°C)　0～–40 dB, 入力レベル≧–55 dBm ±0.4 dB (20～30°C)　0～–40 dB, 入力レベル≧–65 dBm
変調解析 　最大解析帯域幅	TRX 1/2, Test Port 1～4 25 MHz　400 MHz≦設定周波数＜500 MHz 80 MHz　500 MHz≦設定周波数＜600 MHz 200 MHz　600 MHz≦設定周波数≦6000 MHz ただし, (設定周波数 – 解析帯域幅 / 2)＜400 MHz, 6000 MHz＜(設定周波数 + 解析帯域幅 / 2) の場合の測定結果は保証範囲外とする。 TRX 1/2, Test Port 5～12 25 MHz　400 MHz≦設定周波数＜500 MHz 80 MHz　500 MHz≦設定周波数＜600 MHz 200 MHz　600 MHz≦設定周波数≦7300 MHz ただし, (設定周波数 – 解析帯域幅 / 2)＜400 MHz の場合の測定結果は保証範囲外とする。

付録 B　ソフトウェアライセンスについて

本製品には，以下の表に示すパッケージソフトウェアを含んでいます。
ただし，当社のソフトウェア使用許諾の対象外です。

各ライセンスの詳細は，下記 URL を参照してください。
https://my.anritsu.com/myproduct/home

表B-1　パッケージ名とソフトウェアライセンス

パッケージ名	ライセンス	ライセンス参照番号
acl	LGPLv2+	(*03)
acpid	GPLv2+	(*01)
apr	ASL 2.0 and BSD with advertising and ISC and BSD	(*06)(*08)(*12)(*73)
apr-uti	ASL 2.0	(*06)
audit	GPLv2+	(*01)
audit-libs	GPLv2+	(*01)
authconfig	GPLv2+	(*01)
autoconf	GPLv2+ and GFDL	(*01)(*76)
autogen-libopts	LGPLv3+	(*04)
automake	GPLv2+ and GFDL and Public Domain and MIT	(*01)(*76)(*73)
avahi-libs	LGPLv2+	(*03)
basesystem	Public Domain	–
bash	GPLv3+	(*04)
bc	GPLv2+	(*01)
bind-libs-lite	ISC	(*12)
bind-license	ISC	(*12)
binutils	GPLv3+	(*02)
bison	GPLv3+	(*02)
boost-date-time	Boost and MIT and Python	(*07)(*73)(*19)
boost-system	Boost and MIT and Python	(*07)(*73)(*19)
boost-thread	Boost and MIT and Python	(*07)(*73)(*19)
btrfs-progs	GPLv2	(*01)
byacc	Public Domain	–
bzip2	BSD	(*72)
bzip2-libs	BSD	(*72)
ca-certificates	Public Domain	–
centos-logos	Copyright c 2014 The CentOS Project.　All rights reserved.	(*77)
centos-release	GPLv2	(*01)
chkconfig	GPLv2	(*01)
chrony	GPLv2	(*01)
coreutils	GPLv3+	(*02)
cpio	GPLv3+	(*02)
cpp	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*02)(*30)(*72)
cracklib	LGPLv2+	(*03)
cracklib-dicts	LGPLv2+	(*03)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
cronie	MIT and BSD and ISC and GPLv2+	(*73)(*72)(*12)(*01)
cronie-anacron	MIT and BSD and ISC and GPLv2+	(*73)(*72)(*12)(*01)
crontabs	Public Domain and GPLv2	(*01)
cryptsetup-libs	GPLv2+ and LGPLv2+	(*01)(*03)
cscope	BSD and GPLv2+	(*72)(*01)
ctags	GPLv2+ and LGPLv2+ and Public Domain	(*01)(*03)
curl	MIT	(*73)
cyrus-sasl-lib	BSD with advertising	(*08)
dbus	GPLv2+ or AFL	(*01)(*05)
dbus-glib	AFL and GPLv2+	(*05)(*01)
dbus-libs	GPLv2+ or AFL	(*01)(*05)
dbus-python	MIT	(*73)
device-mapper	GPLv2	(*01)
device-mapper-libs	LGPLv2	(*03)
dhclient	ISC	(*12)
dhcp-common	ISC	(*12)
dhcp-libs	ISC	(*12)
diffstat	MIT	(*73)
diffutils	GPLv3+	(*02)
dmidecode	GPLv2+	(*01)
dosfstools	GPLv3+	(*02)
doxygen	GPL+	(*02)
dracut	GPLv2+ and LGPLv2+	(*01)(*03)
dracut-config-rescue	GPLv2+ and LGPLv2+	(*01)(*03)
dracut-network	GPLv2+ and LGPLv2+	(*01)(*03)
dwz	GPLv2+ and GPLv3+	(*01)(*02)
dyninst	LGPLv2+	(*03)
e2fsprogs	GPLv2	(*01)
e2fsprogs-libs	GPLv2 and LGPLv2	(*01)(*03)
ebtables	GPLv2+	(*01)
ed	GPLv3+, GFDL	(*02)(*76)
efivar-libs	LGPLv2.1	(*03)
elfutils	GPLv3+ and (GPLv2+ or LGPLv3+)	(*02)(*01)(*04)
elfutils-default-yama-scope	GPLv2+ or LGPLv3+	(*01)(*04)
elfutils-libelf	GPLv2+ or LGPLv3+	(*01)(*04)
elfutils-libs	GPLv2+ or LGPLv3+	(*01)(*04)
emacs-filesystem	GPLv3+	(*02)
ethtool	GPLv2	(*01)
expat	MIT	(*73)
file	BSD	(*72)
file-libs	BSD	(*72)
filesystem	Public Domain	–
findutils	GPLv3+	(*02)
fipscheck	BSD	(*72)
fipscheck-lib	BSD	(*72)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
firewalld	GPLv2+	(*01)
firewalld-filesystem	GPLv2+	(*01)
flex	BSD and LGPLv2+	(*03)(*72)
freetype	(FTL or GPLv2+) and BSD and MIT and Public Domain and zlib with acknowledgement	(*09)(*01)(*72)(*73)(*79)
fxload	GPLv2+	(*01)
ftp	BSD with advertising	(*08)
gawk	GPLv3+ and GPL and LGPLv3+ and LGPL and BSD	(*02)
gcc	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*01)(*84)(*02)
gcc-c++	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*01)(*84)(*02)
gcc-gfortran	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*01)(*84)(*02)
gdb	GPLv3+ and GPLv3+ with exceptions and GPLv2+ and GPLv2+ with exceptions and GPL+ and LGPLv2+ and BSD and Public Domain	(*01)(*02)
gdbm	GPLv3+	(*02)
GeoIP	LGPLv2+ and GPLv2+ and CC-BY-SA	(*01)(*03)(*83)
gettext	GPLv3+ and LGPLv2+	(*02)(*03)
gettext-common-devel	GPLv3+	(*02)
gettext-devel	LGPLv2+ and GPLv3+	(*02)(*03)
gettext-libs	LGPLv2+ and GPLv3+	(*02)(*03)
git	GPLv2	(*01)
glib2	LGPLv2+	(*03)
glibc	LGPLv2+ and LGPLv2+ with exceptions and GPLv2+	(*01)(*02)
glibc-common	LGPLv2+ and LGPLv2+ with exceptions and GPLv2+	(*01)(*02)
glibc-devel	LGPLv2+ and LGPLv2+ with exceptions and GPLv2+	(*01)(*02)
glibc-headers	LGPLv2+ and LGPLv2+ with exceptions and GPLv2+	(*01)(*02)
gmp	LGPLv3+ or GPLv2+	(*01)(*02)
gnupg2	GPLv3+	(*02)
gnutls	GPLv3+ and LGPLv2+	(*02)(*03)
gobject-introspection	GPLv2+, LGPLv2+, MIT	(*01)(*03)(*73)
gpgme	LGPLv2+	(*03)
gpg-pubkey	pubkey	–
grep	GPLv3+	(*02)
groff-base	GPLv3+ and GFDL and BSD and MIT	(*02)(*72)(*73)(*76)
grub2	GPLv3+	(*02)
grub2-common	GPLv3+	(*02)
grub2-pc	GPLv3+	(*02)
grub2-pc-modules	GPLv3+	(*02)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
grub2-tools	GPLv3+	(*02)
grub2-tools-extra	GPLv3+	(*02)
grub2-tools-minimal	GPLv3+	(*02)
grubby	GPLv2+	(*01)
gzip	GPLv3+ and GFDL	(*02)(*76)
hardlink	GPL+	(*02)
hostname	GPLv2+	(*01)
hwdata	GPLv2+	(*01)
indent	GPLv3+	(*03)
info	GPLv3+	(*03)
initscripts	GPLv2 and GPLv2+	(*01)
intltool	GPLv2 with exceptions	(*01)
iprutils	CPL	(*75)
ipset	GPLv2	(*01)
ipset-libs	GPLv2	(*01)
iproute	GPLv2+ and Public Domain	(*01)
iptables	GPLv2	(*01)
iptables-services	GPLv2	(*01)
iputils	BSD and GPLv2+	(*01)
irqbalance	GPLv2	(*01)
iwl100-firmware	Redistributable, no modification permitted	(*80)
iwl105-firmware	Redistributable, no modification permitted	(*80)
iwl135-firmware	Redistributable, no modification permitted	(*80)
iwl2000-firmware	Redistributable, no modification permitted	(*80)
iwl2030-firmware	Redistributable, no modification permitted	(*80)
iwl3160-firmware	Redistributable, no modification permitted	(*80)
iwl6000g2a-firmware	Redistributable, no modification permitted	(*80)
iwl6000g2b-firmware	Redistributable, no modification permitted	(*80)
iwl7260-firmware	Redistributable, no modification permitted	(*80)
iwl7265-firmware	Redistributable, no modification permitted	(*80)
kbd	GPLv2+	(*01)(*10)
kbd-legacy	GPLv2+	(*01)
kbd-misc	GPLv2+	(*01)
kernel	GPLv2	(*01)
kernel-devel	GPLv2	(*01)
kernel-headers	GPLv2	(*01)
kernel-tools	GPLv2	(*01)
kernel-tools-libs	GPLv2	(*01)
kexec-tools	GPLv2	(*01)
keyutils-libs	GPLv2+ and LGPLv2+	(*01)(*03)
kmod	LGPLv2+	(*03)
kmod-libs	LGPLv2+	(*03)
kpartx	GPL+	(*02)
krb5-libs	MIT	(*73)
libfastjson	MIT	(*73)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
libsemanage	LGPLv2+	(*03)
libpciaccess	MIT	(*73)
lua	MIT	(*73)
libgcrypt	LGPLv2+	(*03)
libedit	BSD	(*72)
libassuan	LGPLv2+ and GPLv3+	(*01)(*02)
libidn	LGPLv2+ and GPLv3+ and GFDL	(*76)(*01)(*02)
libstdc++	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*72)(*01)(*02)(*03)
libgomp	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*72)(*01)(*02)(*03)
libstdc++	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*72)(*01)(*02)(*03)
libmodman	LGPLv2+	(*03)
libcroco	LGPLv2	(*03)
lsscsi	GPLv2+	(*01)
linux-firmware	GPL+ and GPLv2+ and MIT and Redistributable, no modification permitted	(*73)(*03)
libsepol	LGPLv2+	(*03)
libselinux	Public Domain	–
libdb	BSD and LGPLv2 and Sleepycat	(*31)
libuuid	BSD	(*37)
libdrm	MIT	(*73)
libattr	LGPLv2+	(*03)
libacl	LGPLv2+	(*03)
libxml2	MIT	(*73)
libgpg-error	LGPLv2+	(*03)
libmpc	LGPLv3+ and GFDL	(*76)(*02)
libestr	LGPLv2+	(*03)
libtool	GPLv2+ and LGPLv2+ and GFDL	(*76)(*01)(*03)
less	GPLv3+	(*03)
lsof	zlib and Sendmail and LGPLv2+	(*03)(*79)
libunistring	LGPLv3+	(*04)
libnfnetlink	GPLv2+	(*01)
lzo	GPLv2+	(*01)
libss	MIT	(*73)
libdwarf	LGPLv2	(*03)
libselinux-utils	Public Domain	–
libproxy	LGPLv2+	(*03)
libverto	MIT	(*73)
libblkid	LGPLv2+	(*03)
libpwquality	BSD or GPLv2+	(*72)(*01)
logrotate	GPL+	(*02)
libcurl	MIT	(*73)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
libuser	LGPLv2+	(*03)
libgcc	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*30)
libstdc++-devel	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*02)(*04)(*72)
libseccomp	LGPLv2	(*03)
libpcap	BSD with advertising	(*72)
libpipeline	GPLv3+	(*03)
libutempter	LGPLv2+	(*03)
libcom_err	MIT	(*73)
libcap-ng	LGPLv2+	(*03)
libcap	LGPLv2+	(*03)
libffi	MIT and Public Domain	(*73)
libmnl	LGPLv2+	(*03)
libtasn1	GPLv3+ and LGPLv2+	(*02)(*03)
libsysfs	LGPLv2+	(*03)
libnetfilter_conntrack	GPLv2+	(*01)
libquadmath	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*02)(*04)(*72)
libgfortran	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*02)(*04)(*72)
libquadmath-devel	GPLv3+ and GPLv3+ with exceptions and GPLv2+ with exceptions and LGPLv2+ and BSD	(*02)(*04)(*72)
libdb-utils	BSD and LGPLv2 and Sleepycat	(*03)(*72)
libmount	LGPLv2+	(*03)
libgnome-keyring	GPLv2+ and LGPLv2+	(*01)(*03)
libselinux-python	Public Domain	–
libtirpc	SISSL and BSD	(*72)
libssh2	BSD	(*36)
man-pages	GPL+ and GPLv2+ and BSD and MIT and Copyright only and IEEE	(*39)
mariadb-libs	GPLv2 with exceptions and LGPLv2 and BSD	(*03)(*01)(*73)
mailx	BSD with advertising and MPLv1.1	(*38)
make	GPLv2+	(*01)
microcode_ctl	GPLv2+ and Redistributable, no modification permitted	(*01)
m4	GPLv3+	(*03)
mpfr	LGPLv3+ and GPLv3+ and GFDL	(*76)(*02)(*04)
man-db	GPLv2+ and GPLv3+	(*01)(*02)
mozjs17	GPLv2+ or LGPLv2+ or MPLv1.1	(*15)(*01)(*03)
man-pages-ja	Freely redistributable without restriction	–
ntpdate	(MIT and BSD and BSD with advertising) and GPLv2	(*73)(*01)
neon	LGPLv2+	(*03)
net-tools	GPLv2+	(*01)
nsawb	GPLv2	(*01)
nspr	MPLv2.0	(*16)
nss	MPLv2.0	(*16)
numactl-libs	LGPLv2 and GPLv2	(*01)(*03)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
ncurses-libs	MIT	(*73)
newt	LGPLv2	(*03)
nss-util	MPLv2.0	(*16)
nss-softokn-freebl	MPLv2.0	(*16)
nss-pem	MPLv1.1	(*15)
nss-sysinit	MPLv2.0	(*16)
ncurses-base	MIT	(*73)
ntp	(MIT and BSD and BSD with advertising) and GPLv2	(*01)(*73)
nss-softokn	MPLv2.0	(*16)
nettle	LGPLv2+	(*03)
nss-softokn-freebl	MPLv2.0	(*16)
ncurses	MIT	(*73)
newt-python	LGPLv2	(*03)
nss-tools	MPLv2.0	(*16)
openssh-server	BSD	(*72)
openssh	BSD	(*72)
os-prober	GPLv2+ and GPL+	(*01)
openssl	OpenSSL	(*18)
openldap	OpenLDAP	(*17)
openssh-clients	BSD	(*72)
openssl-libs	OpenSSL	(*18)
pygpgme	LGPLv2+	(*03)
python-urlgrabber	LGPLv2+	(*03)
python-configobj	BSD	(*72)
pam	BSD and GPLv2+	(*72)(*01)
python	Python	(*19)
psmisc	GPLv2+	(*01)
perl-Pod-Simple	GPL+ or Artistic	(*02)
perl-Carp	GPL+ or Artistic	(*02)
perl-Time-Local	GPL+ or Artistic	(*02)
parted	GPLv3+	(*03)
perl-Filter	GPL+ or Artistic	(*02)
perl-libs	GPL+ or Artistic	(*02)
perl-Pod-Escapes	GPL+ or Artistic	(*02)
postfix	IBM and GPLv2+	(*02)
perl-parent	GPL+ or Artistic	(*02)
plymouth-scripts	GPLv2+	(*02)
pcre	BSD	(*45)
pth	LGPLv2+	(*03)
python-pycurl	LGPLv2+ or MIT	(*03)(*73)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
python-perf	GPLv2	(*01)
python-firewall	GPLv2+	(*01)
python-slip	GPLv2+	(*01)
python-iniparse	MIT	(*73)
pyliblzma	LGPLv3+	(*04)
python-linux-procfs	GPLv2	(*01)
plymouth-core-libs	GPLv2+	(*01)
pkgconfig	GPLv2+	(*01)
python-decorator	BSD	(*72)
python-libs	Python	(*19)
pinentry	GPLv2+	(*01)
perl-TermReadKey	(Copyright only) and (Artistic or GPL+)	(*02)
perl-XML-Parser	GPL+ or Artistic	(*02)
perl-Data-Dumper	GPL+ or Artistic	(*02)
p11-kit-trust	BSD	(*42)
patch	GPLv3+	(*03)
patchutils	GPLv2+	(*01)
perl-Error	(GPL+ or Artistic) and MIT	(*02)(*73)
pciutils-libs	GPLv2+	(*01)
perl-Getopt-Long	GPLv2+ or Artistic	(*02)
perl-PathTools	(GPL+ or Artistic) and BSD	(*02)(*72)
perl-File-Temp	GPL+ or Artistic	(*02)
perl-Scalar-List-Utils	GPL+ or Artistic	(*02)
perl-constant	GPL+ or Artistic	(*02)
perl-Socket	GPL+ or Artistic	(*02)
perl-Exporter	GPL+ or Artistic	(*02)
perl-Storable	GPL+ or Artistic	(*02)
perl-macros	GPL+ or Artistic	(*02)
perl-Pod-Usage	GPL+ or Artistic	(*02)
perl-Text-ParseWords	GPL+ or Artistic	(*02)
perl-Pod-Perldoc	GPL+ or Artistic	(*02)
perl-HTTP-Tiny	GPL+ or Artistic	(*02)
pciutils	GPLv2+	(*01)
p11-kit	BSD	(*42)
perl-Git	GPLv2	(*01)
plymouth	GPLv2+	(*01)
polkit	LGPLv2+	(*03)
popt	MIT	(*73)
perl-srpm-macros	GPLv3+	(*03)
pakchois	LGPLv2+	(*03)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
python-schedutils	GPLv2	(*01)
python-slip-dbus	GPLv2+	(*01)
pyxattr	LGPLv2+	(*03)
python-gobject-base	LGPLv2+ and MIT	(*03)(*73)
python-pyudev	LGPLv2+	(*03)
procps-ng	GPL+ and GPLv2 and GPLv2+ and GPLv3+ and LGPLv2+	(*78)
perl-Test-Harness	GPL+ or Artistic	(*02)
perl-Thread-Queue	GPL+ or Artistic	(*02)
perl	(GPL+ or Artistic) and (GPLv2+ or Artistic) and Copyright Only and MIT and Public Domain and UCD	(*02)(*04)(*73)
passwd	BSD or GPL+	(*02)
perl-File-Path	GPL+ or Artistic	(*02)
perl-threads-shared	GPL+ or Artistic	(*02)
perl-Time-HiRes	GPL+ or Artistic	(*02)
perl-threads	GPL+ or Artistic	(*02)
perl-Encode	GPL+ or Artistic	(*02)
perl-podlators	GPL+ or Artistic	(*02)
polkit-pkla-compat	LGPLv2+	(*03)
policycoreutils	GPLv2	(*01)
qrencode-libs	LGPLv2+	(*03)
rcs	GPLv3+	(*03)
readline	GPLv3+	(*03)
rootfiles	Public Domain	–
redhat-rpm-config	GPL+	(*02)
rpm-python	GPLv2+	(*02)
rpcbind	BSD	(*72)
rpm-sign	GPLv2+	(*01)
rpm-libs	GPLv2+ and LGPLv2+ with exceptions	(*01)
rpm-build-libs	GPLv2+ and LGPLv2+ with exceptions	(*01)
rsync	GPLv3+	(*03)
rsyslog	(GPLv3+ and ASL 2.0)	(*04)(*06)
rpm-build	GPLv2+	(*01)
rpm	GPLv2+	(*01)
setserial	GPL+	(*02)
setup	Public Domain	(*02)
slang	GPLv2+	(*01)
sudo	ISC	(*12)
systemd-sysv	LGPLv2+	(*03)
sqlite	Public Domain	–
selinux-policy-targeted	GPLv2+	(*01)
systemtap-client	GPLv2+	(*01)

表B-1　パッケージ名とソフトウェアライセンス（続き）

Package Name	License	License Reference No.
systemd	LGPLv2+ and MIT and GPLv2+	(*01)(*03)
sed	GPLv3+	(*03)
smartmontools	GPLv2+	(*01)
systemtap	GPLv2+	(*01)
shared-mime-info	GPLv2+	(*01)
shadow-utils	BSD and GPLv2+	(*72)(*01)
selinux-policy	GPLv2+	(*01)
systemtap-runtime	GPLv2+	(*01)
subversion-libs	ASL 2.0	(*06)
subversion	ASL 2.0	(*06)
sudo	ISC	(*12)
swig	GPLv3+ and BSD	(*48)
snappy	BSD	(*47)
systemd-libs	LGPLv2+ and MIT	(*03)(*73)
systemtap-devel	GPLv2+	(*01)
systemtap	GPLv2+	(*01)
sysvinit-tools	GPLv2+	(*01)
tcp_wrappers-libs	BSD	(*49)
telnet	BSD	(*72)
tzdata	Public Domain	–
trousers	BSD	(*60)
tar	GPLv3+	(*03)
tuned	GPLv2+	(*01)
tcpdump	BSD with advertising	(*50)
TinyXML 2.6.2		(*27)
unzip	BSD	(*61)
ustr	MIT or LGPLv2+ or BSD	(*73)(*03)(*72)
util-linux	GPLv2 and GPLv2+ and LGPLv2+ and BSD with advertising and Public Domain	(*01)(*03)(*72)
vsftpd	GPLv2 with exceptions	(*74)
virt-what	GPLv2+	(*01)
vim-minimal	Vim	–
VXI-11 1.12		(*28)
which	GPLv3	(*03)
xz-libs	LGPLv2+	(*01)(*02)(*03)
xz	LGPLv2+	(*01)(*02)(*03)
xfsprogs	GPL+ and LGPLv2+	(*01)(*02)
yum-metadata-parser	GPLv2	(*01)
yum-plugin-fastestmirror	GPLv2+	(*01)
yum	GPLv2+	(*01)
zip	zlib and Boost	(*71)
zlib	BSD	(*72)

MU887002A の性能試験を行う際に，試験結果をまとめるための用紙例です。性能試験の際に，本章を複写して利用してください。

C.1　試験環境 C-2
C.2　VSWR C-3
C.3　SGレベル確度 C-4
C.4　SGスプリアス C-11
C.5　SAレベル確度 C-13
C.6　SAリニアリティ C-17
C.7　パスロス測定確度 (MX887092A) C-20

C.1 試験環境

試験場所 _____

_____

_____

_____

_____

レポート NO. _____

日付 _____

テスト担当者 _____

周囲温度 _____ ℃

相対湿度 _____ %

機器名 : _____

製造 NO. _____

オプション _____

機器名 : _____

製造 NO. _____

オプション _____

機器名 : _____

製造 NO. _____

オプション _____

機器名 : _____

製造 NO. _____

オプション _____

機器名 : _____

製造 NO. _____

オプション _____

特記事項 _____

_____

_____

C.2 VSWR

VSWR の性能試験記録表は，オプション，Port により変わります。
使用する場合は，測定する TRX とポートの番号を記入してください。

ポート，測定周波数別に使用する性能試験記録表を次の表に示します。

表C.2-1 測定条件と記録表の番号

測定条件	表番号
MU887002A-007 非搭載時 TRX 1/2 Port 1～12 MU887002A-007 搭載時 TRX 1/2 Port 1～4	表C.2-2
MU887002A-007 搭載時 TRX 1/2 Port 5～12	表C.2-3

表C.2-2 VSWR

測定ポート: TRX Port

Frequency (MHz) f	Worst VSWR Frequency (MHz)	Worst VSWR	Spec.
400≦f＜450			＜1.4
450≦f≦2700			＜1.3
2700＜f≦3800			＜1.4
3800＜f≦6000			＜1.4

表C.2-3 VSWR

測定ポート: TRX Port

Frequency (MHz) f	Worst VSWR Frequency (MHz)	Worst VSWR	Spec.
400≦f＜450*1			＜1.4
450≦f≦2700*1			＜1.3
2700＜f≦3800*1			＜1.4
3800＜f＜6000*1			＜1.4
5800≦f≦7400*2			＜1.7

＊1: 設定周波数＜5900 MHz
＊2: 設定周波数≧5900 MHz

C.3 SGレベル確度

SG レベル確度の性能試験記録表は，オプション，Port，周波数により変わります。使用する場合は，測定する TRX とポートの番号を記入してください。

ポート，測定周波数別に使用する性能試験記録表を次の表に示します。

表C.3-1　測定条件と記録表の番号
MU887002A-007 非搭載時 TRX 1/2 Port 1～12
MU887002A-007 搭載時 TRX 1/2 Port 1～4

周波数*1 (MHz)	MU887002A 設定周波数*2 (MHz)	記録表
400	400.001	表C.3-3
1318	1318.001	表C.3-3
2236	2236.001	表C.3-3
3154	3154.001	表C.3-3
3800	3800.001	表C.3-4
4072	4072.001	表C.3-4
4990	4990.001	表C.3-4
5908	5908.001	表C.3-4
6000	5999.999	表C.3-4

表C.3-2　測定条件と記録表の番号
MU887002A-007 搭載時 TRX 1/2 Port 5～12

周波数*1 (MHz)	MU887002A 設定周波数*2 (MHz)	記録表
400	400.001	表C.3-3
1318	1318.001	表C.3-3
2236	2236.001	表C.3-3
3154	3154.001	表C.3-3
3800	3800.001	表C.3-4
4072	4072.001	表C.3-4
4990	4990.001	表C.3-4
5900	5900.001	表C.3-5
6818	6818.001	表C.3-5
7300	7299.999	表C.3-5

＊1: 性能試験記録表で左端の列（Frequency）に記入する周波数です。
＊2: 性能試験記録表で左から 2 番目の列（MU887002A Settings - Frequency）に記入する周波数です。

表C.3-3　SG レベル確度

測定ポート：TRX　　Port　　Freq.≦3800 MHz

Frequency (MHz)	MU887002A Settings		Power Meter	Signal Analyzer					MU887002A	
	Frequency (MHz)	Output Level (dBm)	Level (dBm)	Setting	Center Frequency	#x⇒#1 offset (dB)	Level (dBm)	Converted Level to #1(dBm)	Accuracy (dB)	Spec. (dB)
	= F0	–5 = P1		#1	F0		= B1		= (B1 - A + C) - P1	±1.0
		–10 = P2					= B2		= (B2 - A + C) - P2	±1.0
		–15 = P3					= B3		= (B3 - A + C) - P3	±1.0
		–20 = P4					= B4		= (B4 - A + C) - P4	±1.0
		–22 = P5	= C				= A = B5		= C - P5	±1.0
		–25 = P6					= B6		= (B6 - A + C) - P6	±1.0
		–30 = P7					= B7		= (B7 - A + C) - P7	±1.0
		–35 = P8					= B8		= (B8 - A + C) - P8	±1.0
		–40 = P9					= B9		= (B9 - A + C) - P9	±1.0
		–45 = P10					= B10		= (B10 - A + C) - P10	±1.0
		–50 = P11		#2		= D	= G11	= B11 = G11 + D	= (B11 - A + C) - P11	±1.0

表C.3-3 SG レベル確度 (続き)

Frequency (MHz)	MU887002A Settings		Power Meter	Signal Analyzer					MU887002A	
	Frequency (MHz)	Output Level (dBm)	Level (dBm)	Setting	Center Frequency	#x⇒#1 offset (dB)	Level (dBm)	Converted Level to #1(dBm)	Accuracy (dB)	Spec. (dB)
	= F0	–55 = P12		#2	F0	= D	= G12	= B12 = G12 + D	= (B12 - A + C) - P12	±1.0
		–60 = P13					= G13	= B13 = G13 + D	= (B13 - A + C) - P13	±1.0
		–65 = P14					= G14	= B14 = G14 + D	= (B14 - A + C) - P14	±1.0
		–70 = P15					= G15	= B15 = G15 + D	= (B15 - A + C) - P15	±1.0
		–75 = P16					= G16	= B16 = G16 + D	= (B16 - A + C) - P16	±1.0
		–80 = P17		#3			= G17	= B17 = G17 + D	= (B17 - A + C) - P17	±1.0
		–85 = P18					= G18	= B18 = G18 + D	= (B18 - A + C) - P18	±1.0
		–90 = P19					= G19	= B19 = G19 + D	= (B19 - A + C) - P19	±1.0
		–95 = P20					= G20	= B20 = G20 + D	= (B20 - A + C) - P20	±1.0
		–100 = P21					= G21	= B21 = G21 + D	= (B21 - A + C) - P21	±1.0
		–105 = P22		#4	F0 + 100 Hz		= G22	= B22 = G22 + D	= (B22 - A + C) - P22	±1.0
		–110 = P23					= G23	= B23 = G23 + D	= (B23 - A + C) - P23	±1.0
		–115 = P24		#5			= G24	= B24 = G24 + D	= (B24 - A + C) - P24	±1.0
		–120 = P25					= G25	= B25 = G25 + D	= (B25 - A + C) - P25	±1.0

表C.3-4　SG レベル確度

測定ポート：TRX　　Port　　3800 MHz＜Freq.

Frequency (MHz)	MU887002A Settings		Power Meter	Signal Analyzer					MU887002A	
	Frequency (MHz)	Output Level (dBm)	Level (dBm)	Setting	Center Frequency	#x⇒#1 offset (dB)	Level (dBm)	Converted Level to #1(dBm)	Accuracy (dB)	Spec. (dB)
	= F0	–8 = P1		#1	F0		= B1		= (B1 - A + C) - P1	±1.3
		–10 = P2					= B2		= (B2 - A + C) - P2	±1.3
		–15 = P3					= B3		= (B3 - A + C) - P3	±1.3
		–20 = P4					= B4		= (B4 - A + C) - P4	±1.3
		–22 = P5	= C				= A = B5		= C - P5	±1.3
		–25 = P6					= B6		= (B6 - A + C) - P6	±1.3
		–30 = P7					= B7		= (B7 - A + C) - P7	±1.3
		–35 = P8					= B8		= (B8 - A + C) - P8	±1.3
		–40 = P9					= B9		= (B9 - A + C) - P9	±1.3
		–45 = P10					= B10		= (B10 - A + C) - P10	±1.3
		–50 = P11		#2		= D	= G11	= B11 = G11 + D	= (B11 - A + C) - P11	±1.3

表C.3-4 SG レベル確度 (続き)

Frequency (MHz)	MU887002A Settings		Power Meter	Signal Analyzer					MU887002A	
	Frequency (MHz)	Output Level (dBm)	Level (dBm)	Setting	Center Frequency	#x⇒#1 offset (dB)	Level (dBm)	Converted Level to #1(dBm)	Accuracy (dB)	Spec. (dB)
		–55 = P12		#2	F0		= G12	= B12 = G12 + D	= (B12 - A + C) - P12	±1.3
		–60 = P13					= G13	= B13 = G13 + D	= (B13 - A + C) - P13	±1.3
		–65 = P14					= G14	= B14 = G14 + D	= (B14 - A + C) - P14	±1.3
		–70 = P15					= G15	= B15 = G15 + D	= (B15 - A + C) - P15	±1.3
		–75 = P16					= G16	= B16 = G16 + D	= (B16 - A + C) - P16	±1.3
		–80 = P17		#3			= G17	= B17 = G17 + D	= (B17 - A + C) - P17	±1.3
		–85 = P18					= G18	= B18 = G18 + D	= (B18 - A + C) - P18	±1.3
		–90 = P19					= G19	= B19 = G19 + D	= (B19 - A + C) - P19	±1.3
		–95 = P20					= G20	= B20 = G20 + D	= (B20 - A + C) - P20	±1.3
		–100 = P21					= G21	= B21 = G21 + D	= (B21 - A + C) - P21	±1.3

表C.3-5　SG レベル確度

測定ポート：TRX　　Port　　5900 MHz≦Freq.

Frequency (MHz)	MU887002A Settings		Power Meter	Signal Analyzer					MU887002A	
	Frequency (MHz)	Output Level (dBm)	Level (dBm)	Setting	Center Frequency	#x⇒#1 offset (dB)	Level (dBm)	Converted Level to #1(dBm)	Accuracy (dB)	Spec. (dB)
	= F0	–10 = P1		#1	F0		B1		= (B1 - A + C) - P1	±1.3
		–15 = P2					= B2		= (B2 - A + C) - P2	±1.3
		–20 = P3					= B3		= (B3 - A + C) - P3	±1.3
		–22 = P4	= C				= A = B4		= C - P4	±1.3
		–25 = P5					= B5		= (B5 - A + C) - P5	±1.3
		–30 = P6					= B6		= (B6 - A + C) - P6	±1.3
		–35 = P7					= B7		= (B7 - A + C) - P7	±1.3
		–40 = P8					= B8		= (B8 - A + C) - P8	±1.3
		–45 = P9					= B9		= (B9 - A + C) - P9	±1.3
		–50 = P10		#2		= D	= G10	= B10 = G10 + D	= (B10 - A + C) - P10	±1.3

表C.3-5 SG レベル確度 (続き)

Frequency (MHz)	MU887002A Settings		Power Meter	Signal Analyzer					MU887002A	
	Frequency (MHz)	Output Level (dBm)	Level (dBm)	Setting	Center Frequency	#x⇒#1 offset (dB)	Level (dBm)	Converted Level to #1(dBm)	Accuracy (dB)	Spec. (dB)
		–55 = P11		#2	F0		= G11	= B11 = G11 + D	= (B11 - A + C) - P11	±1.3
		–60 = P12					= G12	= B12 = G12 + D	= (B12 - A + C) - P12	±1.3
		–65 = P13					= G13	= B13 = G13 + D	= (B13 - A + C) - P13	±1.3
		–70 = P14					= G14	= B14 = G14 + D	= (B14 - A + C) - P14	±1.3
		–75 = P15					= G15	= B15 = G15 + D	= (B15 - A + C) - P15	±1.3
		–80 = P16		#3			= G16	= B16 = G16 + D	= (B16 - A + C) - P16	±1.3
		–85 = P17					= G17	= B17 = G17 + D	= (B17 - A + C) - P17	±1.3
		–90 = P18					= G18	= B18 = G18 + D	= (B18 - A + C) - P18	±1.3
		–95 = P19					= G19	= B19 = G19 + D	= (B19 - A + C) - P19	±1.3
		–100 = P20					= G20	= B20 = G20 + D	= (B20 - A + C) - P20	±1.3

C.4 SGスプリアス

SG スプリアスの性能試験記録表は，オプション，Port により変わります。
使用する場合は，測定する TRX とポートの番号を記入してください。

Output Level に記入する値は，–5 dBm です。

表C.4-1　SG スプリアス
MU887002A-007 非搭載時 TRX 1/2 Port 1～12
MU887002A-007 搭載時 TRX 1/2 Port 1～4

測定ポート：TRX　　Port

MU887002A Settings		Signal Analyzer					Spec. (dBc)
Output Level (dBm)	Frequency (MHz)	Fundamental Level (dBm) A	HD2 Level (dBm) B	HD3 Level (dBm) C	HD2 (dBc) = B – A	HD3 (dBc) = C – A	
	400						< –25
	600						< –25
	850						< –25
	900						< –25
	1000						< –25
	1500						< –25
	1800						< –25
	1950						< –25
	2200						< –25
	2450						< –25
	2650						< –25
	3000						< –25

表C.4-2　SG スプリアス
MU887002A-007 搭載時 TRX 1/2 Port 5～12

測定ポート：TRX　　Port

MU887002A Settings		Signal Analyzer					Spec. (dBc)
Output Level (dBm)	Frequency (MHz)	Fundamental Level (dBm) A	HD2 Level (dBm) B	HD3 Level (dBm) C	HD2 (dBc) = B – A	HD3 (dBc) = C – A	
	400						< –25
	600						< –25
	850						< –25
	900						< –25
	1000						< –25
	1500						< –25
	1800						< –25
	1950						< –25
	2200						< –25
	2450						< –25
	2650						< –25
	3000						< –25
	3300						< –25
	3650						< –25

C.5 SAレベル確度

SA レベル確度の性能試験記録表は，オプション，Port，周波数により変わります。使用する場合は，測定する TRX とポートの番号を記入してください。

ポート，測定周波数別に使用する性能試験記録表を次の表に示します。

表C.5-1　測定条件と記録表の番号
MU887002A-007 非搭載時 TRX 1/2 Port 1～12
MU887002A-007 搭載時 TRX 1/2 Port 1～4

MU887002A 設定周波数（MHz）	信号発生器 設定周波数（MHz）	記録表
1318	1317.9	表C.5-3
2236	2235.9	表C.5-3
3154	3153.9	表C.5-3
3800	3799.9	表C.5-3
4072	4071.9	表C.5-4
4990	4989.9	表C.5-4
6000	5999.9	表C.5-4

表C.5-2　測定条件と記録表の番号
MU887002A-007 搭載時 TRX 1/2 Port 5～12

MU887002A 設定周波数（MHz）	信号発生器 設定周波数（MHz）	記録表
1318	1317.9	表C.5-3
2236	2235.9	表C.5-3
3154	3153.9	表C.5-3
3800	3799.9	表C.5-3
4072	4071.9	表C.5-4
4990	4989.9	表C.5-4
5900	5900.1	表C.5-5
6818	6817.9	表C.5-5
7300	7299.9	表C.5-5

表C.5-3　SA レベル確度 (400≦f≦3800 MHz)

測定ポート：TRX　　Port

MU887002A Settings		Signal Generator Settings		Power Meter	Signal Analyzer	MU887002A		
Frequency (MHz)	Input Level (dBm)	Frequency (MHz)	Level (dBm)	Level (dBm)	Level (dBm)	Level (dBm)	Accuracy (dB)	Spec. (dB)
	+35		= P0	= C0 = approx. 0 dBm		= A1	= A1 - C0	±0.5
	+30					= A2	= A2 - C0	±0.5
	+25					= A3	= A3 - C0	±0.5
	+20					= A4	= A4 - C0	±0.5
	+15					= A5	= A5 - C0	±0.5
	+10					= A6	= A6 - C0	±0.5
	+5					= A7	= A7 - C0	±0.5
	0					= A8	= A8 - C0	±0.5
	–5		= P30	= C30 = approx. -30 dBm		= A9	= A9 - C30	±0.5
	–10					= A10	= A10 - C30	±0.5
	–15					= A11	= A11 - C30	±0.5
	–20					= A12	= A12 - C30	±0.5
	–25					= A13	= A13 - C30	±0.5
	–30					= A14	= A14 - C30	±0.5
			= P25	= C25 = approx. -25 dBm	= S			
	–35		P25 – 10		= S1	= A15	= A15 - (C25 + S1 - S)	±0.7
	–40					= A16	= A16 - (C25 + S1 - S)	±0.7
	–45					= A17	= A17 - (C25 + S1 - S)	±0.7
	–50					= A18	= A18 - (C25 + S1 - S)	±0.7
	–55					= A19	= A19 - (C25 + S1 - S)	±0.7
	–60					= A20	= A20 - (C25 + S1 - S)	±0.9
	–65					= A21	= A21 - (C25 + S1 - S)	±0.9

表C.5-4 SA レベル確度 (3800＜f≦6000 MHz)

測定ポート: TRX Port

MU887002A Settings		Signal Generator Settings		Power Meter	Signal Analyzer	MU887002A		
Frequency (MHz)	Input Level (dBm)	Frequency (MHz)	Level (dBm)	Level (dBm)	Level (dBm)	Level (dBm)	Accuracy (dB)	Spec (dB)
	+35					= A1	=A1 - C0	±0.7
	+30					= A2	=A2 - C0	±0.7
	+25					= A3	=A3 - C0	±0.7
	+20					= A4	=A4 - C0	±0.7
	+15					= A5	=A5 - C0	±0.7
	+10					= A6	=A6 - C0	±0.7
	+5					= A7	=A7 - C0	±0.7
	0		= P0	= C0 = approx. 0 dBm		= A8	=A8 - C0	±0.7
	−5					= A9	=A9 - C30	±0.7
	−10					= A10	=A10 - C30	±0.7
	−15					= A11	=A11- C30	±0.7
	−20					= A12	=A12 - C30	±0.7
	−25					= A13	=A13 - C30	±0.7
	−30		= P30	= C30 = approx. -30 dBm		= A14	=A14 - C30	±0.7
			= P25	= C25 = approx. -25 dBm	=S			
	−35		P25 – 10		= S1	= A15	= A15 - (C25 + S1 - S)	±0.9
	−40					= A16	= A16 - (C25 + S1 - S)	±0.9
	−45					= A17	= A17 - (C25 + S1 - S)	±0.9
	−50					= A18	= A18 - (C25 + S1 - S)	±0.9
	−55					= A19	= A19 - (C25 + S1 - S)	±0.9
	−60					= A20	= A20 - (C25 + S1 - S)	±1.1
	−65					= A21	= A21 - (C25 + S1 - S)	±1.1

表C.5-5　SA レベル確度 (5900≦f≦7300 MHz)

測定ポート：TRX　　Port

MU887002A Settings		Signal Generator Settings		Power Meter	Signal Analyzer	MU887002A		
Frequency (MHz)	Input Level (dBm)	Frequency (MHz)	Level (dBm)	Level (dBm)	Level (dBm)	Level (dBm)	Accuracy (dB)	Spec. (dB)
	+30					= A1	= A1 - C0	±1.0
	+25					= A2	= A2 - C0	±1.0
	+20					= A3	= A3 - C0	±1.0
	+15					= A4	= A4 - C0	±1.0
	+10					= A5	= A5 - C0	±1.0
	+5					= A6	= A6 - C0	±1.0
	0		= P0	= C0 = approx. 0 dBm		= A7	= A7 - C0	±1.0
	–5					= A8	= A8 - C30	±1.0
	–10					= A9	= A9 - C30	±1.0
	–15					= A10	= A10 - C30	±1.0
	–20					= A11	= A11 - C30	±1.0
	–25					= A12	= A12 - C30	±1.0
	–30		= P30	= C30 = approx. -30 dBm		= A13	= A13 - C30	±1.0
			= P25	= C25 = approx. -25 dBm	= S			
	–35		P25 – 10			= A14	= A14 - (C25 + S1 - S)	±1.2
	–40					= A15	= A15 - (C25 + S1 - S)	±1.2
	–45					= A16	= A16 - (C25 + S1 - S)	±1.2
	–50					= A17	= A17 - (C25 + S1 - S)	±1.2
	–55					= A18	= A18 - (C25 + S1 - S)	±1.2
	–60					= A19	= A19 - (C25 + S1 - S)	±1.8
	–65				= S1	= A20	= A20 - (C25 + S1 - S)	±1.8

C.6 SAリニアリティ

SA リニアリティの性能試験記録表は，オプション，Port，周波数により変わります。使用する場合は，測定する TRX とポートの番号を記入してください。

ポート，測定周波数別に使用する性能試験記録表を次の表に示します。

表C.6-1　測定条件と記録表の番号
MU887002A-007 非搭載時 TRX 1/2 Port 1～12
MU887002A-007 搭載時 TRX 1/2 Port 1～4

MU887002A 受信周波数（MHz）	信号発生器 設定周波数（MHz）	入力レベル (dBm)	記録表
1318	1317.9	0	表C.6-3
2236	2235.9	0	表C.6-3
3154	3153.9	0	表C.6-3
4072	4071.9	0	表C.6-3
4990	4989.9	0	表C.6-3
6000	5999.9	0	表C.6-3
1318	1317.9	–25	表C.6-4
2236	2235.9	–25	表C.6-4
3154	3153.9	–25	表C.6-4
4072	4071.9	–25	表C.6-4
4990	4989.9	–25	表C.6-4
6000	5999.9	–25	表C.6-4

表C.6-2　測定条件と記録表の番号
MU887002A-007 搭載時 TRX 1/2 Port 5～12

MU887002A 受信周波数 (MHz)	信号発生器 設定周波数 (MHz)	入力レベル (dBm)	記録表
1318	1317.9	0	表C.6-3
2236	2235.9	0	表C.6-3
3154	3153.9	0	表C.6-3
4072	4071.9	0	表C.6-3
4990	4989.9	0	表C.6-3
5900	5900.1	0	表C.6-3
6818	6817.9	0	表C.6-3
7300	7299.9	0	表C.6-3
1318	1317.9	–25	表C.6-4
2236	2235.9	–25	表C.6-4
3154	3153.9	–25	表C.6-4
4072	4071.9	–25	表C.6-4
4990	4989.9	–25	表C.6-4
5900	5900.1	–25	表C.6-4
6818	6817.9	–25	表C.6-4
7300	7299.9	–25	表C.6-4

表C.6-3　SA リニアリティ (0 dBm)

測定ポート: TRX　　Port

MU887002A Settings		Signal Generator Settings		Power Meter	Signal Analyzer	MU887002A		
Frequency (MHz)	Input Level (dBm)	Frequency (MHz)	Level (dBm)	Level (dBm)	Level (dBm)	Level (dBm)	Accuracy (dB)	Spec. (dB)
	0		= P0	=C0 = approx. 0 dBm	= S	= A	–	–
			P0 – 10		= S1	= B1	= (B1 - A) - (S1 - S)	±0.2
			P0 – 20		= S2	= B2	= (B2 - A) - (S2 - S)	±0.2
			P0 – 30		= S3	= B3	= (B3 - A) - (S3 - S)	±0.2
			P0 – 40		= S4	= B4	= (B4 - A) - (S4 - S)	±0.2

表C.6-4　SA リニアリティ (–25 dBm)

測定ポート: TRX　　Port

MU887002A Settings		Signal Generator Settings		Power Meter	Signal Analyzer	MU887002A		
Frequency (MHz)	Input Level (dBm)	Frequency (MHz)	Level (dBm)	Level (dBm)	Level (dBm)	Level (dBm)	Accuracy (dB)	Spec. (dB)
	–25		= P25	= C25 = approx. –25 dBm	= S	= A	–	–
			P25 – 10		= S1	= B1	= (B1 - A) - (S1 - S)	±0.2
			P25 – 20		= S2	= B2	= (B2 - A) - (S2 - S)	±0.2
			P25 – 30		= S3	= B3	= (B3 - A) - (S3 - S)	±0.2
			P25 – 40		= S4	= B4	= (B4 - A) - (S4 - S)	±0.4

C.7 パスロス測定確度 (MX887092A)

表C.7-1　パスロス測定確度 (MX887092A)

測定ポート：TRX　　Port 11

周波数 (MHz)	Accuracy (dB)	Spec. (dB)
400		±0.5
1852.5		±0.5
2700		±0.5
3300		±0.9
4920		±0.9
6000		±0.9

ここでは，ランプの状態表示について説明します。ランプの位置は，「図 2.1-1 正面パネル」を参照してください。

表D-1　ランプの状態表示

名称	表示の説明
Status	赤点灯：自己診断エラー，または温度異常が発生 赤点滅：レベル校正の処理でエラーが発生 橙点灯：電源投入時の起動処理中 橙点滅：予期しないエラーが発生*1 緑点灯：正常動作*2 緑点滅：電源投入時の起動処理中，電源切断時の終了処理中，Calibration 実行中
Remote	緑点灯：リモート制御中 緑点滅：リモート制御によるエラー発生*3 消灯：　未制御状態
1	橙点滅：ソフトウェア更新中*1 消灯：　異常なし
2 (VSG 用途)	赤点灯：波形ファイルの読み込みエラーが発生 緑点灯：RF 信号出力オン 橙点灯：ケーブルの損失補正エラーが発生 (RF 出力信号が設定したレベルより低くなった) 橙点滅：ソフトウェア更新中*1，波形ファイル読み込み中 消灯：　RF 信号出力オフ
3 (VSA 用途)	赤点灯：アプリケーションソフトウェアによる測定・解析でエラーが発生*4 緑点灯：アプリケーションソフトウェアによる測定・解析を実行中*4 橙点滅：ソフトウェア更新中*1 消灯：　異常なし

＊1: モジュールをリモート制御することができません。

＊2: 電源投入後に緑色が点灯しても 1 分程はリモート制御ができないことがあります。リモート制御が可能になるとき，MT8870A のエラーランプ（Error）が赤点滅から消灯となります。ブザー設定がオンであれば，「ピ，ピ，ピ」と音が鳴ります。

＊3: ブザー設定がオンであれば，点灯時に「ピ」と音が鳴ります。
次のコマンドでエラーの情報を取得できます。
:SYSTem:ERRor?，SYSERR?

＊4: アプリケーションソフトウェアの形名は，「表 1.3.3-1 アプリケーションソフトウェア」を参照してください。

注:

- 電源投入時はすべてのランプが橙色に点灯します。ただし，TRX 2 側のランプ 3 は緑色に点灯する場合もありますが，異常ではありません。
- 電源切断時にすべてのランプが橙色に点灯します。橙色が消灯後，赤色に点灯するランプもありますが，異常ではありません。

エラー発生時の処置

Status ランプが赤点灯，橙点滅，緑点滅した場合は，電源を一度切断して電源を再投入してください。現象が改善しない場合は，本書（紙版説明書では巻末，電子版説明書では別ファイル）に記載の「本製品についてのお問い合わせ窓口」へすみやかにご連絡ください。

ランプ 2 が赤点灯した場合は，次のコマンドで波形ファイルを読み込めなかった原因を調べます。
SCPI コマンドモードの場合

```
:SOURce:GPRF:GENerator:ARB:FILE:LOAD? <file_name>
```

Native コマンドモードの場合

```
SOUR:GPRF:GEN:ARB:FILE:LOAD? <file_name>
```

ランプ 3 が赤点灯した場合は，アプリケーションソフトウェアのコマンドで原因を調べます。使用するコマンドを次の表に示します。コマンドの説明は，アプリケーションソフトウェアの取扱説明書を参照してください。

表D-2　エラー調査に使用するコマンド

アプリケーションソフトウェアの形名	コマンド
MX887010A, MX887011A, MX887012A, MX887013A, MX887014A, MX887015A, MX887016A, MX887017A, MX887018A, MX887019A, MX887065A, MX887067A, MX887068A	MSTAT? :FETCh:CELLular:MEASurement:STATe?
MX887030A, MX887031A, MX887032A, MX887033A, MX887040A, MX887050A	STAT:SRW:MEAS? :STATus:SRWireless:MEASurement?
MX887060A	STAT:LRWP:MEAS? :STATus:LRWPan:MEASurement?
MX887061A	STAT:ZWAV:MEAS? :STATus:ZWAVe:MEASurement?

MX887092A パスロス測定機能について説明します。

E.1 概要 E-2
E.1.1 概要 E-2
E.1.2 構成 E-3
E.1.3 規格 E-3
E.2 ライセンス登録/確認 E-5
E.3 パスロス測定 E-6
E.3.1 測定 E-7
E.3.2 パスロス値の設定 E-9
E.3.3 使用例 E-10
E.4 SCPIコマンド E-12
E.4.1 コマンド一覧 E-13
E.4.2 コマンドの詳細 E-15
E.5 Nativeコマンド E-29
E.5.1 コマンド一覧 E-30

E.1　概要

E.1.1　概要

機能

MX887092A パスロス測定機能（Pathloss Measurement Function）は DUT 接続端に Open, Short デバイスを接続したときのそれぞれの反射信号を測定することで MU887002A のテストポート端から DUT 接続端までのパスロス値を測定する機能を提供します。

測定したパスロス値は,「3.5.3 ケーブルの損失補正」の損失補正データとして使用できます。

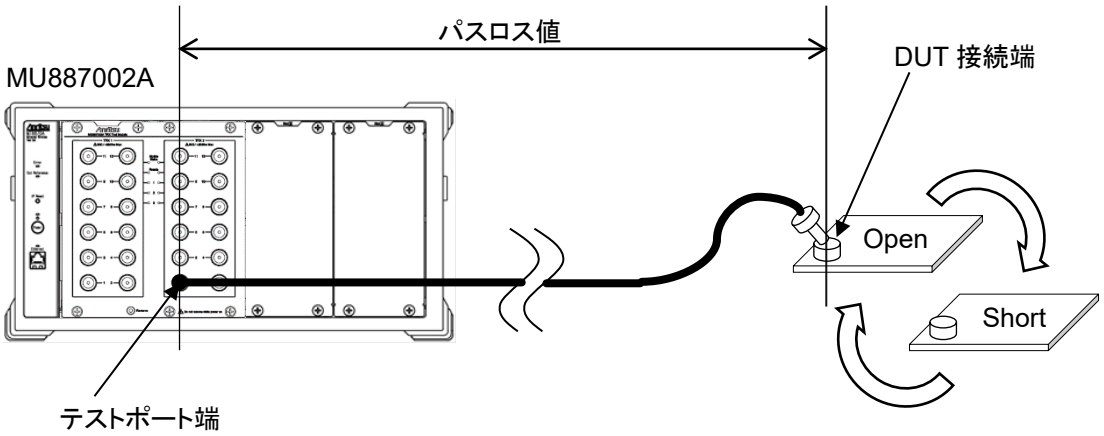


図 E.1.1-1　接続例

注:

DUT 接続端に接続する Open, Short デバイスはお客様に用意していただく必要があります。

E.1.2 構成

MX887092A の標準構成は次のとおりです。

表 E.1.2-1 標準構成

形名	品名	備考
MX887092A	MU887002A パスロス測定機能	MU887000A/01A には対応していません。

E.1.3 規格

規格は，一定の周囲温度でウォームアップ 30 分後の値です。

(Typ.): Typical は 20～30°C における参考データであり，規格値としては保証していません。

(Nom): 設計値であり，規格値としては保証していません。

表 E.1.3-1 規格

項目	規格
周波数範囲	400 MHz≦設定周波数≦6000 MHz
機能	MT8870A Universal Wireless Test Set または MT8872A Universal Wireless Test Set にインストールすることにより，MU887002A を使用した Pathloss 測定を可能とします。
適用	MU887002A TRX 1/2 Port 1～12
測定範囲	0 dB≦Pathloss≦15 dB 1 m≦経路長≦3 m (比誘電率 2.03 の同軸ケーブルにて) 図 E.1.3-1，図 E.1.3-2，図 E.1.3-3 参照
測定確度	400 MHz≦設定周波数≦2700 MHz，移動平均の区間数 11， 経路長 1.5 m，比誘電率 2.03 の同軸ケーブルにて ±0.3 dB (Typ.) ±0.5 dB 3300 MHz≦設定周波数≦6000 MHz，移動平均の区間数 11， 経路長 1.5 m，比誘電率 2.03 の同軸ケーブルにて ±0.6 dB (Typ.) ±0.9 dB
測定再現性	Pathloss≦6 dB，移動平均の区間数 11 にて ±0.1 dB (Nom.) Pathloss≦15 dB，移動平均の区間数 11 にて ±0.15 dB (Nom.)

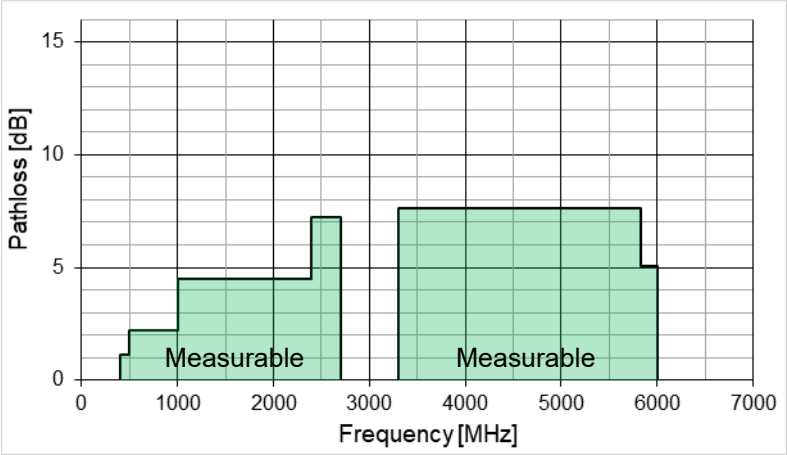


図 E.1.3-1　経路長 1 m 時の測定可能範囲

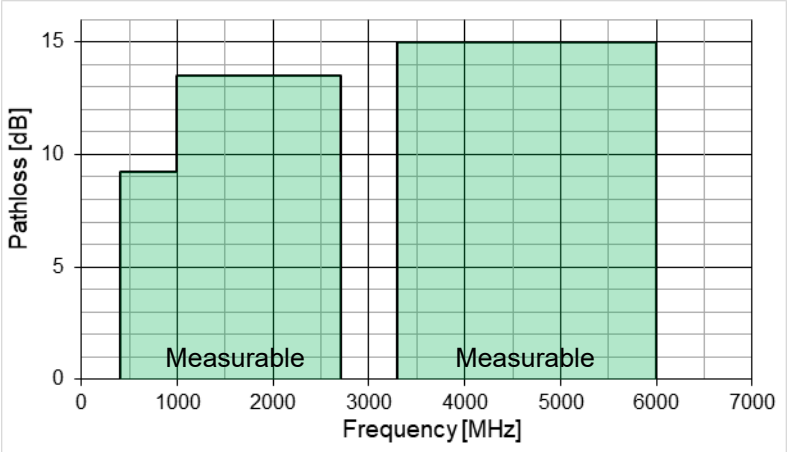


図 E.1.3-2　経路長 1.5 m 時の測定可能範囲

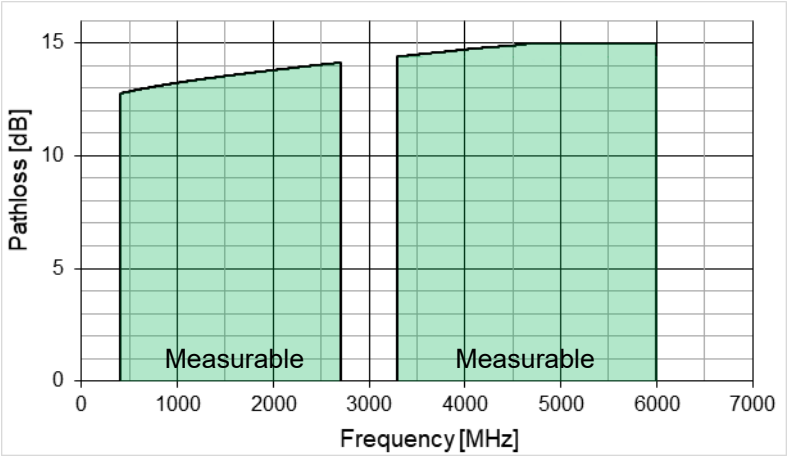


図 E.1.3-3　経路長 3 m 時の測定可能範囲

E.2 ライセンス登録/確認

ソフトウェアライセンスの登録

MX887092A パスロス測定機能を使用するには，MT8870A にライセンスを登録する必要があります。

ライセンス登録の方法は，『MU887002A 送受信テストモジュール 取扱説明書』の「第 8 章ユーティリティツール」を参照してください。

注:

MX887092A は MU887002A でのみ使用可能です。

対応周波数は MU887002A-007 7GHz 拡張機能付きの MU887002A でも 400 MHz～6 GHz です。

ライセンスの確認　　第 5 章，第 6 章のコマンドを参照してください。

- MX887092A ソフトウェアライセンス

  `MFINFO?`

  `:SYSTem:INFormation:MAINframe?`

  UtilityTool　　参照: 8.3.4　機器の情報表示
- MX887092A ソフトウェアライセンスの有効期限

  `KEYINFO?`

  `:SYSTem:BASE:LICense:INFormation?`

E.3　パスロス測定

MX887092A は DUT 接続端に Open, Short デバイスを接続したときのそれぞれの反射信号を測定することで MU887002A のテストポート端から DUT 接続端までのパスロス値を測定します。

⚠ 注意

被測定装置と測定器類は，特に指示する場合を除き少なくとも 30 分間は予熱を行い，十分に安定してから測定を行ってください。最高の測定確度を発揮するには，上記のほかに室温下での実施，AC 電源電圧の変動が少ないこと，騒音・振動・ほこり・湿気などについても全く問題がないことが必要です。

E.3.1　測定

PC からコマンドを送信し，測定を行います。
テストポートごとに，8 つの経路番号（Path index）を設定することができます。

補正バージョンと検査バージョンの確認

- Utility > Pathloss Information　　参照: 8.3.4　機器の情報表示
  補正バージョンと検査バージョンを確認します。
- 校正値の補正バージョンまたは対応周波数範囲
  `SYST:INF:CAL:PLOS:CORR:VERS?`
  `:SYSTem:INFormation:CALibration:PLOSs:CORRection:VERSion?`
- 校正値の検査バージョン
  `SYST:INF:CAL:PLOS:CORR:INSP:VERS?`
  `:SYSTem:INFormation:CALibration:PLOSs:CORRection:INSPection:VERSion?`

測定前の設定

- パスロス値，設定のクリア
  `INIT:CAL:MEAS:PLOS:CLE`
  `:INITiate:CALibration:MEASurement:PLOSs:CLEar`
- パスロス測定周波数の設定
  `CONF:CAL:MEAS:PLOS:MPAT:FREQ`
  `:CONFigure:CALibration:MEASurement:PLOSs:MPATh:FREQuency`
- Path index 1 のパスロス測定周波数の設定
  `CONF:CAL:MEAS:PLOS:FREQ`
  `:CONFigure:CALibration:MEASurement:PLOSs:FREQuency`
- パスロス計算に使用する移動平均の区間数の設定
  `CONF:CAL:MEAS:PLOS:AFT`
  `:CONFigure:CALibration:MEASurement:PLOSs:AFTaps`

測定

- Open デバイスでのパスロス測定の実行
  `INIT:CAL:MEAS:PLOS:OPEN`
  `:INITiate:CALibration:MEASurement:PLOSs:OPEN`
- Short デバイスでのパスロス測定の実行
  `INIT:CAL:MEAS:PLOS:SHOR`
  `:INITiate:CALibration:MEASurement:PLOSs:SHORt`
- Open デバイスで測定したパスロス測定の状態
  `FETC:CAL:MEAS:PLOS:OPEN?`
  `:FETCh:CALibration:MEASurement:PLOSs:OPEN?`
- Short デバイスで測定したパスロス測定の状態
  `FETC:CAL:MEAS:PLOS:SHOR?`
  `:FETCh:CALibration:MEASurement:PLOSs:SHORt?`

計算

- パスロス値の計算

  INIT:CAL:MEAS:PLOS:EVAL

  :INITiate:CALibration:MEASurement:PLOSs:EVALuate
- パスロス計算の状態

  FETC:CAL:MEAS:PLOS:EVAL?

  :FETCh:CALibration:MEASurement:PLOSs:EVALuate?
- パスロス計算結果の周波数

  FETC:CAL:MEAS:PLOS:EVAL:FREQ?

  :FETCh:CALibration:MEASurement:PLOSs:EVALuate:FREQuency?
- パスロス計算結果 (Gain)

  FETC:CAL:MEAS:PLOS:EVAL:GAIN?

  :FETCh:CALibration:MEASurement:PLOSs:EVALuate:GAIN?

E.3.2　パスロス値の設定

測定したパスロス値は，損失補正データとして設定することができます。
詳細は「3.5.3 ケーブルの損失補正」を参照してください。

注：

- パスロス値は Gain なので，補正値にするため，符号を逆にします。
- DL (受信) / UL (送信) 損失の補正値は同値を使用します。

E.3.3　使用例

パスロスを測定してパスロス値を計算する手順を示します。
例として Port 11, Path index 1 とします。

表 E.3.5-1　Pathloss 測定手順

No.	手順	コマンド
1	測定器を接続します。	
2	パスロス値, 設定をクリア（初期化）します。	:INIT:CAL:MEAS:PLOS:CLE
3	パスロス計算に使用する移動平均の区間数を設定します。（例：15）	:CONF:CAL:MEAS:PLOS:AFT 15
4	パスロス測定周波数を設定します。 (例：1000 MHz, 2000 MHz, 3000 MHz の 3 ポイント)	:CONF:CAL:MEAS:PLOS:MPAT:FREQ PORT11,1,3,1000MHZ,2000MHZ,3000MHZ
5	Port 11 につながる DUT 接続端に Open デバイスを接続します。	
6	Open デバイスのパスロス測定を実行します。	:INIT:CAL:MEAS:PLOS:OPEN PORT11,1
7	終了を待ちます。	*OPC?
8	Open デバイスのパスロス測定が正常終了したことを確認します。	:FETC:CAL:MEAS:PLOS:OPEN? PORT11,1
9	Port 1～10, 12 についても, 4～8 を繰り返します。	
10	Port 11 につながる DUT 接続端に Short デバイスを接続します。	
11	Short デバイスのパスロス測定を実行します。	:INIT:CAL:MEAS:PLOS:SHOR PORT11,1
12	終了を待ちます。	*OPC?
13	Short デバイスのパスロス測定が正常終了したことを確認します。	:FETC:CAL:MEAS:PLOS:SHOR? PORT11,1
14	Port 1～10, 12 についても, 10～13 を繰り返します。	
15	パスロス値を計算します。	:INIT:CAL:MEAS:PLOS:EVAL PORT11,1
16	パスロス計算が正常終了したことを確認します。	:FETC:CAL:MEAS:PLOS:EVAL? PORT11,1
17	パスロスの計算結果の周波数を問い合わせます。	:FETC:CAL:MEAS:PLOS:EVAL:FREQ? PORT11,1
18	パスロスの計算結果（Gain）を問い合わせます。	:FETC:CAL:MEAS:PLOS:EVAL:GAIN? PORT11,1
19	Port 1～10, 12 についても, 15～18 を繰り返します。	

表 E.3.5-1　パスロス測定結果を反映する

No.	手順	コマンド
1	言語モード（SCPI）を設定します。	:SYST:LANG SCPI
2	損失補正テーブルのインデックスを設定します。 例）補正テーブルインデックス 1	:CALC:EXTL:TABL:SETT 1
3	損失補正テーブルの補正値を削除します。	:CALC:EXTL:TABL:DEL
4	損失補正テーブルにパスロス計算実行結果の周波数と Gain を設定します。 Gain を補正値にするため，符号を逆にします。 DL（受信）/ UL（送信）損失補正値は同値を使用します。 設定しない Port の補正値は任意とします。 例）　Port:　1 周波数:　701000000 Hz Gain:　−0.21	:CALC:EXTL:TABL:VAL:ALL 701000000,0.21,0.21,0,0,0,0,0,0,0,0,0,0,0,0 ,0,0,0,0,0,0,0,0,0
5	パスロス計算結果の全周波数で 4 を実施します。	
6	損失補正テーブルを有効にします。	:ROUT:EXTL:TABL:SWIT ON

E.4　SCPIコマンド

パスロス測定に使用する SCPI コマンドについて説明します。

SCPI コマンドモードに切り替えるには，SYST:LANG SCPI を送信します。

メッセージの文法の記載に使用する記号と使用方法を次の表に示します。

[ ]	角括弧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。

E.4.1　コマンド一覧

Pathloss measurement

機能	コマンド	クエリ	レスポンス
Clear pathloss data	:INITiate:CALibration:MEASurement:PLOSs:CLEar [<port>[,<pathindex>]]	-----	-----
Pathloss measurement frequency	:CONFigure:CALibration:MEASurement:PLOSs:MPATh:FREQuency <port>,<pathindex>,<numEntries>,<frequency_1>,…,<frequency_numentries>	:CONFigure:CALibration:MEASurement:PLOSs:MPATh:FREQuency? <port>,<pathindex>	<numentries>,<frequency_1>,<frequency_2>,…,<frequency_numentries>
Pathloss measurement frequency for path index 1	:CONFigure:CALibration:MEASurement:PLOSs:FREQuency <port>,<numentries>,<frequency_1>,…,<frequency_numentries>	:CONFigure:CALibration:MEASurement:PLOSs:FREQuency? <port>	<numentries>,<frequency_1>,<frequency_2>,…,<frequency_numentries>
Number of intervals for moving average	:CONFigure:CALibration:MEASurement:PLOSs:AFTaps <avgfiltertaps>	:CONFigure:CALibration:MEASurement:PLOSs:AFTaps?	<avgfilterTaps>
Pathloss measurement with Open	:INITiate:CALibration:MEASurement:PLOSs:OPEN <port>[,<pathindex>]	-----	-----
Pathloss measurement with Short	:INITiate:CALibration:MEASurement:PLOSs:SHORt <port>[,<pathindex>]	-----	-----

Pathloss measurement

機能	コマンド	クエリ	レスポンス
Pathloss measurement status with Open	-----	:FETCh:CALibration:MEASurement:PLOSs:OPEN? <port>[,<pathindex>]	<status>
Pathloss measurement status with Short	-----	:FETCh:CALibration:MEASurement:PLOSs:SHORt? <port>[,<pathindex>]	<status>
Calculate pathloss value	:INITiate:CALibration:MEASurement:PLOSs:EVALuate <Port>[, <pathindex>]	-----	-----
Pathloss calculation status	-----	:FETCh:CALibration:MEASurement:PLOSs:EVALuate? <port>[,<pathindex>]	<status>
Frequency of pathloss calculation result	-----	:FETCh:CALibration:MEASurement:PLOSs:EVALuate:FREQuency? <port>[,<pathindex>]	<numEntries>,<frequency_1>,<frequency_2>,…,<frequency_numentries>
Pathloss calculation result (Gain)	-----	:FETCh:CALibration:MEASurement:PLOSs:EVALuate:GAIN? <port>[,<pathindex>]	<numEntries>,<gain_1>,<gain_2>,…,<gain_numentries>
Correction version of calibration value	-----	:SYSTem:INFormation:CALibration:PLOSs:CORRection:VERSion? [<index>]	<ver> or <type>
Inspection version of calibration value	-----	:SYSTem:INFormation:CALibration:PLOSs:CORRection:INSPection:VERSion?	<ver>

E.4.2 コマンドの詳細

コマンド表の見方，サフィックスコード一覧は，「5.2 コマンドの詳細」を参照してください。
この節では SCPI コマンドを，アルファベット順に説明しています。

:CONFigure:CALibration:MEASurement:PLOSs:AFTaps

Number of intervals for moving average

機能

パスロス計算に使用する移動平均の区間数を設定，または問い合わせます。

コマンド

```
:CONFigure:CALibration:MEASurement:PLOSs:AFTaps <avgfiltertaps>
```

クエリ

```
:CONFigure:CALibration:MEASurement:PLOSs:AFTaps?
```

レスポンス

```
<avgfiltertaps>
```

パラメータ

<avgfiltertaps>	移動平均の区間数
範囲	1～41
初期値	11

使用例

区間数を 1 に設定します:

```
:CONF:CAL:MEAS:PLOS:AFT 1
:CONF:CAL:MEAS:PLOS:AFT?
>1
```

備考

MX887092A ライセンスがない場合，ライセンスエラーとなります。

:CONFigure:CALibration:MEASurement:PLOSs:FREQuency

Pathloss measurement frequency for path index 1

機能

Path index 1 のパスロス測定の周波数を設定, または問い合わせます。

コマンド

```
:CONFigure:CALibration:MEASurement:PLOSs:FREQuency
<port>,<numentries>,<frequency_1>,…,<frequency_numentries>
```

クエリ

```
:CONFigure:CALibration:MEASurement:PLOSs:FREQuency? <port>
```

レスポンス

```
<numentries>,<frequency_1>,<frequency_2>,…,<frequency_numentries>
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<numentries>	測定周波数の数
範囲	1～1000
初期値	0 (設定できません)
<frequency_1>,<frequency_2>,…,<frequency_numentries>	
	測定周波数
範囲	400～6000 MHz
分解能	1 Hz
レスポンス単位	Hz
サフィックスコード	HZ, KHZ, KZ, MHZ, MZ, GHZ, GZ, 省略した場合は Hz として扱われます。

使用例

Test Port 1, Path index 1 に 7 つのパスロス測定の周波数を設定します:

```
:CONF:CAL:MEAS:PLOS:FREQ
PORT1,7,411111111HZ,500000KHZ,600000KZ,700MHZ,800MZ,1GHZ,0.4GZ
:CONF:CAL:MEAS:PLOS:FREQ? PORT1
>7,411111111,500000000,600000000,700000000,800000000,1000000000,400000000
```

関連コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:CLEar
:CONFigure:CALibration:MEASurement:PLOSs:MPATh:FREQuency
```

備考

MX887092A ライセンスがない場合, ライセンスエラーとなります。

:CONFigure:CALibration:MEASurement:PLOSs:MPATh:FREQuency

Pathloss measurement frequency

機能

パスロス測定の周波数を設定します。

コマンド

```
:CONFigure:CALibration:MEASurement:PLOSs:MPATh:FREQuency
<port>,<pathindex>,<numentries>,<frequency_1>,…,<frequency_numentries>
```

クエリ

```
:CONFigure:CALibration:MEASurement:PLOSs:MPATh:FREQuency?
<port>,<pathindex>
```

レスポンス

```
<numentries>,<frequency_1>,<frequency_2>,…,<frequency_numentries>
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号 (Path index)
範囲	1～8
<numentries>	測定周波数の数
範囲	1～1000
初期値	0 (設定できません)
<frequency_1>,<frequency_2>,…,<frequency_numentries>	測定周波数
範囲	400～6000 MHz
分解能	1 Hz
レスポンス単位	Hz
サフィックスコード	HZ, KHZ, KZ, MHZ, MZ, GHZ, GZ, 省略した場合は Hz として扱われます。

使用例

Test Port 1, Path index 1 に 7 つのパスロス測定の周波数を設定します:

```
:CONF:CAL:MEAS:PLOS:MPAT:FREQ
PORT1,1,7,411111111HZ,500000KHZ,600000KZ,700MHZ,800MZ,1GHZ,0.4GZ
:CONF:CAL:MEAS:PLOS:MPAT:FREQ? PORT1,1
>7,411111111,500000000,600000000,700000000,800000000,1000000000,400000000
```

関連コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:CLEar
:CONFigure:CALibration:MEASurement:PLOSs:FREQuency
```

備考

MX887092A ライセンスがない場合, ライセンスエラーとなります。

:FETCh:CALibration:MEASurement:PLOSs:EVALuate?

Pathloss calculation status

機能

パスロス値の計算状態を問い合わせます。

クエリ

```
:FETCh:CALibration:MEASurement:PLOSs:EVALuate? <port>[,<pathindex>]
```

レスポンス

```
<status>
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号（Path index）
範囲	1～8
省略時	1
<status>	計算状態
0	準備中
1	正常終了
2	計算結果不適

使用例

Test Port 1, Path index 1 のパスロス値の計算状態を問い合わせます:

```
:FETC:CAL:MEAS:PLOS:EVAL? PORT1,1
>1
```

関連コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:EVALuate
```

備考

MX887092A ライセンスがない場合，ライセンスエラーとなります。

:FETCh:CALibration:MEASurement:PLOSs:EVALuate:FREQuency?

Frequency for pathloss calculation result

機能

パスロスの計算結果の周波数を問い合わせます。

クエリ

```
:FETCh:CALibration:MEASurement:PLOSs:EVALuate:FREQuency?
<port>[,<pathindex>]
```

レスポンス

```
<numentries>,<frequency_1>,<frequency_2>,…,<frequency_numentries>
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号（Path index）
範囲	1～8
省略時	1
<numentries>	周波数の数
範囲	0～1000
初期値	0
<frequency_1>,<frequency_2>,…,<frequency_numentries>	測定周波数
範囲	400 000 000～6000 000 000
単位	Hz

使用例

Test Port 1, Path index 1 のパスロスの計算結果の周波数を問い合わせます:

```
:FETC:CAL:MEAS:PLOS:EVAL:FREQ? PORT1,1
>7,411111111,500000000,600000000,700000000,800000000,1000000000,400000000
```

関連コマンド

```
:FETCh:CALibration:MEASurement:PLOSs:EVALuate?
:FETCh:CALibration:MEASurement:PLOSs:EVALuate:GAIN?
```

備考

MX887092A ライセンスがない場合, ライセンスエラーとなります。

:FETCh:CALibration:MEASurement:PLOSs:EVALuate:GAIN?

Pathloss calculation result (Gain)

機能

パスロス値の計算結果の Gain を問い合わせます。

クエリ

```
:FETCh:CALibration:MEASurement:PLOSs:EVALuate:GAIN? <port>[,<pathindex>]
```

レスポンス

```
<numentries>,<gain_1>,<gain_2>,…,<gain_numentries>
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号（Path index）
範囲	1～8
省略時	1
<numentries>	
範囲	0～1000
初期値	0
<gain_1>,<gain_2>,…,<gain_numentries>	
	パスロス値
範囲	–100.00～100.00
単位	dB

詳細

本コマンドで取得する値は Gain 値になります。ケーブル損失補正データとして使用する場合，符号を逆にする必要があります。

使用例

Test Port 1，Path index 1 のパスロス値を問い合わせます：

```
:FETC:CAL:MEAS:PLOS:EVAL:GAIN? PORT1,1
>7,-0.26,-0.30,-0.30,-0.27,-0.26,-0.26,-0.29
```

関連コマンド

```
:FETCh:CALibration:MEASurement:PLOSs:EVALuate?
:FETCh:CALibration:MEASurement:PLOSs:EVALuate:FREQuency?
```

備考

MX887092A ライセンスがない場合，ライセンスエラーとなります。

:FETCh:CALibration:MEASurement:PLOSs:OPEN?

Pathloss measurement status with Open

機能

Open デバイスで測定したパスロス測定の状態を問い合わせます。

クエリ

```
:FETCh:CALibration:MEASurement:PLOSs:OPEN? <port>[,<pathindex>]
```

レスポンス

```
<status>
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号 (Path index)
範囲	1～8
省略時	1
<status>	パスロス測定の状態
0	準備中
1	正常終了

使用例

Open デバイスで測定した Test Port 1, Path index 1 のパスロス測定の状態を問い合わせます:

```
:FETC:CAL:MEAS:PLOS:OPEN? PORT1,1
>1
```

関連コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:OPEN
```

備考

MX887092A ライセンスがない場合, ライセンスエラーとなります。

:FETCh:CALibration:MEASurement:PLOSs:SHORt?

Pathloss measurement status with Short

機能

Short デバイスで測定したパスロス測定の状態を問い合わせます。

クエリ

```
:FETCh:CALibration:MEASurement:PLOSs:SHORt? <port>[,<pathindex>]
```

レスポンス

```
<status>
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号（Path index）
範囲	1～8
省略時	1
<status>	パスロス測定の状態
0	準備中
1	正常終了

使用例

Short デバイスで測定した Test Port 1, Path index 1 のパスロス測定の状態を問い合わせます:

```
:FETC:CAL:MEAS:PLOS:SHOR? PORT1,1
>1
```

関連コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:SHORt
```

備考

MX887092A ライセンスがない場合，ライセンスエラーとなります。

:INITiate:CALibration:MEASurement:PLOSs:CLEar

Clear Pathloss data

機能

パスロス値, 設定をクリア（初期化）します。

コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:CLEar [<port>[,<pathindex>]]
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
省略時	全 Port
<pathindex>	経路番号（Path index）
範囲	1～8
省略時	全経路番号

詳細

下記データをクリア（初期化）します。

- 周波数ポイント
- Open デバイスでのパスロス測定値/状態
- Short デバイスでのパスロス測定値/状態
- パスロス計算値/状態

すべてのパラメータを省略した場合, 以下のデータも初期化されます。

- 移動平均の区間数

使用例

Test Port 1, PathIndex 1 のパスロス値, 設定をクリア（初期化）します:

```
:INIT:CAL:MEAS:PLOS:CLE PORT1,1
```

備考

- *RST では, 詳細で記載のデータはクリア(初期化) されません。
- MX887092A ライセンスがない場合, ライセンスエラーとなります。

:INITiate:CALibration:MEASurement:PLOSs:EVALuate

Calculate pathloss value

機能

パスロス値を計算します。

コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:EVALuate <port>[,<pathindex>]
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号（Path index）
範囲	1～8
省略時	1

詳細

本コマンドを使用する場合，下記が必要です。

- パラメータで指定する Port，Path index に対して Open，Short デバイスのパスロス測定が事前に実施されていること
- 周波数ポイント，移動平均の区間数が設定されていること

使用例

Test Port 1，Path index 1 のパスロス値を計算します:

```
:INIT:CAL:MEAS:PLOS:EVAL PORT1,1
```

関連コマンド

```
:FETCh:CALibration:MEASurement:PLOSs:EVALuate?
```

備考

- MX887092A ライセンスがない場合，ライセンスエラーとなります。
- 必要なデータがそろっていないときは実行エラーとなります。

:INITiate:CALibration:MEASurement:PLOSs:OPEN

Pathloss measurement with Open

機能

Open デバイスでのパスロス測定を実行します。

コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:OPEN <port>[,<pathindex>]
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号 (Path index)
範囲	1～8
省略時	1

詳細

本コマンドを使用する場合，パラメータで指定するテストポートにつながる DUT 接続端に Open デバイスを接続します。

使用例

Open デバイスで Test Port 1, Path index 1 のパスロス測定を実行します:

```
:INIT:CAL:MEAS:PLOS:OPEN PORT1,1
```

関連コマンド

```
:FETCh:CALibration:MEASurement:PLOSs:OPEN?
```

備考

- MX887092A ライセンスがない場合，ライセンスエラーとなります。
- 波形メモリに空き容量がないときは実行エラーとなります。その場合，波形メモリから不要な波形ファイルを削除します。波形メモリの空き容量確認および波形ファイルの削除は以下のコマンドを参照してください。

```
:SOURce:GPRF:GENerator:ARB:WAVeform:FREE?
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL
```

:INITiate:CALibration:MEASurement:PLOSs:SHORt

Pathloss measurement with Short

機能

Short デバイスでのパスロス測定を実行します。

コマンド

```
:INITiate:CALibration:MEASurement:PLOSs:SHORt <port>[,<pathindex>]
```

パラメータ

<port>	テストポート番号
範囲	Test Port 1～12
PORT1	Test Port 1
…	…
PORT12	Test Port 12
<pathindex>	経路番号 (Path index)
範囲	1～8
省略時	1

詳細

本コマンドを使用する場合，パラメータで指定するテストポートにつながる DUT 接続端に Short デバイスを接続します。

使用例

Short デバイスで Test Port 1, Path index 1 のパスロス測定を実行します:

```
:INIT:CAL:MEAS:PLOS:SHOR PORT1,1
```

関連コマンド

```
:FETCh:CALibration:MEASurement:PLOSs:SHORt?
```

備考

- MX887092A ライセンスがない場合，ライセンスエラーとなります。
- 波形メモリに空き容量がないときは実行エラーとなります。その場合，波形メモリから不要な波形ファイルを削除します。波形メモリの空き容量確認および波形ファイルの削除は以下のコマンドを参照してください。

```
:SOURce:GPRF:GENerator:ARB:WAVeform:FREE?
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete
:SOURce:GPRF:GENerator:ARB:WAVeform:DELete:ALL
```

:SYSTem:INFormation:CALibration:PLOSs:CORRection:INSPection:VERSion?

Inspection version of calibration value

機能

校正値の検査バージョンを問い合わせます。

クエリ

```
:SYSTem:INFormation:CALibration:PLOSs:CORRection:INSPection:VERSion?
```

レスポンス

<ver>　　検査バージョン

パラメータ

なし

詳細

校正値の検査バージョンを問い合わせます。

使用例

校正値の検査バージョンを問い合わせます:

```
:SYST:INF:CAL:PLOS:CORR:INSP:VERS?
> 0
```

関連コマンド

```
:SYSTem:INFormation:CALibration:PLOSs:CORRection:VERSion?
```

:SYSTem:INFormation:CALibration:PLOSs:CORRection:VERSion?

Correction version of calibration value

機能

校正値の補正バージョンまたは対応周波数範囲を読み出します。

クエリ

```
:SYSTem:INFormation:CALibration:PLOSs:CORRection:VERSion? [<index>]
```

レスポンス

パラメータが 0, または省略の場合

<ver>	補正バージョン
範囲	0～2147483647

パラメータが 1 の場合

<type>	対応周波数範囲
6GHZ	400 MHz～6 GHz
6GHZ&7GHZ	400 MHz～7.3 GHz (将来対応予定)

パラメータ

<index>	レスポンス種別
0	補正バージョン
1	対応周波数範囲
省略	0 として扱われます。

詳細

校正値の補正バージョンを読み出します。
校正値の対応している周波数範囲を読み出します。

使用例

校正値の補正バージョンを問い合わせます:

```
:SYST:INF:CAL:PLOS:CORR:VERS?
> 0
```

校正値の対応している周波数範囲を問い合わせます:

```
SYST:INF:CAL:PLOS:CORR:VERS? 1
> 6GHZ
```

関連コマンド

```
:SYSTem:INFormation:CALibration:PLOSs:CORRection:INSPection:VERSion?
```

E.5 Nativeコマンド

パスロス測定に使用する Native コマンドについて説明します。
Native コマンドモードに切り替えるには，SYST:LANG NAT を送信します。

Native コマンドモードでは，SCPI コマンドの省略形を使用します。

例
:INITiate:CALibration:MEASurement:PLOSs:CLEar
→ INIT:CAL:MEAS:PLOS:CLE
:CONFigure:CALibration:MEASurement:PLOSs:MPATh:FREQuency
→ CONF:CAL:MEAS:PLOS:MPAT:FREQ

メッセージの文法の記載に使用する記号と使用方法を次の表に示します。

[ ]	角括弧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。

E.5.1 コマンド一覧

Pathloss measurement

機能	コマンド	クエリ	レスポンス
Clear pathloss data	INIT:CAL:MEAS:PLOS:CLE [port,[pathIndex]]	-----	-----
Pathloss measurement frequency	CONF:CAL:MEAS:PLOS:MPAT:FREQ port,pathIndex,numentries, frequency_1,…,frequency_numentries	CONF:CAL:MEAS:PLOS:MPAT:FREQ? port,pathindex	numentries,frequency_1,frequency_2,…,frequency_numentries
Pathloss measurement frequency for path index 1	CONF:CAL:MEAS:PLOS:FREQ port,numentries,frequency_1,…,frequency_numentries	CONF:CAL:MEAS:PLOS:FREQ? port	numentries,frequency_1,frequency_2,…,frequency_numentries
Number of intervals for moving average	CONF:CAL:MEAS:PLOS:AFT avgfiltertaps	CONF:CAL:MEAS:PLOS:AFT?	avgfiltertaps
Pathloss measurement with Open	INIT:CAL:MEAS:PLOS:OPEN port[,pathindex]	-----	-----
Pathloss measurement with Short	INIT:CAL:MEAS:PLOS:SHOR port[,pathindex]	-----	-----

Pathloss measurement

機能	コマンド	クエリ	レスポンス
Pathloss measurement status with Open	-----	FETC:CAL:MEAS:PLOS:OPEN? port[,pathindex]	status
Pathloss measurement status with Short	-----	FETC:CAL:MEAS:PLOS:SHOR? port[,pathindex]	status
Calculate pathloss value	INIT:CAL:MEAS:PLOS:EVAL port[,pathindex]	-----	-----
Pathloss calculation status	-----	FETC:CAL:MEAS:PLOS:EVAL? port[,pathindex]	status
Frequency of pathloss calculation result	-----	FETC:CAL:MEAS:PLOS:EVAL:FREQ? port[,pathindex]	numentries, frequency_1,frequency_2,…,frequency_numentries
Pathloss calculation result (Gain)	-----	FETC:CAL:MEAS:PLOS:EVAL:GAIN? port[,pathindex]	numentries,gain_1,gain_2,…,gain_numentries
Correction version of calibration value	-----	SYST:INF:CAL:PLOS:CORR:VERS? [index]	ver or type
Inspection version of calibration value	-----	SYST:INF:CAL:PLOS:CORR:INSP:VERS?	ver

参照先はページ番号です。

■アルファベット順

I

IPv4 3-3
IPv6 3-3
IPアドレスの初期値 3-4

N

Native 3-2
　Pathloss measurement..... E-30, E-31

S

SCPI 3-2
　Pathloss measurement..... E-13, E-14
SGシーケンサデータ 4-21
SGシーケンサテーブル.....4-19

V

VSG 4-2
VSWR 7-8

■50音順

あ

アップグレード 8-27
アプリケーションソフトウェア 1-6

い

イーサネットの初期値..... 3-4
一括アップグレード画面 8-32

お

応用部品 1-8
オプション 1-5

き

共通パラメータ 4-16

く

クエスチョナブルレジスタ 3-29

け

言語モード..... 3-2

さ

サービスリクエストイネーブルレジスタ..... 3-22

し

シーケンステーブル 4-16
シーケンスモード 4-3, 4-15
実行状態レジスタ 3-26

す

ステータスバイトレジスタ 3-22
　ビット定義..... 3-23
ステータスランプ 2-4
スプリアス..... 7-22

せ

静電気..... 2-9
性能試験用機器一覧表 7-3
接続を確認する..... 2-7
全バンド校正..... 3-14

そ

測定確度 7-41
損失補正 3-17

ち

直接ファイル転送 8-37

て

テストポート.....3-18

と

トリガコネクタ 3-18

ね

ネットワーク設定 8-22

の

ノーマルモード 4-3, 4-4

は

バージョン..... 8-7
波形ファイル 1-6
波形ファイル転送 8-35
波形リストテーブル.....4-18
バンド校正 3-15

ひ

標準イベントステータスレジスタ
　ビット定義.....3-24
標準構成..... 1-4

ふ

ファイル転送 3-8, 8-30

も

モジュール設定.....8-24

ら

ライセンス登録.....8-34

り

リニアリティ.....7-33
リモートランプ 2-4

れ

レジスタの構成 3-19
レジスタのビットの10進数変換値.....3-21
レベル確度 7-12, 7-25

ろ

ログ 8-26