

SmartMesh IP Embedded Manager CLI ガイド

目次

1	本書について	4
1.1	関連資料	4
1.2	表記規則	6
1.3	改訂履歴	7
2	はじめに	8
2.1	CLIアクセス	8
2.1.1	ログイン/ログアウト	8
2.1.2	ユーザおよびビューアのパスワードの管理	9
2.1.3	モートのコマンド	9
3	コマンド	10
3.1	delete acl	10
3.2	delete mote	11
3.3	exec clearStat	12
3.4	exec exchJoinKey	13
3.5	exec exchNetId	14
3.6	exec restore	15
3.7	exec setAdv	16
3.8	exec setDnFrame	17
3.9	exec sendData	18
3.10	exec start	19
3.11	help	20
3.12	log	21
3.13	login	22
3.14	logout	23
3.15	onechan	24
3.16	ping	25
3.17	radiotest	26
3.17.1	radiotest on/off	26
3.17.2	radiotest tx	27
3.17.3	radiotest rx	29
3.17.4	radiotest stat	30
3.18	reset	31
3.19	set acl	32
3.20	set config	33

3.21	seti	35
3.22	show	36
3.22.1	show acl	37
3.22.2	show configおよびshow curconfig	38
3.22.3	show mac	39
3.22.4	show mote	40
3.22.5	show motever	42
3.22.6	show path	43
3.22.7	show stat	44
3.22.8	show status	46
3.22.9	show time	48
3.22.10	show trace	49
3.22.11	show ver	50
3.23	showi	51
3.24	sm	52
3.25	su	53
3.26	trace	54
3.26.1	trace bw	56
3.26.2	trace fa	57
3.26.3	trace glbcmd	58
3.26.4	trace io(予約済み)	59
3.26.5	trace iodata(予約済み)	60
3.26.6	trace link	61
3.26.7	trace loop	62
3.26.8	trace monitor	63
3.26.9	trace motest	64
3.26.10	trace netmode	65
3.26.11	trace opt	66
3.26.12	trace power	67
3.26.13	trace rawio_enc(予約済み)	68
3.26.14	trace route	69
3.26.15	trace spl_taskおよびtrace spl_ack(予約済み)	70
3.26.16	trace stats	71
3.26.17	trace timeout	72
3.26.18	trace tplglock(予約済み)	73

1 本書について

1.1 関連資料

SmartMesh IPネットワーク向けに以下の資料が提供されています。

スタータ・キットのクイック・ガイド

- [SmartMesh IP Easy Start Guide](#) - 基本的なインストール方法とネットワークの動作確認テストについて説明しています。
- [SmartMesh IPツール・ガイド](#) - インストールのセクションではシリアル・ドライバのインストール手順について説明しており、Easy Start Guideやその他のチュートリアルで使用されるサンプル・プログラムも含まれています。

ユーザ・ガイド

- [SmartMesh IPユーザ・ガイド](#) - ネットワーク概念についての説明と、モートおよびマネージャのAPIを使用して特定のタスク(データ送信や統計情報の収集など)を実行する方法について説明しています。この資料は、APIガイドを使用するための予備知識を提供します。

デバイスの対話操作インターフェース

- [SmartMesh IP Manager CLI Guide](#) - ユーザがマネージャとやり取りするために使用します(クライアントの開発中やトラブルシューティングなど)。このガイドは、CLIの接続とそのコマンド・セットについて説明しています。
- [SmartMesh IP Manager API Guide](#) - プログラムを使用してマネージャとやり取りするために使用します。このガイドは、APIの接続とそのコマンド・セットについて説明しています。
- [SmartMesh IP Mote CLI Guide](#) - ユーザがモートとやり取りするために使用します(センサー・アプリケーションの開発中やトラブルシューティングなど)。このガイドは、CLIの接続とそのコマンド・セットについて説明しています。
- [SmartMesh IP Mote API Guide](#) - プログラムを使用してモートとやり取りするために使用します。このガイドは、APIの接続とそのコマンド・セットについて説明しています。

ソフトウェア開発ツール

- [SmartMesh IPツール・ガイド](#) - [SmartMesh SDK](#)に含まれる各種の評価および開発サポート・ツールについて説明しています。モートおよびマネージャAPIの使用とネットワークの視覚化のためのツールを含みます。

アプリケーション・ノート

- [SmartMesh IPアプリケーション・ノート](#) - SmartMesh IPネットワーク固有の各種トピックと、SmartMeshネットワーク全般に当てはまるトピックが含まれています。

新規設計の開始時に役立つ資料

- [LTC5800-IPM SoC](#)またはこれに基づく[モジュール](#)のデータシート。
- [LTC5800-IPR SoC](#)またはこれに基づく[組み込みマネージャ](#)のデータシート。
- モート/マネージャSoC用または[モジュール用のハードウェア統合ガイド](#) - 設計にSoCまたはモジュールの統合を盛り込むためのベスト・プラクティスを提供しています。

- 組み込みマネージャ用の[ハードウェア統合ガイド](#) - 設計する際に組み込みマネージャを統合させるためのベスト・プラクティスを提供しています。
- [Board Specific Integration Guide](#) - SoCモートおよびマネージャに対するデフォルトのI/O設定方法と、「ヒューズ表」を使用した水晶発振器のキャリブレーション情報について説明しています。
- [Hardware Integration Application Notes](#) - SoC設計チェックリスト、アンテナ選定ガイドなどを含みます。
- [ESP Programmer Guide](#) - DC9010 Programmer Boardと、デバイスへのファームウェアのロードに使用するESPソフトウェアのガイドです。
- ESPソフトウェア - モートまたはモジュールにファームウェア・イメージをプログラミングするために使用します。
- Fuse Tableソフトウェア - [Board Specific Configuration Guide](#)で説明されているヒューズ表を作成するために使用します。

その他の役立つ資料

- SmartMesh技術文書で使用されるワイヤレス・ネットワークの用語については、[SmartMesh IPユーザ・ガイド](#)を参照してください。
- [よくある質問の一覧](#)。

1.2 表記規則


本書では、以下の表記規則を使用します。


コンピュータ・タイプ (Computer type) は、URLの指定など、ユーザが入力する情報を示します。


太字は、ボタン、フィールド、メニュー・コマンド、デバイス・ステート、モードを示します。


斜体は、新しい用語やAPIとそのパラメータを示します。

i>

 ヒントは、製品に関して役立つ情報を提供します。

 情報テキストは、背景や前後関係の理解に役立つ追加情報を提供します。

 注記は、概念についてより詳しい説明を提供します。

 警告！ 警告は、データ損失やハードウェアまたはユーザへの物理的な損害を引き起こす可能性のある動作をユーザに知らせます。

コード・ブロックは、コード例を示します。

CLIコマンドの記述には、以下の表記と用語を使用します。

	フィールドに対する選択肢を表します。例えば、 <moteld> #<MAC> は、モートの指定にモートIDまたはMACアドレスを使用できることを意味します。
<>	必須フィールドを表します。
{ }	フィールドのグループを表します。
[]	省略可能フィールドを表します。
MAC アドレス	MACアドレスの指定では、スペースを使用しないでください。先頭のゼロとハイフンは省略できます。MACアドレスまたはモートIDのどちらかを指定できるコマンド構文では、MACアドレスの先頭に#記号を付加する必要があります。 以下の例はいずれも有効です。 22CA 00000000000022CA 00-00-00-00-00-00-22-CA

1.3 改訂履歴

リビジョン	日付	説明
1	2012/07/17	初期リリース
2	2012/08/10	radiotest txコマンドの更新
3	2013/03/18	多数の軽微な変更
4	2013/10/22	軽微な修正
5	2014/04/04	無線テスト・コマンドの更新および明確化
6	2014/10/28	show moteverコマンドの明確化、その他の軽微な変更
7	2015/04/22	autostartコマンドの廃止、その他の軽微な変更
8	2015/12/03	ソフトウェア・ライセンス・コマンドの廃止、show memoryコマンドの追加、組込みマネージャに焦点を合わせるためのガイド名の変更
9	2016/11/07	deleteコマンドとset configコマンドの更新、show macコマンドとexec startコマンドの追加

2 はじめに

本書では、コマンド・ライン・インターフェース (CLI) にログオンすることで SmartMesh IP Manager に送信できるコマンドについて説明します。CLI は、シリアル・ターミナル・プログラムを Manager に接続することで使用できます。CLI は、開発中やインタラクティブなトラブルシューティングなどで、ユーザがマネージャとやり取りするためのものです。ほとんどのコマンドはアトミックであるため、1つのコマンドとその引数が CLI に入力されると、1つのレスポンスが返されます。例を挙げると、`help` コマンドでは使用可能なコマンドのリストを返します。トレースはアトミックではないため、コマンドを取り消すまで出力が非同期で生成されます。

マシン間通信 (ホストプログラムとマネージャの通信など) には、アプリケーション・プログラミング・インターフェース (API) を使用します。詳細については、[SmartMesh IP Manager API Guide](#) を参照してください。

2.1 CLI アクセス

SmartMesh IP Manager には専用シリアル・ポートが2つあります。1つは外部アプリケーションとの API 通信用であり、もう1つはこのコマンド・ライン・インターフェース専用です。

この CLI には、任意のシリアル・ターミナル・プログラム (HyperTerminal や Tera Term など) からログオンできます。

- **シリアル0** – FTDI の USB シリアル・インターフェースを統合した評価用ボードに接続する場合、システムにマッピングされた3番目の COM ポートで CLI を使用できます。

デフォルトのシリアル・ポート設定は、9600 ボー、8 データ・ビット、パリティなし、ストップ・ビット1、フロー制御なしです。

2.1.1 ログイン／ログアウト

このシステムに設定された権限は、ユーザとビューアの2種類です。ユーザ権限ではシステム設定を指定できますが、ビューア権限で実行できるのは、マネージャおよびネットワーク情報の表示のみです。

ターミナル・プログラムからマネージャの CLI にログインするには、以下のどちらかのユーザ名およびパスワードを入力します。

```
login viewer  
  
login user
```

マネージャの CLI からログアウトするには、以下を入力します。

```
logout
```


2.1.2 ユーザおよびビューアのパスワードの管理

デフォルト・パスワードは、以下のコマンドを使用して変更する必要があります(「ユーザ」権限でログインした後)。

```
set config pwdviewer <newpassword>  
  
set config pwduser <newpassword>
```

2.1.3 モートのコマンド

「m」で始まるコマンド(mtraceやminfoなど)は、アクセス・ポイントであるモート専用です。詳細については、[SmartMesh IP Mote CLI Guide](#)を参照してください。

3 コマンド

本書では、SmartMesh IP Managerで使用できるCLIコマンドについて説明します。このCLIでは、大文字と小文字を区別する必要があります。ほとんどの場合、コマンドは最も短い一義的な文字列で認識されるため、以下はいずれも同じ意味になります。

```
> trace rawio_enc on  
  
> trace rawio on  
  
> trace raw on
```

3.1 delete acl

説明

`delete acl`コマンドは、アクセス制御リスト(ACL)からモートのエントリを削除し、モートの参加鍵と参加カウンタをクリアします。ACLからモートを削除すると、モートはネットワークに参加できなくなります。既にネットワークに参加している場合も、再参加することはできません。ただし、強制的にネットワークから切断されることはありません。すべてのモートがACLから削除された場合、システムは元どおりに共通参加鍵を使用します。これは永続的な変更です。

構文

```
delete acl <macAddress | all>
```

パラメータ

パラメータ	説明
macAddress	特定モートのMACアドレスまたは全モート(all)

例

```
delete acl 00-17-0D-00-00-38-00-21  
  
delete acl all
```

3.2 delete mote

説明

マネージャからモートを削除します。このコマンドは、稼働中のモートをネットワークから削除するのではなく、マネージャが認識しているモート・リストからモートを削除することだけを目的としています。ネットワークからモートを削除できるのは、モートのステータスが**Lost**または**Idle**であるか、モートが**ブリンク・モート**として使用された場合のみです。これは永続的な変更です。

モートの参加カウンタが完全にリフレッシュされた場合、このコマンドを使用して、モートの参加カウンタをクリアする必要があります。カウンタをクリアしない場合、参加カウンタが一致するまでモートがネットワークに参加できません。同じことがblinkモードを使用したパケットにも当てはまり、参加カウンタが一致するまで何も受信できなくなります。

構文

```
delete mote <moteId | mac>
```

パラメータ

パラメータ	説明
moteId mac	ネットワークから削除するモートのIDまたはMACアドレス

例

```
delete mote 2
```

```
delete mote 00-17-0D-00-00-60-06-11
```

3.3 exec clearStat

説明

ネットワーク統計情報をクリアします。

構文

```
exec clearStat
```

パラメータ

パラメータ	説明
-------	----

例

```
exec clearStat
```

3.4 exec exchJoinKey

説明

指定されたモートの参加鍵を置き換えます。このメッセージはモートに送信され、ACL内のエントリも変更されます。これは永続的な変更です。

構文

```
exec exchJoinKey <address> <joinKey>
```

パラメータ

パラメータ	説明
address	変更対象のモートのモートIDまたはMACアドレス
joinKey	16バイトの参加鍵

例

```
exec exchJoinKey 00-17-0D-00-00-38-00-21 000102030405060708090A0B0C0D0E0F
```

3.5 exec exchNetId

説明

ネットワークIDを交換します。このコマンドは、マネージャとネットワークに接続されたすべてのモートのネットワークIDを変更します。新しいネットワークIDは、次にネットワークが再起動されたときに有効になります。0および65535のネットワークIDは予約済みのため、使用しないでください。これは永続的な変更です。

構文

```
exec exchNetId <netId>
```

パラメータ

パラメータ	説明
netId	1～65534の整数

例

```
exec exchNetId 100
```

3.6 exec restore

説明

すべての設定を工場出荷時のデフォルト設定に戻します。これは永続的な変更です。

Managerバージョンが1.3.0未満でライセンスが必要な場合、オプション機能の有効化に使用されるライセンスは、復元中も保持されます。

構文

```
exec restore
```

パラメータ

パラメータ	説明
-------	----

例

```
exec restore
```

3.7 exec setAdv

説明

このコマンドはネットワーク内のアドバタイズメントを制御します。offに設定するとすべてのアドバタイズメントが無効化され、大幅に電力を節約できますが、モートがアドバタイズメントを受信できず、ネットワークにも参加できません。onに設定するとアドバタイズメントが有効になり、モートがネットワークに参加できます。

⚠ ネットワーク内のアドバタイズメントを無効にすることは危険です。アドバタイズメントを無効にした場合、新しいモートがネットワークに参加できず、リセット後にモートが再参加することもできません。非常に珍しいケースで、例えばネットワークにモートを参加させないようにしたり、消費電力を節約したりするために、アドバタイズメントの無効化が役立つ場合もありますが、たいていの場合、アドバタイズメントはマネージャの制御下に置いたままにすることを推奨します。

構文

```
exec setAdv <on|off>
```

パラメータ

パラメータ	説明
state	on = アドバタイズメントの有効化、off = アドバタイズメントの無効化

例

```
exec setAdv on
```


3.8 exec setDnFrame

説明

下りスーパーフレーム・サイズを変更します。このコマンドが有効なのは、下りスーパーフレーム乗数設定パラメータ (*dnfr_mult*) を2または4に設定して、ネットワークを実行している場合のみです。このコマンドは、スロットフレーム乗数をオン(標準)またはオフ(高速)にします。このコマンドを使用すると、ネットワーク形成後のスーパーフレーム・サイズの自動切替えが無効になります。例えば、ネットワークの開始フレーム・サイズが256スロットで、*dnfr_mult*が4に設定されている場合、「normal」に変更すると、スーパーフレーム・サイズが4倍の1024スロットになります。

構文

```
exec setDnFrame <mode>
```

パラメータ

パラメータ	説明
mode	「fast」または「normal」

例

```
exec setDnFrame normal
```

3.9 exec sendData

説明

指定されたペイロードを含むパケットをモートに送信します。このCLIコマンドは、APIコマンドの*sendData*を実行した場合に相当します。

構文

```
exec sendData <destination> <srcPort> <destPort> <priority> <payload>
```

パラメータ

パラメータ	説明
destination	パケットの宛先。モートIDまたはMACアドレスのどちらかを指定できます
srcPort	パケットのUDP送信元ポート
destPort	パケットのUDP宛先ポート
priority	パケットの優先順位。0(低い)~2(高い)
payload	パケットのペイロード・バイト(16進数)。ペイロードの最大サイズは40バイト


例

モートID = 2、送信元ポート = 20、宛先ポート = 20、優先順位 = 1、ペイロード・バイト = 「0x11,0x22,0x33,0x44,0x55」で、モートにパケットを送信します。

```
exec sendData 2 20 20 1 1122334455
```

3.10 exec start

説明

 このコマンドは廃止されているため、使用しないでください。

`exec start`コマンドは、マネージャに指示して、ネットワーク形成を開始できるようにします(デバイスからの参加リクエストの受け入れを開始する)。

3.11 help

説明

ヘルプを表示します。パラメータなしでこのコマンドを入力すると、現在のユーザおよびモードで使用できるすべてのコマンドが表示されます。特定のコマンドに関するヘルプを表示するには、コマンド名を引数として入力します。

構文

```
help [command]
```

パラメータ

パラメータ	説明
command	任意のCLIコマンド

例

```
help
```

3.12 log

説明

モートのフラッシュ・メモリからデバッグ・ログ情報を取得します。これは、モートがネットワークからリセットされた場合のデバッグで使用します。

構文

```
log <moteId>
```

パラメータ

パラメータ	説明
moteld	対象モートのID

例

```
log 2
```

3.13 login

説明

CLIインターフェースはログインを要求し、入力されたパスワードでセッションに使用される権限を特定します。デフォルト・パスワードは2つの権限レベルと同じで、viewerおよびuserです。viewerは、マネージャの設定を変更できません。userは、すべてのコマンドを使用できます。ログアウトなしで、loginコマンドを繰り返し使用して、権限レベルを切替えることができます。2つの権限レベルのパスワードを変更するには、set configコマンドを使用します。

構文

```
login [<user>:] <password>
```

パラメータ

パラメータ	説明
user	viewerまたはuser
password	権限レベルのパスワード。 出荷時のデフォルト・パスワードは、user用が「user」で、viewer用が「viewer」です。 注:パスワードの変更には、set configコマンドを使用します。

例

```
login user: me$h
```

```
login me$h
```

3.14 logout

説明

現在のCLIセッションからログアウトします。

構文

```
logout
```

パラメータ

パラメータ	説明
-------	----


例

```
logout
```

3.15 onechan

説明

ネットワークが単一チャンネルで実行されるように設定します。このコマンドは、RF適合性テストで使用できます。コマンドが有効になるのは、システム・リセットの実行後です。これは永続的な設定であり、通常動作に戻す場合は無効化する必要があります。

 このコマンドのチャンネル番号は0～15で、IEEE 2.4GHzチャンネルの11～26に相当します。

構文

```
onechan <channel>
```

パラメータ

パラメータ	説明
channel	動作チャンネル(0-15)の動作オンまたはオフ状態

例

```
onechan 2  
onechan off
```


3.16 ping

説明

ネットワーク内のモートからの応答を要求します。モートは応答時に温度と電圧を返します。これはDustコマンドであり、ICMPエコーは使用しません。

構文

```
ping <moteId>
```

パラメータ

パラメータ	説明
moteld	ping対象モートのID

例

```
> ping 2  
Sending ping request to mote 2  
> Ping response from mote 2, time=350 msec v=3603 t=26
```

3.17 radiotest

3.17.1 radiotest on/off

説明

デバイス無線テスト・モードを有効化または無効化します。無線テスト機能は、認定やテストの目的で無線を実行するために使用します。このコマンドは再起動後に有効になり、次に変更するまで選択したモードが持続します。つまり、モードがオンの場合、リセットやパワー・サイクル後もオンのままであり、モードがオフに変更されてデバイスが再起動されるまでこれが持続します。

構文

```
radiotest <mode>
```

パラメータ

パラメータ	説明
mode	on - 再起動後にデバイスを無線テスト・モードにする off - 再起動後にデバイスを通常のマスタ・モードにする

例

デバイスを無線テスト・モードにします。

```
radiotest on
```

デバイスを通常動作モードに戻します。

```
radiotest off
```

3.17.2 radiotest tx

説明

`radiotest tx`コマンドを実行すると、ユーザが無線伝送テストを開始できるようになります。このコマンドを実行できるのは、無線テスト・モードのときだけです。以下の4種類の伝送テストがサポートされています。


- pk - パケット伝送
- cm - 連続変調
- cw - 連続波(非変調信号)
- pkcca - クリア・チャンネル・アセスメント(CCA)を有効にしたパケット伝送(1.3.0以上のIP Managerと1.4.0以上のIP Moteで使用可能)


パケット伝送テストでは、デバイスで`repeatCnt`個のパケット・シーケンスが生成されます。各シーケンスは最大10個のパケットで構成され、サイズと遅延を設定できます。各パケットは、最大125バイトのペイロードと末尾にある2バイトの802.15.4 CRCで構成されます。バイト0には送信側の`stationId`が含まれます。バイト1および2にはパケット番号(ビッグエンディアン形式)が含まれ、この番号は伝送パケットごとにインクリメントされます。バイト3~Nにはカウンタ(0~N-3)が含まれ、このカウンタはペイロード内のバイトごとにインクリメントされます。伝送は、`chanMask`で定義され、疑似ランダムな順序で選択された一連のチャンネルで実行されます。

連続変調テストでは、指定された単一チャンネルを中心に、連続した疑似ランダム変調信号がデバイスで生成されます。デバイスをリセットすると、テストが停止されます。

連続波テストでは、指定された単一チャンネルを中心に、非変調トーンがデバイスで生成されます。デバイスをリセットすると、テスト・トーンが停止されます。

CCAを有効にしたパケット伝送テストでは、パケット伝送テストとまったく同じようにデバイスが設定されます。ただし、デバイスは各伝送の前にクリア・チャンネル・アセスメントを実行し、チャンネルがビジーの場合はパケットをアボートします。

 チャンネル番号は0~15で、IEEE 2.4GHzチャンネルの11~26に相当します。

 `stationId`は、SmartMesh IP Moteのバージョン1.4以上、SmartMesh IP Managerのバージョン1.3.0以上、SmartMesh WirelessHARTモートのバージョン1.1.2以上で使用できます。

構文

```
radiotest tx <testType> <chanMask> <power> [<stationId> <repeatCnt> {<pkLen><delay>...}]
```

パラメータ

パラメータ	説明
testType	開始する伝送テストの種類。pk = パケット、cm = 連続変調、cw = 連続波、pkcca = CCA付きパケット。
chanMask	テスト用のチャンネル(0~15)の16進数ビットマスク。ビット0がチャンネル0に相当します。連続波および連続変調テストでは、1つのチャンネルのみを有効化してください。
power	送信出力(dBm)。有効な値は、0および8です。
stationId	送信側の一意のステーションID(0~255)。受信側のステーションIDと一致しなければなりません。
repeatCnt	パケット・シーケンスを繰り返す回数(0 = 中断なし)。パケット伝送テストのみに適用されます。
pkLen	パケットの長さ(2~125バイト)
delay	伝送後の遅延(0~65535マイクロ秒)

例

チャンネル0、1(chMap = 0x03)で、出力送信電力 = 0dBm、ステーションID = 26を指定して、パケット・テストを開始します。50バイトのパケット、20ミリ秒の遅延、30バイトのパケット、20ミリ秒の遅延で、シーケンスを5回繰り返します。

```
radiotest tx pk 0x3 0 26 5 50 20000 30 20000
```

チャンネル0に対して、8dBmの出力送信電力で連続変調の伝送を開始します。

```
radiotest tx cm 0x1 8
```

チャンネル1に対して、8dBmの出力送信電力で連続波の伝送を開始します。

```
radiotest tx cw 0x2 8
```

3.17.3 radiotest rx

説明

radiotest rxコマンドは、無線を受信モードに切替えて、パケット受信に関する統計情報を収集します。ゼロ以外のステーションIDを指定する場合、送信側のステーションIDと一致しなければなりません。これは、同じ無線空間で複数のテストが実行されている場合にトラフィックを分離するために必要です。統計情報を表示するには、radiotest statコマンドを使用します。

⚠ チャンネル番号は0～15で、IEEE 2.4GHzチャンネルの11～26に相当します。

⚠ stationIdは、SmartMesh IP Moteのバージョン1.4以上、SmartMesh IP Managerのバージョン1.3.0以上、SmartMesh WirelessHARTモートのバージョン1.1.2以上で使用できます。

構文

```
radiotest rx <chanMask> <time> <stationId>
```

パラメータ

パラメータ	説明
chanMask	テスト用のチャンネル(0～15)の16進数ビットマスク。ビット0がチャンネル0に相当します。このコマンドで指定できるのは、単一チャンネルのみです。
time	受信テストの時間(秒)。0 = 停止なし
stationId	受信側の一意のID(1～255)。送信側のステーションIDと一致する必要があります。すべての送信側からパケットを受け取る場合、ステーションIDに0を使用します。

例

ステーションIDに26を使用して、チャンネル2で60秒間、デバイスを受信モードにします。

```
radiotest rx 0x4 60 26
```

3.17.4 radiotest stat

説明

`radiotest stat` コマンドは、前回の `radiotest rx` コマンドの実行中に収集されたパケットの受信に関する統計情報を表示します。このコマンドを使用できるのは、デバイスが無線テスト・モードになっているときだけです。

構文

```
radiotest stat
```

パラメータ

パラメータ	説明
-------	----

例

```
>radiotest stat
Radio Test Statistics
  OkCnt   : 0
  FailCnt : 0
```

3.18 reset

説明

ネットワーク内の指定されたエンティティ(モートまたはマネージャ)をリセットします。このコマンドにはユーザ権限が必要です。

構文

```
reset <entity>
```

パラメータ

パラメータ	説明
entity	<ul style="list-style-type: none">「system」を指定すると、マネージャがリセットされるため、ネットワーク全体もリセットされます。「mote」を指定する場合、モートIDまたはMACアドレスで指定できます。

例

```
reset mote 2
```

```
reset system
```

3.19 set acl

説明

set acl コマンドは、新しいACLエントリを追加するか、既存エントリを変更するために使用します。参加鍵は、1バイトあたり2文字の16進数です。これは永続的な変更です。エントリの最大数は1,200です。

構文

```
set acl mac=<macAddr> key=<joinKey>
```

パラメータ

パラメータ	説明
macAddr	モートのMACアドレス
joinKey	16バイト(32文字)の参加鍵


例

```
set acl mac=01-23-45-67-89-AB-CD-EF key=000102030405060708090A0B0C0D0E0F
```


3.20 set config

説明

Managerの設定パラメータをセットします。この変更は次のシステム開始時(リセットまたはパワー・サイクル後)に有効になります。これは永続的な変更です。ライセンスを必要とする機能には2回のリセットが必要です。1回はライセンスを有効にする(および設定変更を有効にする)ため、もう1回は設定を反映させるためです。

 licenseフィールドは、1.3.0以上のManagerでは廃止されています。モート数が32を上回るネットワークではライセンスは不要です。

構文

```
set config <param>=<value>
```

パラメータ

パラメータ	説明
param	<ul style="list-style-type: none"> ● <i>netid</i>: ネットワークID ● <i>txpower</i>: 無線送信電力 ● <i>frprofile</i>: フレーム・プロファイルID ● <i>maxmotes</i>: モート(APを含む)の最大数(1~101、マネージャの製品番号とインストールされたライセンスによって異なる) ● <i>basebw</i>: ベース帯域幅(0、または10以上) ● <i>dnfr_mult</i>: 下りフレーム乗数(1、2、4) ● <i>numparents</i>: 親の最小数(1~4) ● <i>cca</i>: CCA値(0、1、2、3) ● <i>channellist</i>: ホワイトリストに含まれ通信に使用されるチャンネルのビットマップ。その他はすべてブラックリストに含まれます。ビット0×0001がチャンネル0に相当し、ビット0×8000がチャンネル15に相当します(0 = 未使用、1 = 使用)。チャンネル数の制限については、SmartMesh IP User's GuideのChannel Blacklistingセクションを参照してください。 ● <i>autostart</i>: 廃止済み - 使用しないでください ● <i>locmode</i>: 予約済み ● <i>bbmode</i>: バックボーン・モード(0 = オフ、1 = 上りのみ、2 = 双方向) ● <i>bbsize</i>: バックボーンのフレーム・サイズ(bbmode = 1の場合、bbsize = 1、2、4、8。bbmode = 2の場合、bbsize = 2) ● <i>license</i>: ライセンス(16進数、例:00-12-34...) ● <i>pwdviewer</i>: ビューア・レベルのパスワード ● <i>pwduser</i>: ユーザ・レベルのパスワード ● <i>commonjoinkey</i>: 共通参加鍵 ● <i>ip6prefix</i>: IPv6アドレスのプレフィックス ● <i>ip6mask</i>: IPv6マスク ● <i>radiotest</i>: 無線テスト・モード(0 = オフ、1 = オン) ● <i>bwmult</i>: 値 / 100でオーバープロビジョニングを実施します(100~1000) ● <i>onechannel</i>: 単一チャンネル・ネットワークで使用するチャンネル(0~14、0xFF)
value	値(設定パラメータ表を参照)

例

```
set config netid=100
```

3.21 seti

説明

setiコマンドは、内部のINIパラメータを変更するために使用します。このコマンドは、アプリケーション・ノートの指示に従った高度な設定向けであり、superuserの権限を必要とします(suを参照)。

構文

```
seti ini <param>[= | ' ' ]<args>
```

パラメータ

パラメータ	説明
parameter	変更するパラメータ。マネージャの動作を変更して特定の結果を達成するためのパラメータ変更については、各種のアプリケーション・ノートを参照してください。
args	指定されたパラメータに渡す引数

例

```
seti ini iscascading 1
```

3.22 show

説明

各種のシステム・オブジェクトのステート情報を表示します。

構文

```
show <object> [args]
```

パラメータ

パラメータ	説明		
args	指定できる引数については、オブジェクト表を参照してください。		
object	オブジェクト	説明	引数
	acl	アクセス制御リスト	
	curconfig	現在使用中の設定パラメータ	
	config	永続的な設定パラメータ	
	mote	モートのステータス	-aまたは-l、<motelD macAddr *>を指定可能
	motever	モートのバージョン	<motelD macAddr>
	path	パス情報	
	stat	統計情報(信頼性、安定性など)	
	status	ネットワーク・ステータス	
	time	UTC(設定されている場合)およびASNでの現在の時刻	
	trace	アクティブ・トレース	
	ver	Managerのバージョン情報	
	memory	マネージャのメモリ設定。Type 0 = プライマリRAM、type 1= 外部RAM、type 2 = 2kのセカンダリ内蔵RAM。このコマンドは、Managerのバージョン1.2.0以上で使用可能です。	

例

```
show mote 2
```

3.22.1 show acl

```
> show acl
ACL:

MAC: 00-17-0D-00-00-38-FF-FF

MAC: 00-17-0D-00-00-37-6E-A1
```

このコマンドは、マネージャのアクセス制御リストにホワイトリストとして現在含まれているモートを表示します。

3.22.2 show configおよびshow curconfig

```
> show config
netid = 302
txpower = 8
frprofile = 1
maxmotes = 33
basebw = 9000
dnfr_mult = 1
numparents = 2
cca = 0
channellist = 00:00:7f:ff
autostart = 1
locmode = 0
bbmode = 0
bbsize = 1
license = 00:00:00:00:00:00:00:00:00:00:00:00
ip6prefix = fe:80:00:00:00:00:00:00:00:00:00:00:00:00:00:00
ip6mask = ff:ff:ff:ff:ff:ff:ff:00:00:00:00:00:00:00:00:00
radiotest = 0
bwmult = 300
onechannel = 255
```

どちらのコマンドでも同じデータ構造が返されます。`show config`コマンドは、次回の起動後に使用される永続的なパラメータを表示します。`show curconfig`コマンドは現在使用されているパラメータを表示します。各パラメータの説明については、[set config](#)を参照してください。

3.22.3 show mac

```
> show mac
MACs:
----
```

予約済みです。使用しないでください。これは、使用が許可されていない内部コマンドです。

3.22.4 show mote

```

> show mote 2
Mote #2, mac: 00-17-0D-00-00-38-16-6B
  State: Oper, Hops: 1.1, Uptime: 0-00:30:39, Age: 1
  Regular. Route/TplgRoute.
  Power Cost: Max 65534, FullTx 110, FullRx 65
  Capacity links: 200, neighbours 31
  Number of neighbors (parents, descendants): 8 (2, 15)
  Bandwidth total / mote exist (requested): 90 / 954 (987)
    Links total / mote exist (requested): 64.0 / 6.0 (5.8)
    Link Utilization : 1.0
  Number of total TX links (exist / extra): 64 / 0
  Number of links : 130
    Compressed : 5
  Upstream tx/rx : 122 (64/58) (Rx10=58.0)
  Downstream rx : 3
  Neighbors:
    -> # 1 Q: 71% RSSI: -59/0
    -> # 7 Q: 55% RSSI: -55/-49
    <- # 9 Q: 97% RSSI: -39/-40
    <- #15 Q: 94% RSSI: -37/-39
    <- #17 Q: 87% RSSI: -40/-43
    <- #18 Q: 91% RSSI: -44/-44
    <- #20 Q: 29% RSSI: -76/-74
    <- #24 Q: 29% RSSI: -76/-75

```

フィールドの説明

- Mote: モートの短縮アドレス
- mac: モートのEUI-64
- State: マネージャによって割り当てられた現在のモート・ステート。**Idle**、**Negot1~2**、**Conn1~5**、**Oper**、**Lost**のいずれか
- Hops: このモートの上りパケットが経由する平均ホップ数。APでの受信時のTTLで測定
- Uptime: 前回のモート・ステート変更以降の時間
- Age: マネージャが前回このモートから上りパケットを受信してから経過した時間(秒)
- Power type: 電力(powerSrcInfoパラメータ = 0xffffのmaxStCurrent)、RegularまたはLow Power(ルーティングに必要な電力よりmaxStCurrentが低い場合)
- Route type reported by the mote: RouteまたはNo-route(routingModeパラメータより)
- Route type as assigned by the manager: TplgRouteまたはTplgNo-Route
- Power Cost: 参加処理中のモートから返されるpowerSrcInfoパラメータ
- Number of neighbors: 最初のエントリは、親の近隣モート数 + 子の近隣モート数 = 近隣モート数。カッコ内の最初のエントリが親の近隣モート数、2番目のエントリが子孫の近隣モート数。ここから、子の近隣モート数 = 近隣モート数 - 親の近隣モート数
- Bandwidth(ミリ秒/パケット): 帯域幅セクションは上りトラフィックおよび上りリンクのみを対象としています。totalの値は、モート自体のトラフィックと下位トラフィックを合わせた値を示します。mote existの値は、モート自体が処

理する帯域幅を示し、カッコ内の*requested*の値はサービス・リクエストを介してモートに要求された帯域幅です。ここで示される値が低いほど、帯域幅は広くなります。一般に、*mote exist*値は*requested*値よりも少しだけ低くなります。これは、マネージャが追加できない、端数リンクによって生じる、丸め処理から生じるためです。

- Links (リンク/スーパーフレーム): *total*は、モート自体のトラフィックと下位トラフィックをサポートするために追加されたリンク数の合計を示します。*mote exist*の値は、このモートのサービス・リクエスト専用追加されたリンクの数を示し、カッコ内の*requested*の値は、トラフィックをサポートできるリンクの浮動小数点です。したがって、*mote exist*値は、*requested*値を最も近い整数に切り上げた値になります。
- Link Utilization: 0から1の間の数値で、マネージャから見て、モートがどれだけプロビジョニングの限界に近づいているかを示します。例えば、マネージャが予想するモートの送信速度が1パケット/秒で、プロビジョニングが3xで、モートの上りリンクが毎秒10回割り当てられた場合、使用率は、 $1 * 3 / 10 = 0.3$ になります。この値は、低トラフィックのネットワークでリンク数を低く保つ目的で、トラフィック要件をモートの親に拡張するために使用します。
- Number of total TX links: *exist*の値は、現在割り当てられている上りリンクの数を示し、*extra*の値は次の最適化サイクルで削除される数を示します。
- Number of links: 上りリンクだけでなくすべてのスロットフレームを通じた合計リンク数。その下に表示される3行の合計値と一致します。
- Compressed: アドバタイズメント、参加リスニング、検出に使用される圧縮リンクの数。これらはモートの参加処理中に割り当てられ、変更されないため、圧縮形式で保存することでマネージャのメモリを節約できます。
- Upstream: 合計数、(送信数/受信数、整数)、(必要な受信数、端数処理なし)
- Downstream: 親からの受信リンク数。1ホップのモートが、APからの下り受信リンクを2つ(ブロードキャストとマルチキャスト)持つ場合があります。
- Neighbors: 関係 (-> 親方向、<- 子方向、- 検出済み)、近隣モートID、パス品質 (パス安定性が測定されるまでは30%または74%)、このモートによって測定されたRSSI、近隣モートによって測定されたRSSI

「ルーティング・タイプ」は、モートまたはマネージャに対して設定できます。モートまたはマネージャで、特定のモートを非ルーティングとして宣言している場合、このモートには子モートやアドバタイズメント・リンクは割り当てられません。

3.22.5 show motever

```
> show motever 2
Mote #2, mac: 00-17-0D-00-00-38-06-28. Ver SW: 1.1.0-29 HW: 1.33
Mote #2, mac: 00-17-0D-00-00-38-06-28. Vendor: 1. App ID/Ver: 1/1.1.0-29
```

このコマンドは、ネットワーク内のモートに関するすべてのバージョン情報を表示します。このコマンドは無線でモートに送信されるため、即時には応答が返されません。

表示される応答は2種類あります。1番目がネットワークスタックのソフトウェアバージョンで、2番目はベンダーIDとアプリケーション層のソフトウェアバージョンです。

3.22.6 show path

```
> show path
MoteID * {NeighborId:{P|C|-}:Quality(Src)}...
  P - neighbor is parent, C - neighbor is child, - - no parent relationship
  Quality - quality of path (100-best, 0-worst)
  Src - source for quality calculation (s-statistics, r-RSSI)
Network Average Quality: 78
Paths:
1 * 2:C:97(s) 3:C:74(r)
2 * 1:P:97(s) 3:C:74(r)
3 * 1:P:74(r) 2:P:74(r)
```

このコマンドは、マネージャが現在認識しているネットワーク内のすべてのパスを、アクティブ・リンクを持つかどうかに関係なく表示します。それぞれのモートに対して1行表示されます。2行目を見ると、モートID2に関連付けられたパスが表示されています。上の例ではPaths以下が示す行は、モート1が親(P)であり、この子モート2から親モート1へのパス品質は97%で、通信成功／失敗の統計情報(s)に基づいて評価されたことを示しています。パス品質はパスの「安定性」とも呼ばれています。更にレポートの例では、モート3が子(C)で、このパスの安定性はまだ評価されていないため、RSSI(r)を使用して品質が推定されたことも分かります。

3.22.7 show stat

```
> show stat

Manager Statistics -----
established connections: 1
dropped connections    : 0
transmit OK           : 0
transmit error        : 0
transmit repeat       : 0
receive OK            : 0
receive error         : 0
acknowledge delay avrg : 0 msec
acknowledge delay max  : 0 msec

Network Statistics -----
reliability: 100% (Arrived/Lost: 7217/0)
stability: 99% (Transmit/Fails: 14304/204)
latency: 200 msec

Motes Statistics -----
Mote   Received   Lost  Reliability Latency Hops
#2      257         0   100%       580   1.3
#3      249         0   100%       450   1.0
#4     6463         0   100%       250   2.4
#5      248         0   100%       150   1.2
```

このコマンドは、モートおよびネットワークの統計情報を表示します。表示されるカウンタのインクリメントおよび平均計算の対象期間は、ネットワークの存続期間全体か、`exec clearStat` CLIコマンドまたは`clearStatistics` APIコマンドを使用して自発的にクリアされた後の期間のどちらかになります。

この例では、モート4がどのモートよりも頻繁にデータを生成しています。モート4は2.4ホップの位置にあり、すべてのパケットは別のモートを介して転送されます。モート3は1.0ホップの位置にあるため、すべてのパケットは直接APIに送信されます（これを「単一親」モートと呼ぶ）。モート2および5は、トラフィックの一部をモート3経由で転送しています。

数値の分析

reliability: Arrived = 7217 = 257 + 249 + 6463 + 248、Lost = 0。到着済み／紛失カウンタと信頼性は、マネージャでリアルタイムで維持されるため、常に最新かつ正確です。

stability: Transmit = 14304 = 7217(到着したパケット数) + 204(失敗し、再送信の必要なパケット数) + 6463(モート4から送信され、モート2、3、5による転送が必要なパケット数) + 50(モート3によって転送されたモート5のパケット数の20%) + 77(モート3によって転送されたモート2のパケット数の30%) + 293(モート4からモート2および5を経由し、モート3を介した転送が必要なパケット数)。安定性カウンタと平均安定性は、モート健全性レポートに基づいて計算されるため、信頼性統計には最大15分の遅れが発生します。

マネージャの統計情報

- *established connections*: 確立されたAPI接続の数
- *dropped connections*: 切断されたAPI接続の数
- *transmit OK*: クライアント・アプリケーションによってアクノリッジされたAPIパケット(通知など)の数
- *transmit error*: アクノリッジされなかったAPIパケットの数
- *transmit repeat*: リトライされたAPIパケットの数(複数回リトライされた場合、transmit errorより大きい値になります)
- *receive OK*: マネージャがアクノリッジしたAPIコマンド
- *receive error*: マネージャがアクノリッジできなかったAPIコマンド(フレーミングなどのエラーによる)
- *acknowledge delay avrg*: APIパケットに対する応答を受信するまでの平均時間
- *acknowledge delay max*: APIパケットに対する応答を受信するまでの最大時間

3.22.8 show status

```
> show status
S/N:      00-17-0D-00-00-38-07-13
MAC:      00-17-0D-00-00-38-07-13
IP6:      FE:80:00:00:00:00:00:00:17:0D:00:00:38:07:13
HW ver: 1.33
NetworkID: 302
Base bandwidth (min links): 9000 (1)
Frame size Up/Down 259/259. Number of working timeslots 256.
Available channels      15
Base timeslot(TS0)     235
Network mode is        Steady State
Downstream frame       259 timeslots
Optimization is        On
Advertisement is        On
AP output CTS is       Ready
Backbone is            off
Location is            off
API is                 Connected
License                 00:00:00:00:00:00:00:00:00:00:00:00
Network                 Regular Mode
Total saved links      28
One channel mode:      OFF.
```

このコマンドは、ネットワークに関するグローバルな固定情報と、ネットワーク存続時間の経過と共に変化する可能性のあるステータスを表示します。

- S/N: マネージャのEUI-64
- MAC: マネージャのロング・アドレス(EUI-64とは一致しない場合あり)
- IP6: IPv6アドレス。ここではリンクローカルのプレフィックスを使用
- HW ver: ハードウェアのバージョン
- NetworkID: 0~65535
- Base Bandwidth: 各モートをターゲットにしたグローバル帯域幅(ミリ秒)
- Frame size: ランダム化された256~284スロット。すべての値を取りうるわけではありません。動作スロットは常に256で、アクティブ・リンクを割り当て可能なスロットフレームのスロット数を表します
- Available Channels: 15(ブラックリストに追加された場合を除く)
- Base Timeslot: 複数のネットワークが空間を共有している場合、性能を上げるために、0~255のランダムな数ですべてのリンクをオフセットします
- Downstream Frame: 設定パラメータ*dnfr_mult*で指定されます。上記のフレーム・サイズを1倍、2倍、または4倍にした数値です。
- Optimization: デフォルトではオンです。
- Advertisement: デフォルトではオンです。`exec setAdv on`または`exec setAdv off`を介して、有効化または無効化することができます
- AP output: APのパケット受け入れ準備完了
- Backbone: デフォルトではオフです。*bbmode*設定パラメータの指定により、有効化できます
- Location: 予約済み
- API: APIに接続されたデバイスがあるかどうかを示します
- License: 鍵

- Network: モード
- Total saved links: マネージャが保存するリンクの数。メモリの制約により500までに制限されます。
- One channel mode: テストで使⽤します。チャンネルxでのチャンネル・テストを有効にする場合、`set config onechannel x`を実行します。無効にする場合、`set config onechannel 255`を実行します。

3.22.9 show time

```
> show time
Current UTC:      1025666459.752 sec
Current second:  1260 sec
Current OS time: 1260438 ticks
Current ASN:     173759 slots
```

このコマンドは、UTC、システム時刻(ミリ秒tick)、現在のネットワークのASNという3つの基準で時刻を表示します。

3.22.10 show trace

```
> show trace
   io
   motest
```

このコマンドは現在アクティブなマネージャのトレースを表示します。この例では、入出力とモート・ステートのトレースがアクティブになっています。

3.22.11 show ver

```
> show ver  
SmartMesh IP Manager ver 1.1.0.25.
```

このコマンドは、マネージャ上で実行されているコードのバージョンを表示します。

3.23 showi

説明

各種の内部システム・オブジェクトのステート情報を表示します。このコマンドは、アナログ・デバイセズでのマネージャ開発中のデバッグを目的としています（顧客やシステム・インテグレータによる使用は想定されていません）。

構文

```
showi <object>
```

パラメータ

パラメータ	説明
object	<i>ini</i> : システムINIパラメータ(変更不可) <i>linkmap</i> : 割り当てられたリンクのビットマップ <i>loopfinder</i> : ループ・ファインダ表 <i>memory</i> : メモリ・プール情報 <i>stacks</i> : タスク・スタックの使用状況 <i>toplock</i> : トポロジ・ロックの統計情報

例

```
showi ini
```

3.24 sm

説明

ネットワーク内のモートを表示します。

構文

```
sm [-v]
```

パラメータ

パラメータ	説明
-v	詳細表示

例

```
> sm
      MAC                MoteId State Nbrs Links Joins Age StateTime
00-17-0D-00-00-38-07-13    1 Oper  2   25  1  0  0-00:45:39
00-17-0D-00-00-38-06-28    2 Oper  2   15  1  1  0-00:45:21
00-17-0D-00-00-38-04-2F    3 Oper  2   15  2  1  0-00:28:49
Number of motes (max 33): Total 3, Live 3, Joining 0
```

このコマンドは、ネットワーク内に現在ある(または過去にあった)モートをすべて表示します。

- MAC: モートのEUI-64
- MoteID: マネージャによってこのモートに割り当てられた短縮アドレス。MoteID 1は常にAPです。
- State: 各モートの現在の状態(Nego、Conn、Oper、Lost)
- Nbrs: このモートがアクティブ・リンクを持つ近隣モートの数
- Links: 総リンク数(圧縮および標準)
- Joins: モートがOperational状態まで進んだ回数
- Age: マネージャがこのモートから最後のパケットを受信してから経過した秒数。
- StateTime: モートが現在の状態まで進んでからの時間(d-hh:mm:ss)。モートの状態がOperationalの場合、このモートがどれだけ長くネットワークに参加しているかがStateTimeに表示されます

3.25 su

説明

suコマンドは、現在のユーザを変更するために使用します。このコマンドは通常、viewerからuserの認証情報に変更するために使用します。

構文

```
su user
```

パラメータ

パラメータ	説明
password	権限レベルのパスワード。出荷時のデフォルト・パスワードは、user用が「user」で、viewer用が「viewer」です。 注:パスワードの変更には、set configコマンドを使用します。 setiコマンドを有効にするには、superuserのパスワードである「becareful」を使用します。

例

```
su user
```

3.26 trace

説明

トレースを有効または無効にします。このコマンドは評価およびデバッグを目的とするもので、*user*レベルでログインしている場合のみ使用できます。ユーザが問題を診断中である場合を除いて、現場のネットワークでトレースを実行したままにしないでください。一部の予約済みトレースには、特別なマネージャ設定が必要です。引数なしで実行すると、すべてのトレースの現在のステートが返されます。

⚠ 特にネットワークが大規模な場合、一部のトレースはマネージャ性能に悪影響を与える量のアクティビティを生成し、デフォルトのCLIボーレートでは出力が損なわれるおそれがあります。このような場合、トレースが「このCLIボーレートに対してサポートされていない」ことを示すエラーが返されます。CLIボーレートを変更せずに、これらのトレースを有効にするには、事前に以下のコマンドを実行します。この設定は、リセット時にクリアされます。

```
> su becareful
> debug hscli on
```

構文

```
trace [<module> <mode>]
```

パラメータ

パラメータ	説明
module	<i>bw</i> : 帯域幅／サービス <i>glbcmd</i> : グローバル・ユニキャスト・コマンド <i>io</i> : マネージャの受信／送信トラフィック(予約済み) <i>iodata</i> : アプリケーションの受信／送信トラフィック(予約済み) <i>loc</i> : 予約済み <i>link</i> : リンクの作成／削除 <i>monitor</i> : 監視イベント <i>motest</i> : モート・ステートの変更 <i>netmode</i> : ネットワーク・モード <i>opt</i> : 最適化 <i>rawio_enc</i> : 暗号化入出力パケット(予約済み) <i>power</i> : 電力コストの計算 <i>route</i> : モート・ルートの計算 <i>spl_task</i> : シリアル・イベント(予約済み) <i>spl_ack</i> : シリアルACKのタイムアウト(予約済み) <i>stats</i> : 統計情報の計算 <i>timeout</i> : 信頼できるメッセージのタイムアウト <i>tplglock</i> : トポロジ(予約済み) <i>fa</i> : フラッシュ・アレイ <i>loop</i> : ループ・ファインダ

パラメータ	説明
	<i>all</i> : すべてのトレースを有効化
mode	<i>on</i> : 指定したモジュールのトレースを有効化 <i>off</i> : 指定したモジュールのトレースを無効化

例

```
trace motest on
```

3.26.1 trace bw

```
>trace bw on  
3031355 : New Bandwidth Request. BW: 30000 msec. Links 1  
3060627 : New Bandwidth. #2 BW: 5631 msec. Links (req/real/extra) 1/1/0
```

このトレースは、モートから要求されたすべての新しいサービスを表示します。マネージャが新しいサービス・リクエストを受信した時点で1回出力され、変更が完了したときにもう1回出力されます。この例では、5631ミリ秒ごとにパケットを送信できるだけの帯域幅がモートに残されています。

3.26.2 trace fa

```
> trace fa on
```

```
1790561 : Flash Array. Save Data. Record #22 File 2mngarr1.bin. Offset 792
```

このトレースは、マネージャとフラッシュ・メモリ間のすべての相互作用を出力します。ファイルとメモリの場所が指定されます。

3.26.3 trace glbcmd

```
> trace glbcmd on

2124787 : GlbCmd Send Timeout: 30 sec
2124788 : GlbCmd 1 Request for Send
2124789 : GlbCmd New Command 1 Execute Time: 1 sec
2124790 : GlbCmd 1 Send command to mote #1
2124799 : GlbCmd NETID. NetId: 200
Start Global Unicast Command exchNetId
2124829 : GlbCmd 1 Received ack from mote #1
2154788 : GlbCmd 1 Request for Send
2154789 : GlbCmd 1 Send command to mote #2
2154798 : GlbCmd NETID. NetId: 200
2155720 : GlbCmd 1 Received ack from mote #2
2184788 : GlbCmd 1 Request for Send
2184790 : GlbCmd 1 Send command to mote #3
2184798 : GlbCmd NETID. NetId: 200
2185829 : GlbCmd 1 Received ack from mote #3
2185830 : GlbCmd 1 finishes
```

このトレースは、グローバルな(すべてのモートに対する)下りコマンドの送信プロセスを表示します。この例では、マネージャが、ネットワーク内のAPおよび2つのモートとNetIdを交換しています。グローバル・コマンドは信頼できるユニキャストとして送信されるため、それぞれのモートが応答してからコマンドが終了します。

3.26.4 trace io (予約済み)

```
> trace io on

732985 : TX ie=32:2e mesh=1223 ttl=127 asn=35486 gr=2 dst=10 rt=r8 r=4:7:8:9:10 tr=r:req:10; sec=s
ofs=157 nc=0 ie=fe:00 IP=7e77 UDP=f7 sP=f0b0 dP=f0b0; cmd=GET_REQ_SRV len=3 type=0;
```

この「入出力」トレースは、マネージャが送受信するトラフィックを表示します。アプリケーション・データは表示しません。解析されたネットおよびIPヘッダと、ペイロード情報が含まれます。

- OSタイムスタンプ(マネージャが起動されてからの時間、ミリ秒)
- TXはマネージャによって送信される下りパケットで、RXはマネージャによって受信される上りパケットです
- ie: 15.4情報エレメント・ディスクリプタ、ここではヘッダIE
- mesh: メッシュ・ディスパッチャ
- TTL: デクリメントするメッシュTTLカウンタ。受信(RX)パケットの場合、ホップを127ttlとして計算
- ASN: パケットが生成された時刻のASN
- gr: パケットの移動グラフ。通常、下りが2で上りが1になります
- dst: 宛先モート
- rt: ルート・タイプ、ここでは最大8ホップのソース・ルート
- r: ホップごとの明確なソース・ルート。マネージャや最初のホップは含みません
- tr: 伝送ヘッダ。ここでは、信頼できるリクエストでカウンタ = 10
- sec: セキュリティ・セッション・タイプ。参加鍵を使用する最初の参加パケットの場合は「j」となり、その後のセッション鍵を使用したパケットは「s」になります。
- ofs: 暗号化オフセット
- nc: 臨時カウンタ(現在はバグのため、常にゼロが表示されます)
- ie: 15.4のIEディスクリプタ、ここではターミネータ
- IP: 圧縮IPヘッダ
- UDP: 圧縮UDPヘッダ
- sP: 送信元ポート
- dP: 宛先ポート
- cmd: ネットワーク・スタック・コマンド
- len: コマンドの長さ

この例では、マネージャがモート10にサービス・レポートを要求しています。

3.26.5 trace iodata (予約済み)

```
> trace iodata on
3508141 : RX ie=32:1e mesh=4002 ttl=126 asn=24959 gr=1 src=3 rt=g tr=u:req:0; sec=s ofs=23 nc=208
ie=fe:00 IP=7e77 UDP=f7 sP=f0b9 dP=f0b9;
```

この「入出力データ」トレースは、マネージャへのアプリケーション・データ・トラフィックを表示します。コマンド・トラフィックは表示しません。この例では、マネージャがモート3からデータ・パケットを受信しました。情報の解析方法については、[trace io](#)を参照してください。

3.26.6 trace link

```
> trace link on

3628489 : Link Add      #2 -> #1 (1:211:0)
3628491 : Link Add      #2 -> #1 (1:83:6)
3628493 : Link Add      #2 -> #1 (1:243:5)
3628495 : Link Add      #2 -> #1 (1:179:2)
3628498 : Link Add      #2 -> #1 (1:115:7)
```

このトレースは、マネージャによる、リンク追加および削除に関するすべての決定を表示します。この後に、ポートに対する実際の変更コマンドが続きます。これらは、`trace io`がオンになっている場合に表示されます。リンクに対するポートペアの後に、リンクのスロットフレーム:スロット:オフセットが表示されます。

3.26.7 trace loop

```
> trace loop on

1657824 : Loopfinder. Connect 2 -> 6
1660018 : Loopfinder. Disconnect 2 -> 3
1660019 : Loopfinder. Recalculation
```

このトレースは、ループ・チェッカーに関連するマネージャの個別操作を表示します。2つの新しいモートが親子関係で接続された場合、新しいエントリは簡単に親および上位テーブルに追加することができます。2つのモートが切断された場合、表全体を再計算する必要があります。

3.26.8 trace monitor

```
> trace monitor on

3722406 : Monitor event 'waitq'
3722407 : Monitor event 'waitq' finish. Res 0
3723409 : Monitor event 'waitq'
3723409 : Monitor event 'waitq' finish. Res 0
```

このトレースは、マネージャのコード内で発生したすべてのイベントを表示します。このトレースはデバッグのみを目的としたものです。

3.26.9 trace motest

```
> trace motest on
84410 : Mote #2 State: Idle  -> Negot1
86139 : Mote #2 State: Negot1 -> Negot2
88329 : Mote #2 State: Negot2 -> Conn1
90185 : Mote #2 State: Conn1  -> Conn2
91613 : Mote #2 State: Conn2  -> Conn3
93468 : Mote #2 State: Conn3  -> Oper
2731420 : Mote #2 State: Oper -> Lost
```

このトレースは、OSタイムスタンプ(ミリ秒)、モートID、すべてのステート変更を出力します。モートが**Idle**から**Operational**になるまでの参加ステートの遷移が表示されます。一定時間後にモートがネットワークから遮断されると、**Lost**ステートへの遷移が出力されます。

3.26.10 trace netmode

```
> trace netmode on

224235 : Check Steady State Start
224238 : Check Steady State. Check.
284243 : Check Steady State. Check.
344244 : Check Steady State. Check.
404245 : Check Steady State. Check.
464247 : Check Steady State. Check.
524248 : Check Steady State. Check.
584249 : Check Steady State. Check.
644250 : Check Steady State. Check.
704251 : Check Steady State. Check.
764253 : Check Steady State. Check.
824254 : Check Steady State. Check.
884255 : Check Steady State. Check.
884256 : Check Steady State. Set Steady State.
884257 : Check Steady State Stop
```

このトレースは、最後のモードが参加してから10分以上経過したことをマネージャがチェックしていることを表示します。マネージャは1分ごとにチェックを実行し、10分経過すると、ネットワーク・ステートを**Building**から**Steady State**に変更します。

3.26.11 trace opt

```
> trace opt on  
  
884258 : Optimization Start  
884262 : Optimization Delete  
884272 : TPLGBLD_OPTDEL for #2
```

このトレースは、トポロジ最適化プロセスのすべてのステップを表示します。この例では、ネットワーク内の唯一のモードに対して最適化削除サイクルが実行されています。

3.26.12 trace power

```
> trace power on
```

```
550296 : Power Cost Calc. Try Add 1 RX-links. Max Power 65534
```

```
550316 : Pwer Cost Calc: TxCost 110, RxCost 65, NumTx: 4, NumRx 4, PowerCost 348
```

このトレースは、モードに新しいリンクを追加しようとするときにマネージャが行う計算を表示します。各モードは、参加プロセス中に最大許容電流 $maxStCur$ を報告します。新しいリンクによって合計電力が $maxStCur$ を超える場合、マネージャはこのリンクを追加しません。

3.26.13 trace rawio_enc(予約済み)

```
> trace rawio_enc on

171656 : SecRX len=52
  000 : 27 32 32 1e 40 02 7e 5c 10 01 00 02 00 17 9e 95 2b 84 e5 fe
  020 : 00 7e 77 f7 99 00 00 37 eb f7 00 e0 20 ba d6 e8 29 87 67 1e
  040 : 27 1a ad 64 36 c3 4c 2e af ce fa 6a
```

このトレースは、マネージャがパケットを送受信するたびに、すべての未処理バイトを暗号化形式で表示します。パケットは1行に20バイトずつ出力されます。

3.26.14 trace route

```
> trace route on  
1083138 : ROUTE. New #5 Hops 3. 3-5-0
```

このトレースは、マネージャがモートに対する下りソース・ルートを計算する過程を表示します。この例でのソース・ルートは宛先がモート5で、AP-3-5を経由します。ルートの終了を示すため、常に末尾にゼロが付加されます。

3.26.15 trace spl_taskおよびtrace spl_ack(予約済み)

```
> trace spl_task on
> trace spl_ack on

3138150 : SPL_TASK: RX type=22,REQ RLBL , seqNo=1 retNo=0
3138152 : SPL_TASK: Packet received. Len=8
3138153 : SPL_TASK: Wait Packet
3138154 : SPL_TASK: Packet Type 22
3138161 : SPL_TASK: Command Type 22 Result 0
3138163 : SPL_TASK: TX type=22,ACK RLBL , seqNo=1 retNo=0
3138177 : SPL_TASK: apinotif_data Res 0
3138179 : SPL_TASK: TX type=20,REQ RLBL , seqNo=1 retNo=0
3138204 : SPL_TASK: RX type=20,ACK RLBL , seqNo=1 retNo=0
3138205 : SPL_TASK: Packet received. Len=5
3138206 : SPL_TASK: Acknowledge timeout 27 msec
3138209 : 3138221 : SPL_TASK: apinotif_data Res 0
SPL_TASK: Acknowledge Received
```

このトレースは、マネージャと外部アプリケーションとの間のシリアル・プロトコル(spl)による相互作用を表示します。spl_taskトレースは下り(TX)メッセージを表示し、spl_ackトレースは上り(RX)レスポンスを表示します。上の例では、マネージャとStargazerアプリケーションとの間で交換された最初のパケットが表示されています。

3.26.16 trace stats

```
> trace stats on

1383718 : STAT: #5: ASN Rx/Tx = 59692/59686, latency = 43, hops = 2
1383719 : STAT: new average hops10=20 latency/10=23
```

このトレースがアクティブになっている場合、マネージャが受信したすべてのパケットについて、ホップ数と遅延が出力されます。

- OSタイムスタンプ(ミリ秒)
- ASN Rx/Tx: マネージャがパケットを受信(Rx)した時間、および、当該パケットがモートによって生成(Tx)された時間またはモートがsendTo APIコマンド使用でパケットを受信した時間
- latency (ms): スロット数(Rx ASNとTx ASNの差分) * 7.25ミリ秒(スロット長)。
- hops: このパケットが経由したホップ数。
- new average: ホップ数と遅延の新しい平均値が次の行に出力されます。Hops10の単位は0.1ホップで、latency/10の単位は10ミリ秒であるため、この例に出力された平均値は、2.0ホップと230ミリ秒を表します。

3.26.17 trace timeout

```
>trace timeout on  
224238 : Reliable Timeout Mote: #2 Pkt:Act RetrNum: 1 Timout 180
```

この「信頼できるタイムアウト」トレースは、信頼できる下りパケットをすべて表示します。

- OSタイムスタンプ(ミリ秒)
- Mote: パケットの送信先
- Pkt: パケット・タイプ(この例では「アクティブ化」パケット)
- RetrNum: リトライ回数(この例では1回目)
- Timout: 次のリトライまでのタイムアウト時間(秒)


3.26.18 trace tplglock(予約済み)

```
> trace tplglock on

1261676 : tpls_shortLock()
1261678 : tpls_shortUnlock()
1262023 : tpls_shortLock()
1262026 : tpls_shortUnlock()
1262843 : tpls_shortLock()
```

このトレースは、マネージャによるネットワーク・トポロジのロックおよびロック解除を表示します。トポロジのロック／アンロック間、ルートまたは帯域幅を変更することはできません。これらのイベントは頻繁に発生し、ロックを表示する理由もないため、通常は内部開発中のマネージャ・テストでのみ使用されます。

商標

Eterna、Mote-on-Chip、SmartMesh IPは、Dust Networks, Incの商標です。Dust Networksロゴ、Dust、Dust Networks、SmartMeshは、Dust Networks, Incの登録商標です。LT、LTC、LTM、は、アナログ・デバイセズの登録商標です。第三者のブランド名および製品名は各社の商標であり、情報提供のみを目的として使用されています。

著作権

本書は、米国著作権法、国際著作権法、その他の知的財産法および産業財産法によって保護されています。本書はアナログ・デバイセズおよびその実施許諾者によって専有されており、制限付きライセンスに従って配布されます。アナログ・デバイセズの書面による事前の認可なく、本書の全部または一部を使用、複製、変更、逆アSEMBル、逆コンパイル、リバース・エンジニアリング、配布、再配布することは、その形式、手段にかかわらず禁じられています。

制限付き権利:米国政府による使用、複製、開示は、FAR 52.227-14(g)(2)(6/87)およびFAR 52.227-19(6/87)、またはDFAR 252.227-7015(b)(6/95)およびDFAR 227.7202-3(a)ならびにこれに準ずる法律および規制と後継の法律および規制に規定された制限の対象となります。

免責事項

本書は現状のまま提供され、明示、暗示を問わず一切の保証を行なわないものとします。かかる保証には、特定目的に対する商品性または適合性の黙示的保証が含まれますが、これに限定されません。

本書には技術的な誤りやその他の間違いが含まれる場合があります。訂正と改善は、新しいバージョンの文書に取り入れられる可能性があります。

アナログ・デバイセズは、製品やサービスの適用または使用により発生する責任を負いかねます。また、間接的あるいは偶発的損害を含むがそれに限定されない、いかなる責任も負わないものとします。

アナログ・デバイセズの製品は、誤動作がユーザの深刻な人身傷害につながると合理的に予想できる生命維持装置、デバイス、またはその他のシステムでの使用、またはその機能不全により生命維持装置またはシステムの故障あるいはその安全性や有効性に影響すると合理的に予想できる生命維持装置またはシステムの重要な部品としての使用を目的として設計されていません。このような用途での使用を目的としてこれらの製品を使用または販売しているアナログ・デバイセズの顧客は、顧客自身の責任でそれを行い、このような意図しないまたは不正な使用に関連する人身傷害または死亡に直接または間接的に起因するすべての主張、費用、損害、支出、および妥当な額の弁護士費用、また、かかるクレームでアナログ・デバイセズに該当製品の設計または製造に関わる過失があったと主張される場合でも、これを完全に補償し、アナログ・デバイセズとその役員、従業員、子会社、関連会社、および販売代理店に何ら損害を与えないことに同意するものとします。

アナログ・デバイセズは、いつでも製品またはサービスに対する修正、変更、拡張、改良、その他の変更を行う権利を保有し、製品またはサービスを予告なく中止する権利を有します。顧客は、発注の前に最新の関連情報を入手し、その情報が最新で完全であることを確認する必要があります。すべての製品は、注文承諾時または販売時に提供される、販売に関するDust Networkの契約条件に従い販売されます。

アナログ・デバイセズは、アナログ・デバイセズの製品またはサービスが使用される組み合わせ、マシン、またはプロセスに関連するアナログ・デバイセズの特許、著作権、マスクワーク権、その他のアナログ・デバイセズの知的所有権に従って、明示か黙示かにかかわらず、ライセンスが付与されることを保証または主張するものではありません。第三者の製品またはサービスに関してアナログ・デバイセズが公開した情報は、その製品またはサービスを使用するためのアナログ・デバイセズからのライセンス提供、あるいはその保証または推奨を意味するものではありません。このような情報を使用する場合、第三者の特許または他の知的所有権に従って第三者からのライセンスが必要になるか、またはアナログ・デバイセズの特許または他の知的所有権に従ってアナログ・デバイセズからのライセンスが必要になります。

Dust Networks, Incは、アナログ・デバイセズの完全所有子会社です。

© Analog Devices, Inc. 2012-2016 All Rights Reserved.