

PG-FP6

R20AN0573JJ0100

Rev.1.00

2020.06.26

通信コマンドを用いた活用法

要旨

本書は、フラッシュプログラマ PG-FP6 を用いたシリアル通信によるコマンド制御について紹介するものです。通信コマンドを使用することで、自動制御が可能です。

対象デバイス

RA ファミリ

RL78 ファミリ

RX ファミリ

RH850 ファミリ

Renesas Synergy™ マイクロコントローラ

RE ファミリ

パワーマネジメント（電池管理 IC）

Renesas USB Power Delivery ファミリ（C30 グループ）

モータドライバ/アクチュエータドライバ IC（モータ制御用 IC）

SuperH ファミリ

V850 ファミリ

78K ファミリ

R8C ファミリ

目次

1. 概要	3
2. シリアル通信によるコマンド制御までの基本手順	3
2.1 システムの接続	4
2.2 ターゲットシステムの接続	4
2.3 設定ファイルの作成	4
2.4 コマンド実行	8
2.5 通信ソフトウェアでコマンド実行	10
2.6 マクロファイルの作成	11
2.7 マクロファイルの実行	12
3. マクロファイルによる通信コマンドの実行例	13
3.1 パラメータファイル、設定ファイル、プログラムファイルをダウンロード	13
3.2 複数プログラムファイルの書き込み	14
3.3 異なる書き込み環境を指定した連続書き込み	14
3.4 複数のターゲットコマンド（消去、書き込み、ベリファイ等）の連続実行	17
3.5 ユニークコードの埋め込み	18
3.6 指定した PG-FP6 での書き込み	19
3.7 1つの PC で複数デバイスへの同時書き込み（ギャング書き込み）	19

1. 概要

フラッシュプログラマ PG-FP6 はホスト PC からシリアル通信による通信コマンドを使用して PG-FP6 を操作することが可能です。また、通信ソフトウェアなどを使用して PG-FP6 を操作することで自動書き込みも可能です。PG-FP6 の詳細についてはユーザーズマニュアルを参照して下さい。

<https://www.renesas.com/pg-fp6>

2. シリアル通信によるコマンド制御までの基本手順

この章では通信ソフトウェアによる PG-FP6 の通信コマンドを実行するまでの基本的な一連の手順を理解していただくために、RL78/G13 をターゲットマイコンにした場合を例に手順を説明します。

- ◆ この章で解説する一連の条件は次の通りです。

ホスト PC インタフェース	: USB
プログラミングエリア	: エリア 0
ターゲットマイコン	: R5F100LE (RL78/G13)
プログラミング GUI	: FP6 Terminal V1.04.01
通信ソフトウェア	: Tera Term Version 4.105※
電源設定	: PG-FP6 から 5V 供給
供給クロック	: 32MHz (高速オンチップ・オシレータ)
書き込み通信	: 1 wire UART (1,000,000bps)
セキュリティ設定	: セキュリティ無効状態
フラッシュ操作	: 消去、書き込み、ベリファイ
動作モード	: チップモード
フラッシュオプション	: 設定しない

※Tera Term の公式サイト : <http://ttssh2.osdn.jp>

- ◆ この章で解説する一連の手順は次の通りです。

なお、1.から 4.は FP6 Terminal の操作について説明しています。理解されている場合、5.から参照して下さい。

1. システムの接続
2. ターゲットシステムの接続
3. 設定ファイルの作成
4. コマンド実行
5. 通信ソフトウェアでコマンド実行
6. マクロファイルの作成
7. マクロファイルの実行

2.1 システムの接続

ホスト PC の USB ポートと PG-FP6 を USB ケーブルで接続します。PG-FP6 と添付の電源アダプタを接続します。PG-FP6 の電源スイッチを ON にします。

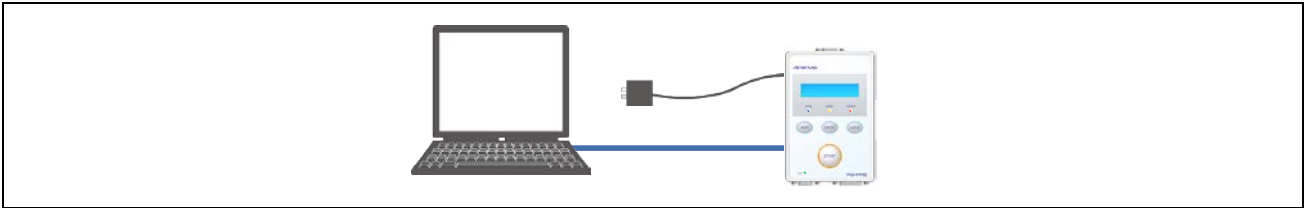


図 2.1 システムの接続

2.2 ターゲットシステムの接続

PG-FP6 のターゲットケーブルをターゲットシステムに接続します。

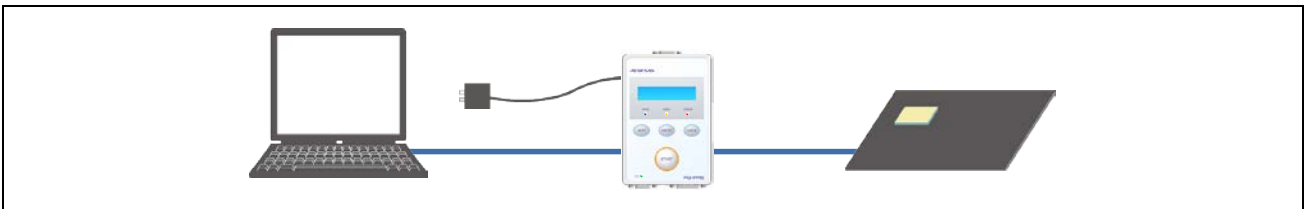


図 2.2 ターゲットシステムの接続

2.3 設定ファイルの作成

FP6 Terminal を起動するとメインウィンドウが開きます。

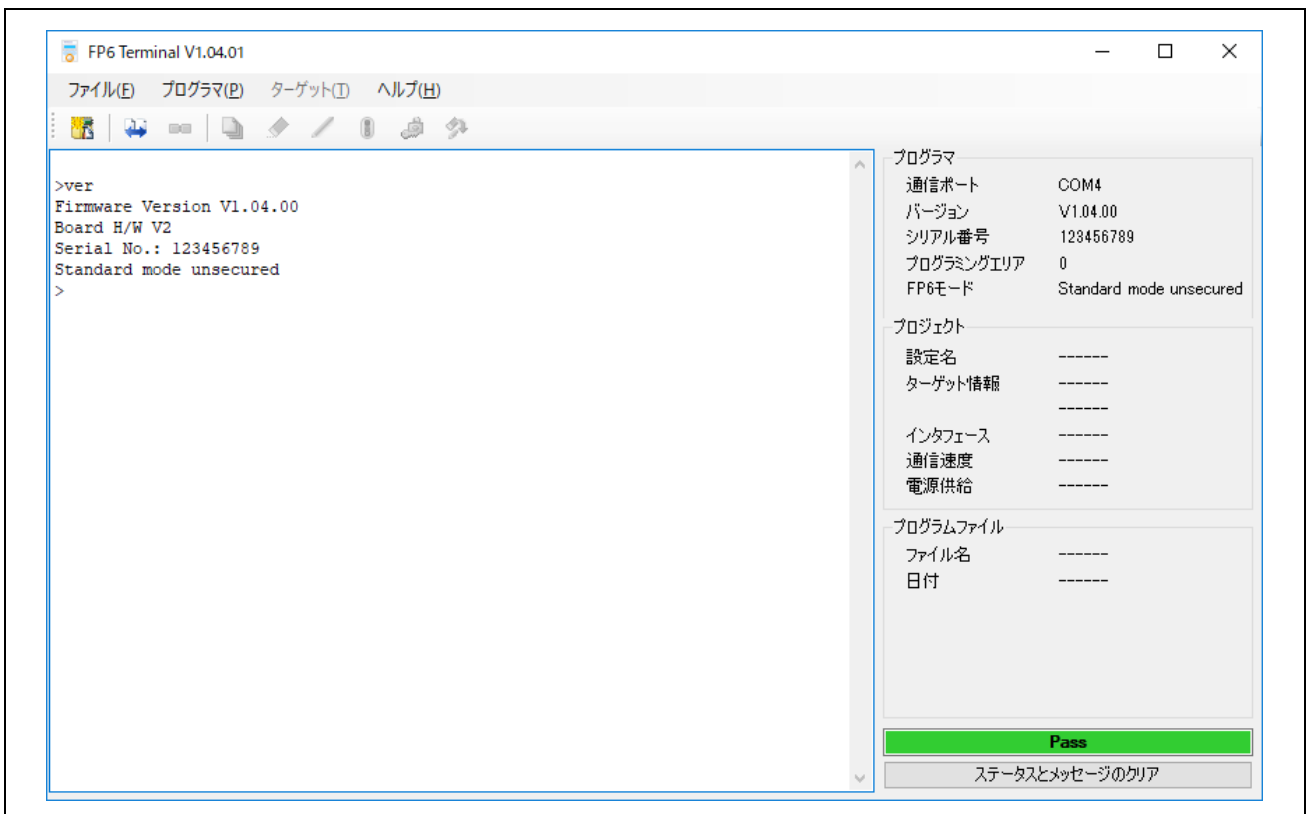


図 2.3 メインウィンドウ

メニューバーの[ファイル]→[新しい設定ファイルを作成]を選択すると、[新しい設定の作成]ダイアログが開きます。

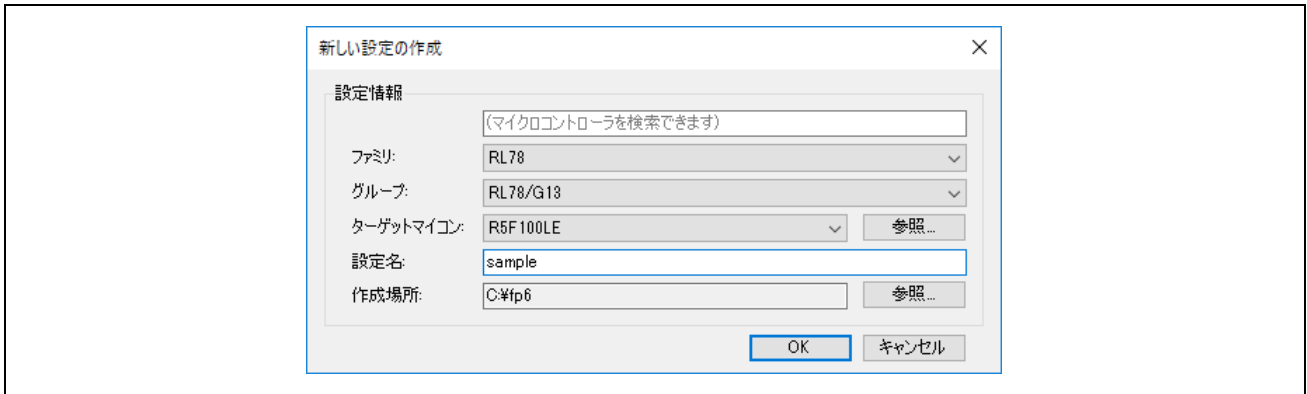


図 2.4 [新しい設定の作成]ダイアログ

[ファミリー]は“RL78”を選択、[グループ]は“RL78/G13”を選択、[ターゲットマイコン]は“R5F100LE”を選択、[設定名]は“sample”を入力、[作成場所]は“C:\fp6”を指定します。
 ボタンをクリックすると、[セットアップ]ダイアログが開きます。

[プログラムファイル]タブの[プログラムファイル]は“C:\fp6\sample.mot”を指定します。

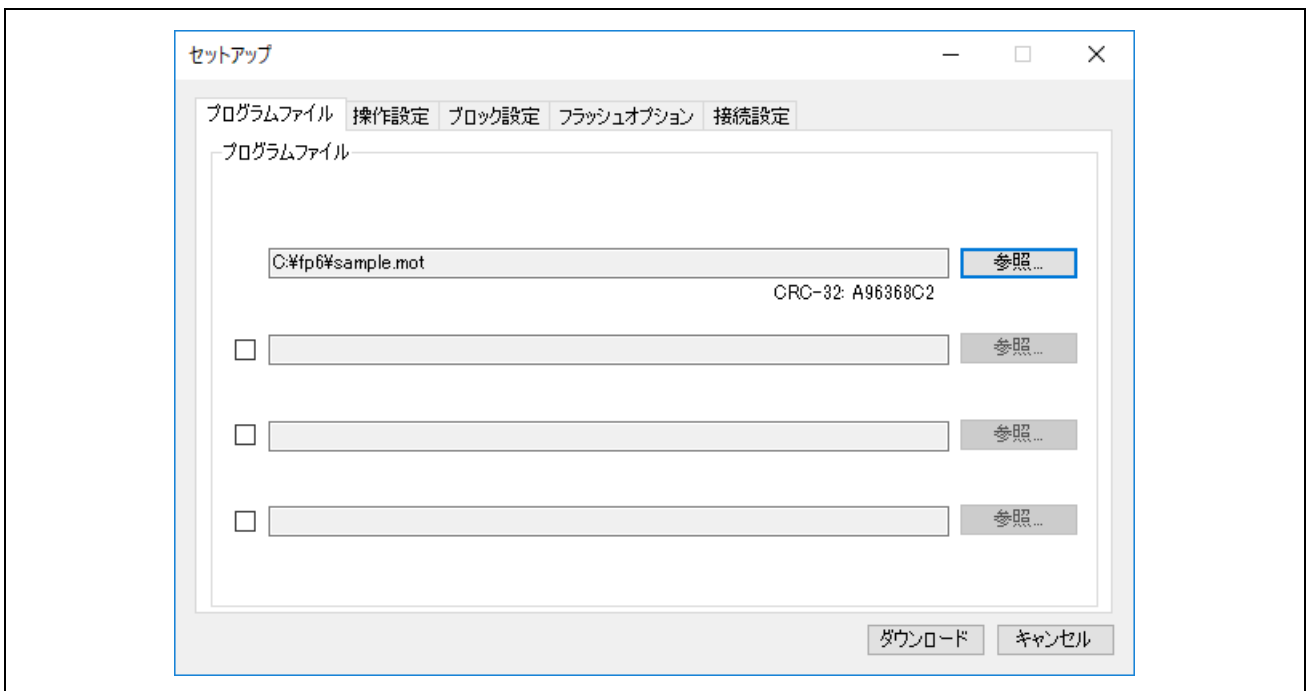


図 2.5 [プログラムファイル]タブ

[操作設定]タブの[コマンド]は“ベリファイ”をチェックします。

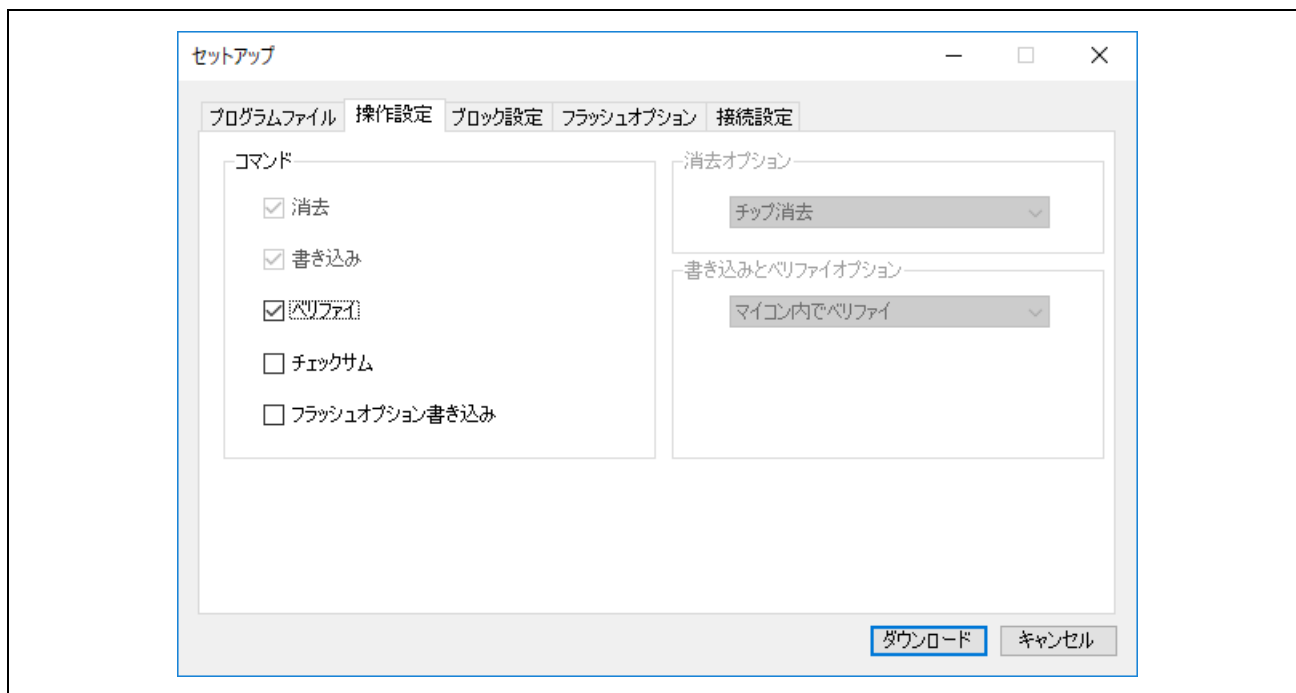


図 2.6 [操作設定]タブ

[ブロック設定]タブは[動作モード]が“チップモード”であることを確認してください。

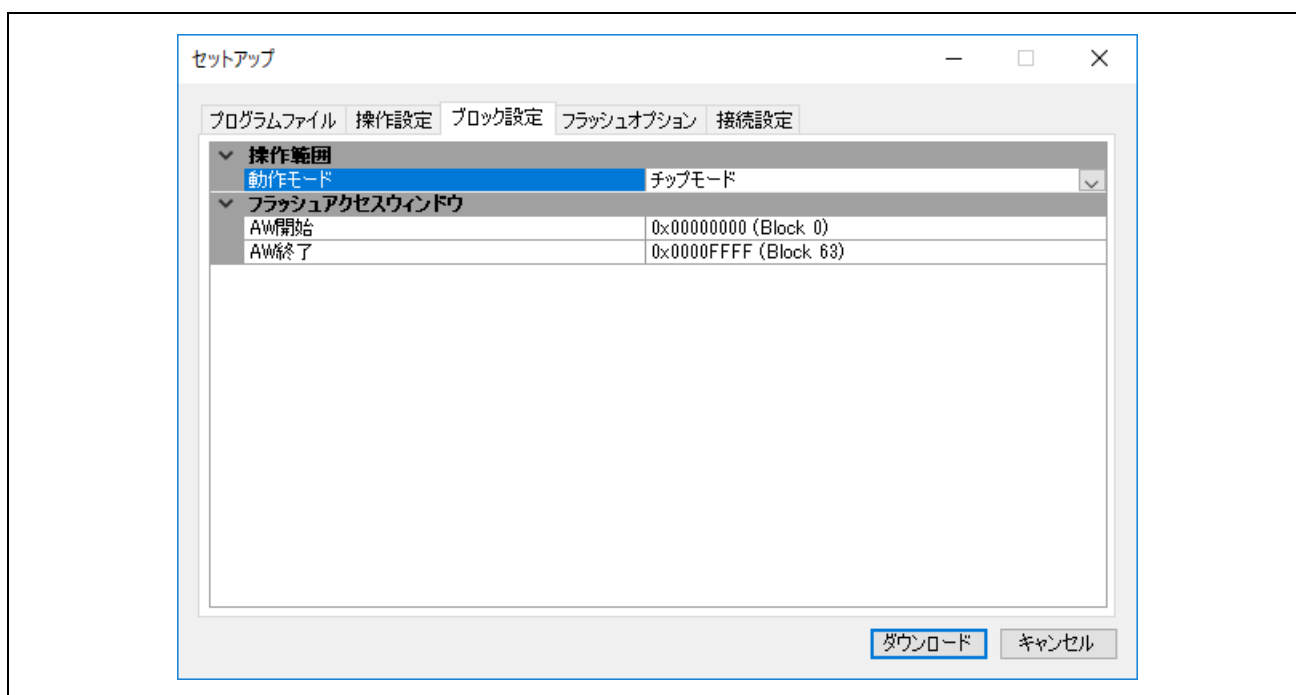


図 2.7 [ブロック設定]タブ

[フラッシュオプション]タブの[設定オプション]が“何もしない”であることを確認してください。

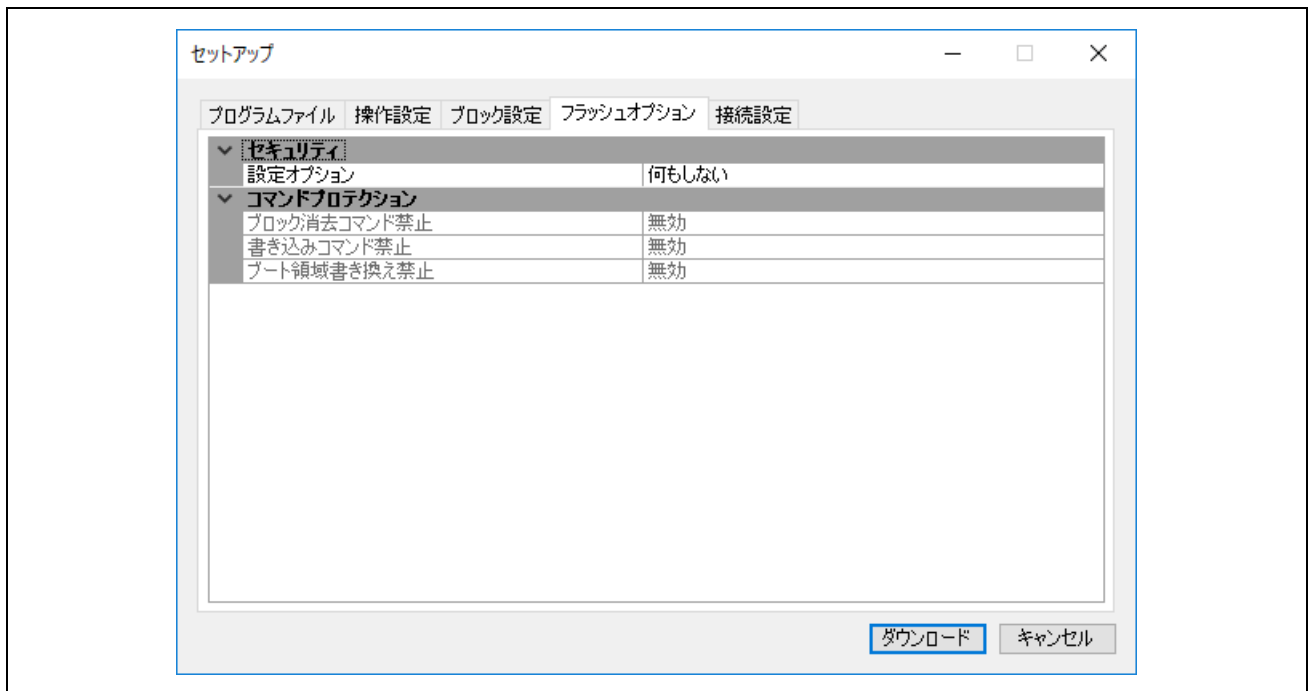


図 2.8 [フラッシュオプション]タブ

[接続設定]タブの[速度]は“1,000,000” bps に設定します。[電源設定]は“FP6 から供給する” “5.0” [V]であることを確認してください。

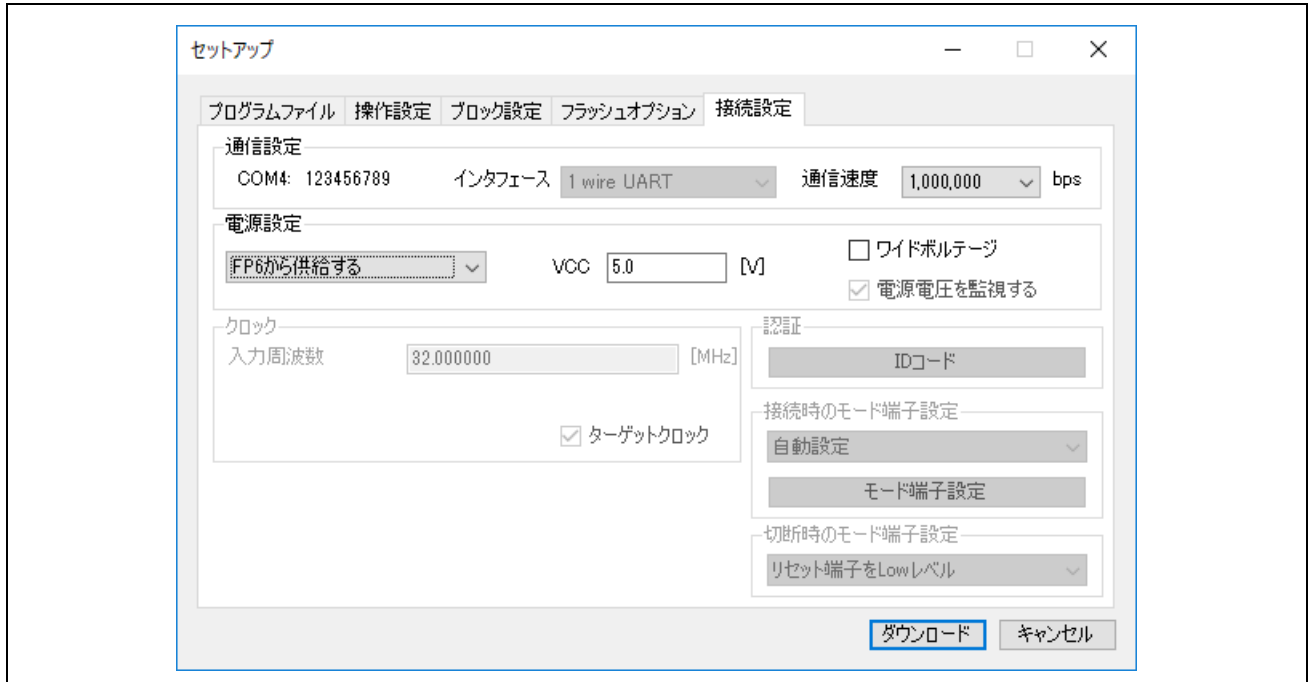


図 2.9 [接続設定]タブ

ダウンロードボタンによりパラメータファイル、設定ファイル、プログラムファイルを PG-FP6 にダウンロードします。

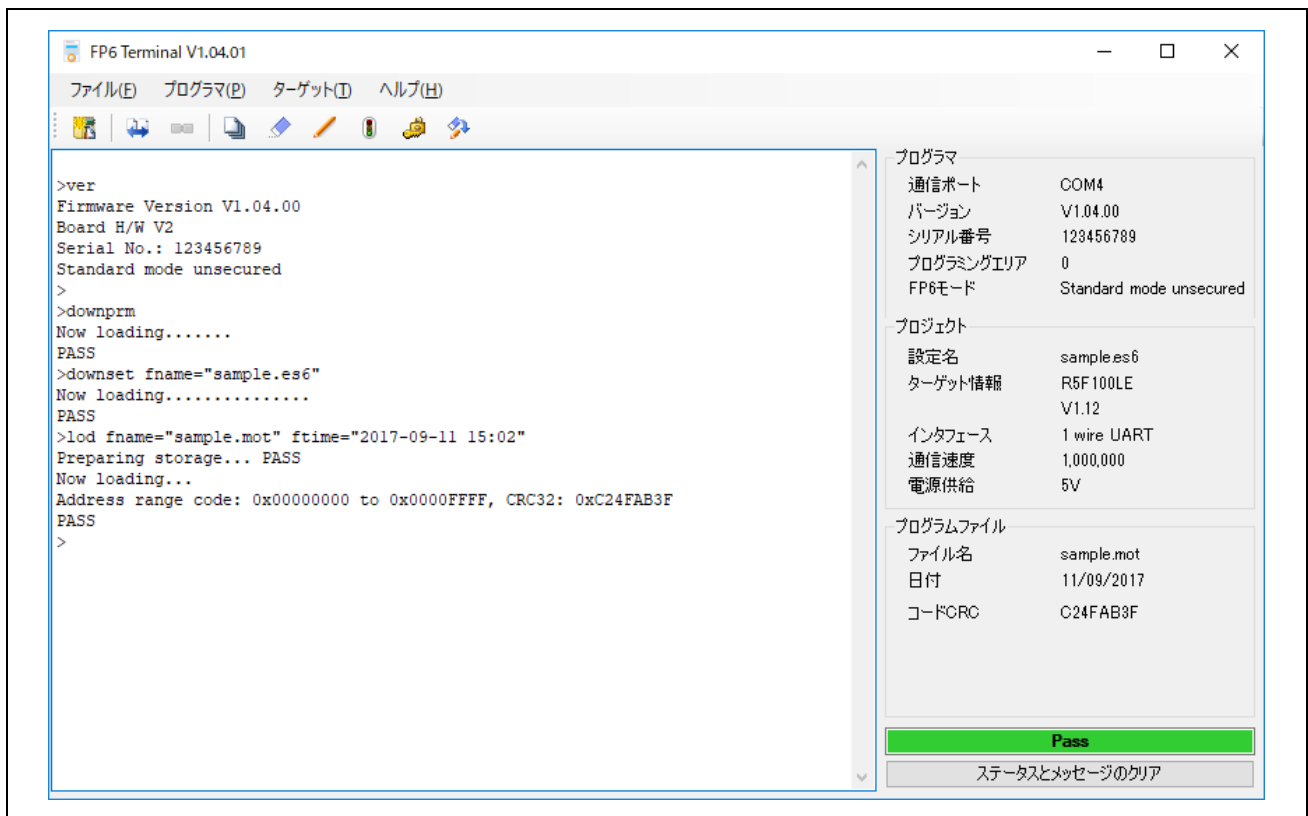


図 2.10 ダウンロード

2.4 コマンド実行

メニューバーの[ターゲット]→[コマンド実行]を実行すると、R5F100LE に対して[消去]→[書き込み]→[ベリファイ]を実行します。

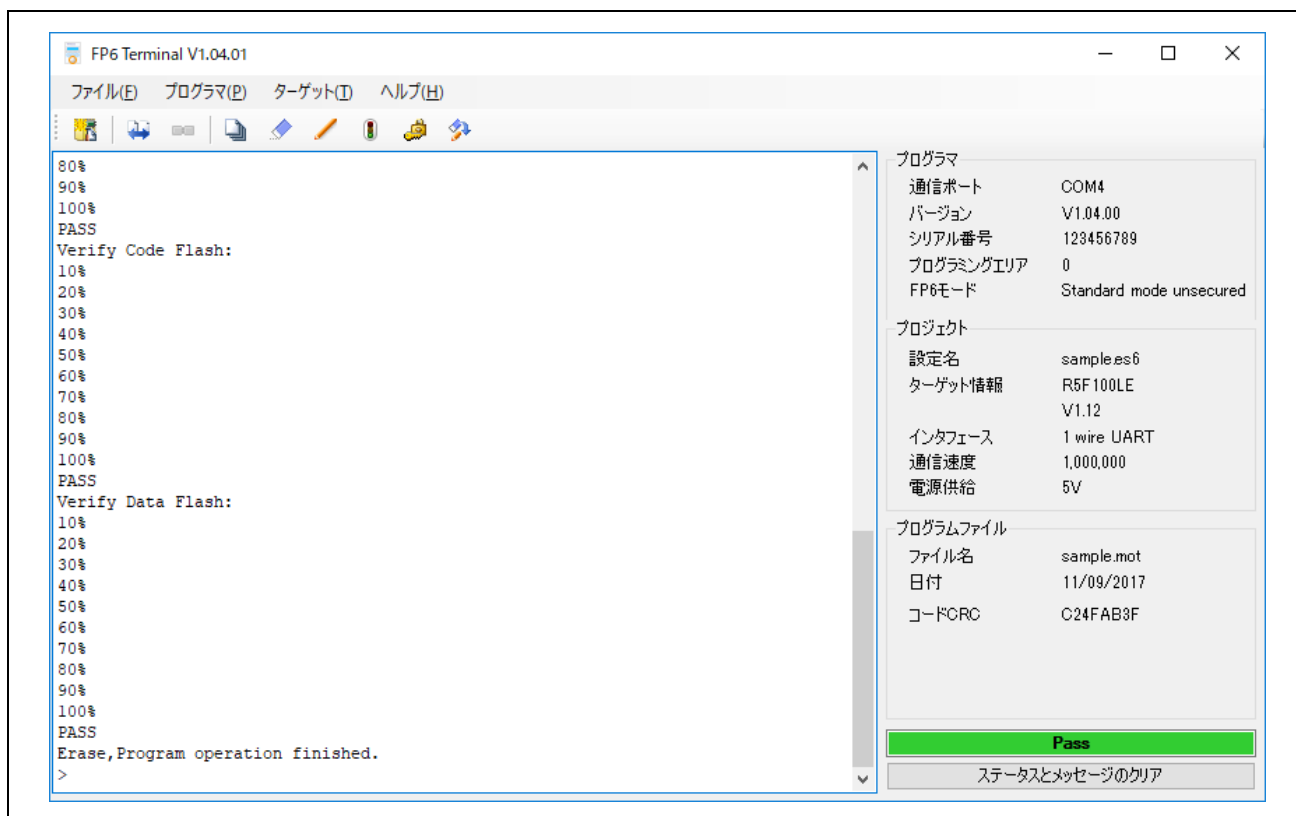


図 2.11 コマンド実行

メニューバーの[ファイル]→[終了]を選択して FP6 Terminal を終了します。

2.5 通信ソフトウェアでコマンド実行

Tera Term を起動すると[新しい接続]ダイアログが開きます。“シリアル” “COMx: PG-FP6(CDC) (COMx)” を選択します。OK ボタンを押すとメインウィンドウが開きます。

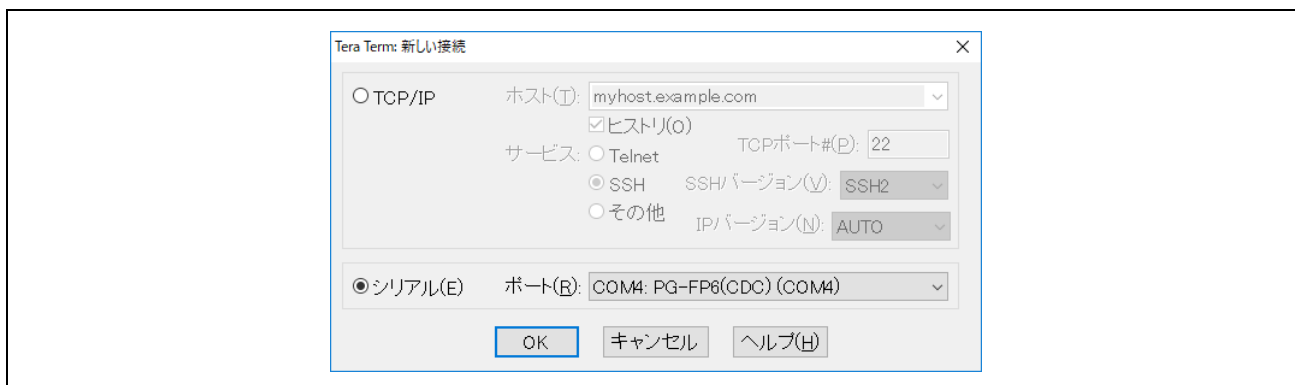


図 2.12 メインウィンドウ

Enter キーを押すと “>” が表示します。

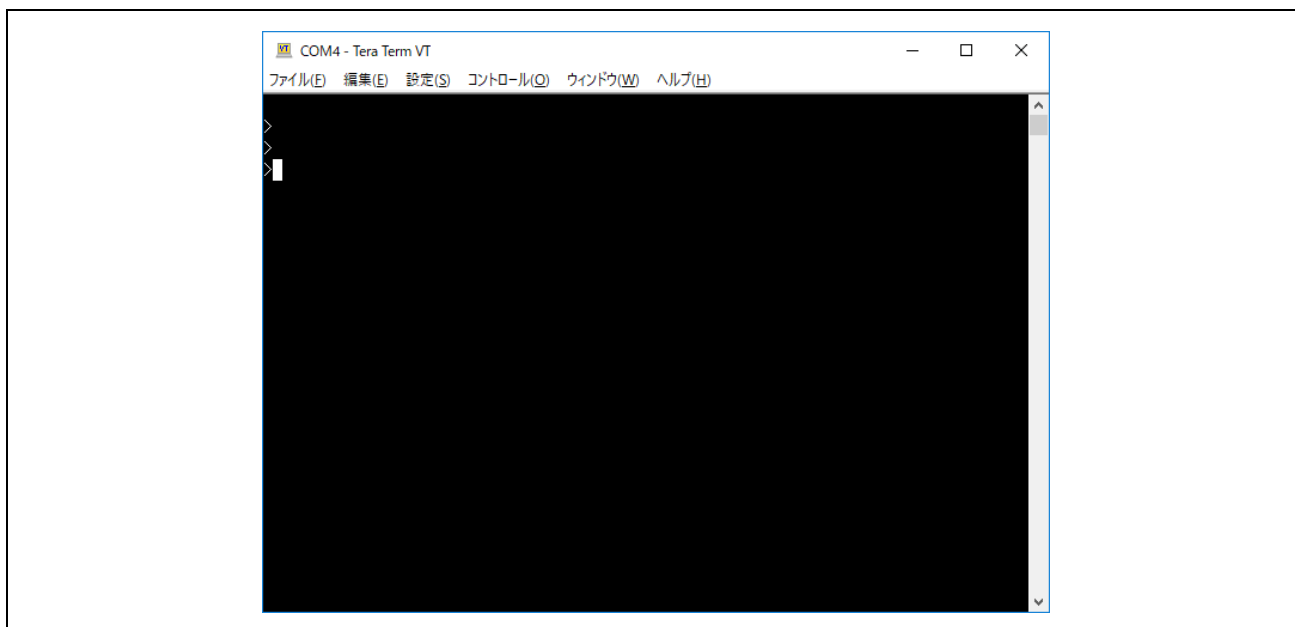


図 2.13 メインウィンドウ

[コマンド実行] “ep” を送信すると、R5F100LE に対して[消去]→[書き込み]→[ベリファイ]を実行します。

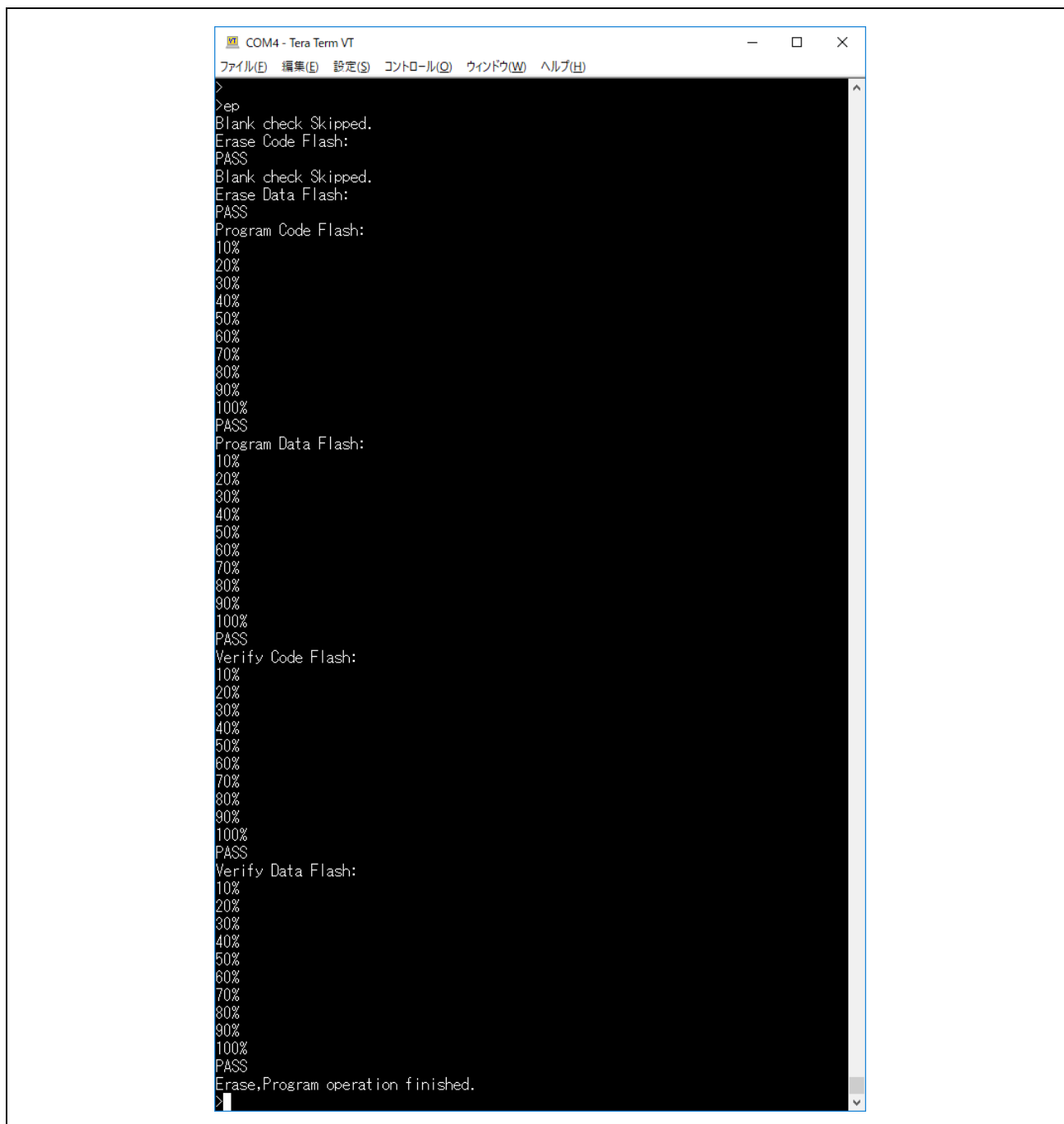


図 2.14 コマンド実行

2.6 マクロファイルの作成

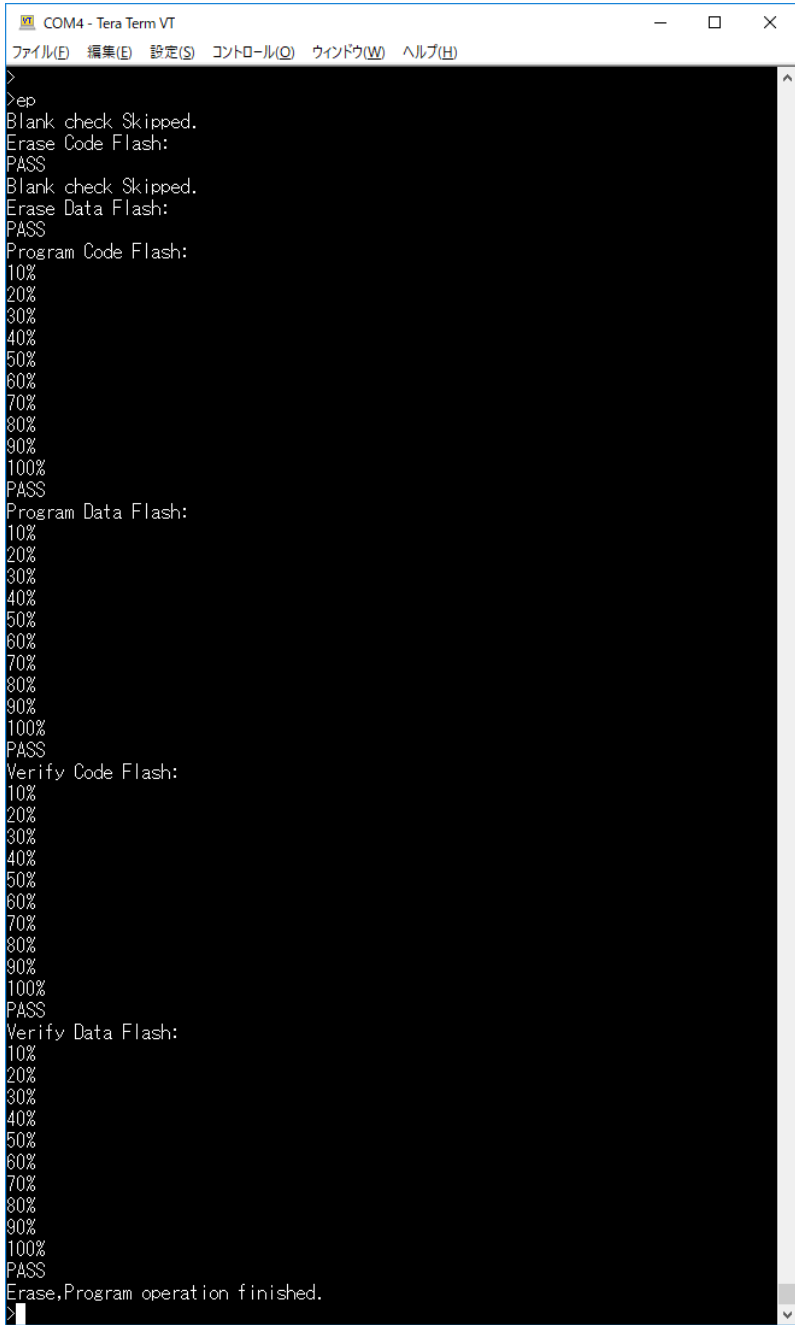
テキストエディタで[コマンド実行] “ep” を送信する内容を入力し “C:%fp6%sample.ttl” に保存します。

```
sendln 'ep'
end
```

図 2.15 マクロファイルの作成

2.7 マクロファイルの実行

メニューバーの[コントロール]→[マクロ]でマクロファイル “sample.ttl” を開くと、R5F100LE に対して [消去]→[書き込み]→[ベリファイ]を実行します。



```
COM4 - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
>
>ep
Blank check Skipped.
Erase Code Flash:
PASS
Blank check Skipped.
Erase Data Flash:
PASS
Program Code Flash:
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
PASS
Program Data Flash:
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
PASS
Verify Code Flash:
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
PASS
Verify Data Flash:
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
PASS
Erase, Program operation finished.
>
```

図 2.16 マクロファイルの実行

3. マクロファイルによる通信コマンドの実行例

この章では Tera Term のマクロファイルを用いた PG-FP6 の通信コマンドの自動制御方法について紹介します。通信コマンドの詳細についてはユーザーズマニュアルを参照して下さい。

<https://www.renesas.com/pg-fp6>

3.1 パラメータファイル、設定ファイル、プログラムファイルをダウンロード

downprm コマンド、downset コマンド、lod コマンドを使用することで、パラメータファイル、設定ファイル、プログラムファイルをダウンロードすることが可能です。

パラメータファイル “R5F100LE.pr5”、設定ファイル “sample.es6”、プログラムファイル “sample.mot” を指定してダウンロードする例です。

```
sendln 'downprm'
wait 'Now loading...'
```

```
sendfile 'C:\%fp6%\sample\R5F100LE.pr5' 1
wait '>'
```

```
sendln 'downset'
wait 'Now loading...'
```

```
sendfile 'C:\%fp6%\sample\sample.es6' 1
wait '>'
```

```
sendln 'lod fname="sample.mot" ftime="2017-09-11 15:02"'
wait 'Now loading...'
```

```
sendfile 'C:\%fp6%\sample.mot' 1
end
```

Tera Term 上に下記内容が出力されます。

```
>downprm
Now loading.....
PASS
>downprm
Now loading.....
PASS
>downset
Now loading.....
PASS
>lod fname="sample.mot" ftime="2017-09-11 15:02"
Preparing storage... PASS
Now loading...
Address range code: 0x00000000 to 0x0000FFFF, CRC32: 0xC24FAB3F
PASS
>
```

3.2 複数プログラムファイルの書き込み

lod コマンドの add オプションを使用することで複数のプログラムファイルをダウンロードすることが可能です。

2つのプログラムファイル “sample1.mot” と “sample2.mot” をダウンロードして書き込む例です。

```
sendln 'lod fname="sample1.mot" ftime="2020-06-18 16:24"'
wait 'Now loading...'
sendfile 'C:\%fp6%\sample1.mot' 1
wait '>'
sendln 'lod add fname="sample2.mot" ftime="2020-06-18 16:26"'
wait 'Now loading...'
sendfile 'C:\%fp6%\sample2.mot' 1
wait '>'
sendln 'ep'
end
```

Tera Term 上に下記内容が出力されます。

```
>lod fname="sample1.mot" ftime="2020-06-18 16:24"
Preparing storage... PASS
Now loading...
Address range code: 0x00000000 to 0x00007FFE, CRC32: 0xC7D7C74C
PASS
>lod add fname="sample2.mot" ftime="2020-06-18 16:26"
Now loading...
Address range code: 0x00008000 to 0x0000FFFF, CRC32: 0x25FDCC28
PASS
>ep
:(省略)
Erase,Program operation finished.
>
```

3.3 異なる書き込み環境を指定した連続書き込み

PG-FP6 のプログラミングエリアに異なる書き込み環境をダウンロードして、エリア番号を指定して連続書き込みが可能です。

1種類のマイコン（1種類のパラメータファイル）と、2種類の設定ファイルとプログラムファイルを、プログラミングエリア0とプログラミングエリア1にダウンロードし、エリア0、エリア1を指定して連続書き込みする例です。

```
sendln 'progarea 0'  
wait '>'  
sendln 'downprm'  
wait 'Now loading...'  
sendfile 'C:\fp6\sample\R5F100LE.pr5' 1  
wait '>'  
sendln 'downset'  
wait 'Now loading...'  
sendfile 'C:\fp6\sample\sample1.es6' 1  
wait '>'  
sendln 'lod fname="sample1.mot" ftime="2020-06-18 16:24"  
wait 'Now loading...'  
sendfile 'C:\fp6\sample1.mot' 1  
wait '>'  
sendln 'progarea 1'  
wait '>'  
sendln 'downprm'  
wait 'Now loading...'  
sendfile 'C:\fp6\sample\R5F100LE.pr5' 1  
wait '>'  
sendln 'downset'  
wait 'Now loading...'  
sendfile 'C:\fp6\sample\sample2.es6' 1  
wait '>'  
sendln 'lod fname="sample2.mot" ftime="2020-06-18 16:26"  
wait 'Now loading...'  
sendfile 'C:\fp6\sample2.mot' 1  
wait '>'  
sendln 'progarea 0'  
wait '>'  
sendln 'ep'  
wait '>'  
sendln 'progarea 1'  
wait '>'  
sendln 'ep'  
end
```

Tera Term 上に下記内容が出力されます。

```
>progarea 0
Active Program Area: 0
>downprm
Now loading.....
PASS
>downset
Now loading.....
PASS
>lod fname="sample1.mot" ftime="2020-06-18 16:24"
Preparing storage... PASS
Now loading...
Address range code: 0x00000000 to 0x00007FFE, CRC32: 0xC7D7C74C
PASS
>progarea 1
Active Program Area: 1
>downprm
Now loading.....
PASS
>downset
Now loading.....
PASS
>lod fname="sample2.mot" ftime="2020-06-18 16:26"
Preparing storage... PASS
Now loading...
Address range code: 0x00008000 to 0x0000FFFF, CRC32: 0x25FDCC28
PASS
>progarea 0
Active Program Area: 0
>ep
Blank check Skipped.
Erase Code Flash Block 0000 to 0031:
PASS
Program Code Flash Block 0000 to 0031:
:(省略)
PASS
Verify Code Flash Block 0000 to 0031:
```



```

:(省略)
PASS
Erase,Program operation finished.
>progarea 1
Active Program Area: 1
>ep
Blank check Skipped.
Erase Code Flash Block 0032 to 0063:
PASS
Program Code Flash Block 0032 to 0063:
:(省略)
PASS
Verify Code Flash Block 0032 to 0063:
:(省略)
PASS
Erase,Program operation finished.
>

```

3.4 複数のターゲットコマンド（消去、書き込み、ベリファイ等）の連続実行

複数のターゲットコマンド（消去、書き込み、ベリファイ等）を連続実行することが可能です。

3つのコマンド“ers:消去と prg : 書き込みと vrf : ベリファイ”を連続実行する例です。

```

sendln 'ers'
wait '>'
sendln 'prg'
wait '>'
sendln 'vrf'
end

```

Tera Term 上に下記内容が出力されます。

```

>ers
:(省略)
PASS
Erase operation finished.
>prg
:(省略)
PASS
Program operation finished.

```

```

>vrf
:(省略)
PASS
Verify operation finished.
>

```

3.5 ユニークコードの埋め込み

serno コマンドを使用することで、一時的にプログラムファイル内の一部データをユニークコードに差し換えて、書き込むことが可能です。

予め、FP6 Terminal のメニューバーの[プログラマ]→[FP6 管理設定]で[FP6 管理設定]ダイアログを開き、[ユニークコード機能を制限する]のチェックを外してください。

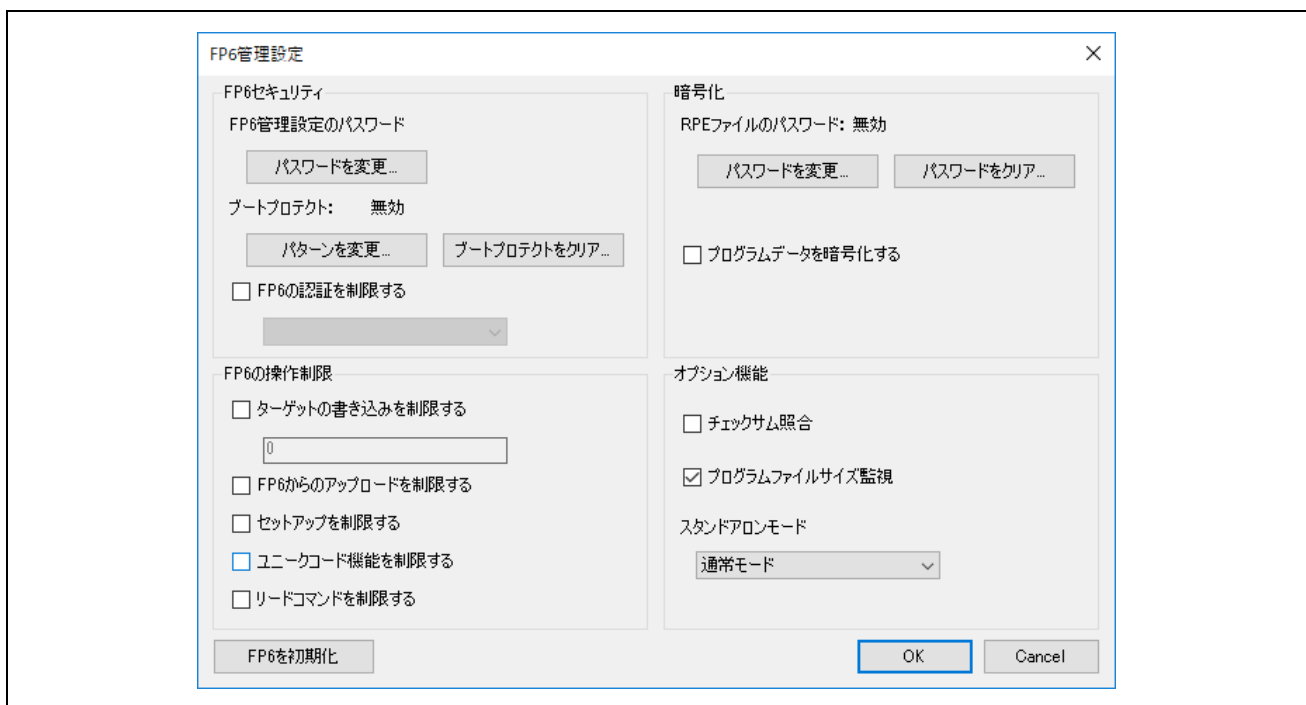


図 2.1 [FP6 管理設定]ダイアログ

0x8000 番地に 0x00010203 データを埋め込んで書き込む例です。

```

sendln 'serno 8000 00010203'
wait ''
sendln 'ep'
end

```

Tera Term 上に下記内容が出力されます。

```

>serno 8000 00010203
OK
>ep
:(省略)

```

```
Erase,Program operation finished.
>
```

3.6 指定した PG-FP6 での書き込み

Tera Term マクロの connect コマンドを使用することで、PG-FP6 が接続された COM ポート番号を指定することが可能です。

COM4 に接続された PG-FP6 を指定して書き込む例です。

```
connect '/C=4'
sendln 'ep'
end
```

Tera Term 上に下記内容が出力されます。

```
>ep
:(省略)
Erase,Program operation finished.
>
```

3.7 1つの PC で複数デバイスへの同時書き込み（ギャング書き込み）

Tera Term は 1 つ PC で複数起動することが可能です。また、PG-FP6 が接続された COM ポート番号を指定することが可能です。また、Windows の start コマンドを使用することで、バッチファイル内から複数のバッチファイルを同時に呼び出して実行することが可能です。

補足：FP6 Terminal に付属のプログラミング GUI「FP6 Gang Programmer」を使用することで複数の PG-FP6 を使用してプログラムを同時に書き込みすることができる可能です。

1 つの書き込み環境（パラメータファイル、設定ファイル、プログラムファイル）で Tera Term を 2 つ同時に起動して、2 つの PG-FP6 に接続されたそれぞれのデバイスに対し、同時に書き込む例です。

マクロファイルの実行は Tera Term の ttpmacro.exe を使用します。

事前に 1 つの書き込み環境（パラメータファイル、設定ファイル、プログラムファイル）を 2 つの PG-FP6 にダウンロードしておきます。

1 つ目の PG-FP6（COM 番号：x）用バッチファイル “sample1.bat” の例です。

```
ttpmacro.exe "C:%fp6%sample1.ttl"
exit
```

1 つ目の PG-FP6（COM 番号：x）用マクロファイル “sample1.ttl” の例です。

```
connect '/C=x'
sendln 'ep'
wait 'Erase,Program operation finished!' 'ERROR'
    if result=0 goto timeouterror
    if result=1 goto pass
```

```
        if result=2 goto comerror
wait '>'
        if result=0 goto timeouterror

:pass
messagebox 'Erase,Program operation finished.' 'PASS'
closett
end

:comerror
messagebox 'Command operation error' 'ERROR'
closett
end

:timeouterror
messagebox 'Timeout error' 'ERROR'
closett
end
```

2 つ目の PG-FP6（シリアル番号 : y）用バッチファイル “sample2.bat” の例です。

```
ttpmacro.exe "C:%fp6%sample2.ttl"
exit
```

2 つ目の PG-FP6（COM 番号 : y）用マクロファイル “sample2.ttl” の例です。

```
connect '/C=y'
sendln 'ep'
wait 'Erase,Program operation finished.' 'ERROR'
        if result=0 goto timeouterror
        if result=1 goto pass
        if result=2 goto comerror
wait '>'
        if result=0 goto timeouterror

:pass
messagebox 'Erase,Program operation finished.' 'PASS'
closett
end
```

```
:comerror
messagebox 'Command operation error' 'ERROR'
closett
end

:timeouterror
messagebox 'Timeout error' 'ERROR'
closett
end
```

“sample1.bat” と “sample2.bat” を呼び出して実行するバッチファイル “sample.bat” の例です。

```
:LOOP
start sample1.bat
start sample2.bat
PAUSE
goto LOOP
```

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。お客様の機器・システムの設計において、回路、ソフトウェアおよびこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含まれます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品、本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
5. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。

標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等

高品質水準： 輸送機器（自動車、電車、船舶等）、交通管制（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

- 当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。
6. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
 7. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
 8. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関し、当社は、一切その責任を負いません。
 9. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
 10. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものとなります。
 11. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
 12. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。
- 注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。
- 注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.4.0-1 2017.11)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。