

S1C17 Family Application Note

S1C17 シリーズ
ブートローダ
サンプルソフトウェア

評価ボード・キット、開発ツールご使用上の注意事項

1. 本評価ボード・キット、開発ツールは、お客様での技術的評価、動作の確認および開発のみに用いられることを想定し設計されています。それらの技術評価・開発等の目的以外には使用しないで下さい。本品は、完成品に対する設計品質に適合していません。
2. 本評価ボード・キット、開発ツールは、電子エンジニア向けであり、消費者向け製品ではありません。お客様において、適切な使用と安全に配慮願います。弊社は、本品を用いることで発生する損害や火災に対し、いかなる責も負いかねます。通常の使用においても、異常がある場合は使用を中止して下さい。
3. 本評価ボード・キット、開発ツールに用いられる部品は、予告無く変更されることがあります。

本資料のご使用につきましては、次の点にご留意願います。
本資料の内容については、予告無く変更することがあります。

1. 本資料の一部、または全部を弊社に無断で転載、または、複製など他の目的に使用することは堅くお断りいたします。
2. 本資料に掲載される応用回路、プログラム、使用方法等はあくまでも参考情報であり、これらに起因する第三者の知的財産権およびその他の権利侵害あるいは損害の発生に対し、弊社はいかなる保証を行うものではありません。また、本資料によって第三者または弊社の知的財産権およびその他の権利の実施権の許諾を行うものではありません。
3. 特性値の数値の大小は、数直線上の大小関係で表しています。
4. 製品および弊社が提供する技術を輸出等するにあたっては「外国為替および外国貿易法」を遵守し、当該法令の定める手続きが必要です。大量破壊兵器の開発等およびその他の軍事用途に使用する目的をもって製品および弊社が提供する技術を費消、再販または輸出等しないでください。
5. 本資料に掲載されている製品は、生命維持装置その他、きわめて高い信頼性が要求される用途を前提としていません。よって、弊社は本（当該）製品をこれらの用途に用いた場合のいかなる責任についても負いかねます。
6. 本資料に掲載されている会社名、商品名は、各社の商標または登録商標です。

目次

1. 概要.....	1
1.1. ロード可能なデータ形式.....	1
1.2. C17の端子割り当ておよびシリアル通信フォーマット.....	1
2. 機能.....	2
2.1 ブートローダ.....	2
2.2 ユーザプログラム.....	4
2.3 PC.....	4
3. メモリの使用.....	5
3.1 更新インジケータ ([indicator]領域).....	5
3.2 自己書き換えライブラリの予約RAM ([fls_ram]領域).....	5
改訂履歴表.....	7

1. 概要

このパッケージは、C17 自己書き換えライブラリを呼び出すブートローダの例と、ブートローダによりロード可能なプログラムのサンプルを含みます。

このパッケージのフォルダ構成は以下の通りです。

- s1c17xxx_bootloader ブートローダ
- s1c17xxx_loadsample ロード可能なサンプルプログラム

注 xxx：機種名

1.1. ロード可能なデータ形式

ロード可能なデータ形式は、モトローラ S レコードです。ブートローダのテストには、GNU17 の生成した PSA ファイル (S レコードファイルの一種) を使用しました。ダウンロードされたデータの終端は、S7/S8/S9 レコードのいずれかである必要があり、これにより、ブートローダはダウンロードが完了したことを認識します。PSA ファイルの末尾は S8 レコードです。

1.2. C17 の端子割り当ておよびシリアル通信フォーマット

ブートローダは UART チャンネル 0 を使用して PC と通信します。UART チャンネル 0 は以下の端子を使用します。

端子名	機能
P00	UART/チャンネル 0/USOUT0
P01	UART/チャンネル 0/USIN0

Flash 書き換え電圧を内部生成できない MCU のブートローダは、VPP (Flash 書き換え用外部電源) の ON/OFF を制御するために以下の端子から High/Low レベルの信号を出力します。

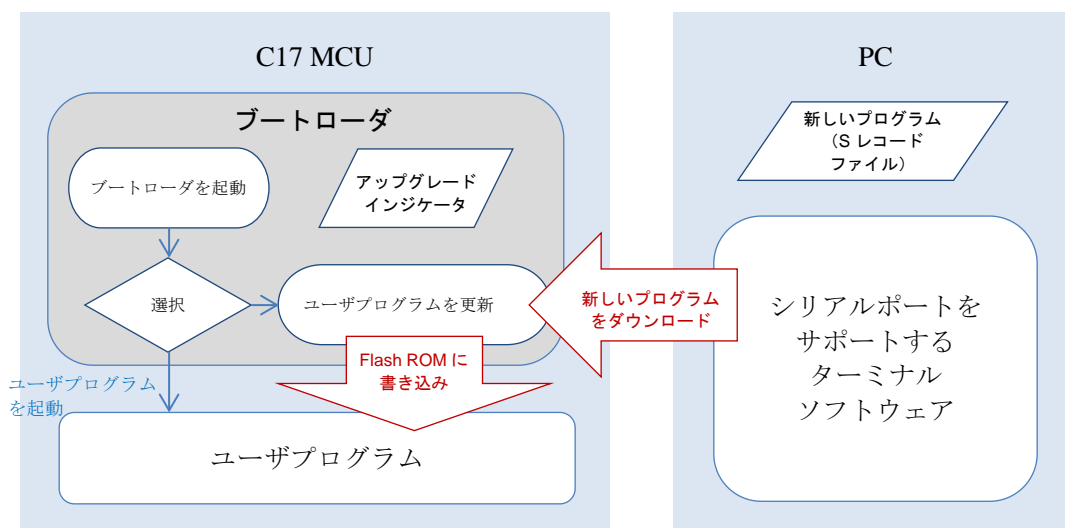
端子名	機能
P10	VPP コントロール信号

UART のチャンネル 0 は以下の形式で初期化されています。

通信速度	115200bps(C17W18, C17W23) 230400bps(上記以外の MCU)
データ長	8 ビット
パリティ	なし
ストップビット	1 ビット
フロー制御	ソフトウェア (XON/XOFF)

2. 機能

2. 機能



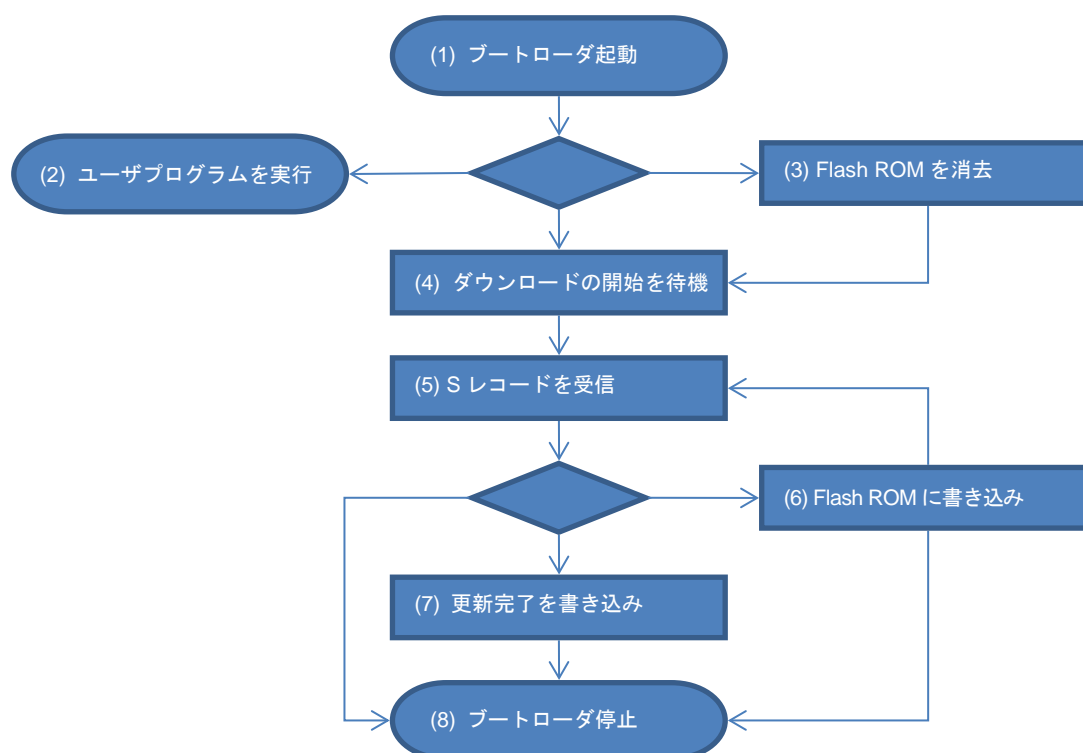
2.1 ブートローダ

ブートローダは、Flash ROM の 0x8000 から 0xBFFF までの領域にあります (16KB)。

MCU がリセットされると、ブートローダが動作を開始し、ユーザプログラムの起動、更新の2つの機能のうちのいずれかを選択します。

プログラムの更新が完了しているならば、ブートローダは 0xC000 以降に存在するユーザプログラムを起動します。

更新中であれば、ブートローダは UART を使用して新しいプログラムを受信し、0xC000 以降の領域に書き込みます。新しいプログラムの受信が終了するとブートローダが停止するので、C17 MCU をリセットしてください。プログラムの更新が完了しているので、リセット後は新しいプログラムが起動します。



ブートローダをデバイスに実装するときは、以下の点を検討してください。

- A) アップグレード開始の表示
ブートローダが(1)から(3) (4)に移行するときには、「アップグレード開始」をユーザに表示する必要があります。この表示により、適切なタイミングで UART ケーブルを接続するように、ユーザを案内してください。さもないと、通信を開始したときに UART エラーが発生します。
- B) アップグレード終了の表示
ブートローダが(7)から(8)に移行するときには、「アップグレード終了」をユーザに表示する必要があります。
- C) 問題を検出したときにブートローダがどのように動作すべきか。
- 通信タイムアウト
このブートローダの(4)(5)は UART の受信またはタイムアウトをモニタします。しかし、タイムアウトが発生しても、(4)(5)を再実行します。ブートローダは、XOFF を送信した後も、タイムアウト期間 (3 ミリ秒) 待機します。このタイムアウトは短縮可能です。
 - 通信エラー
UART 通信でエラーが発生した場合、レコードの再送を要求する必要があります。このブートローダの(4)(5)は UART エラーをモニタしませんが、これは、PC 上で動作しているターミナルソフトが再送要求を受け付けられないためです。
 - データの誤り
このブートローダは、S レコードのチェックサムを確認し(5)、エラー時には書き込みを行わずに停止します(8)。レコードの再送を要求したほうがよいです。
 - Flash ROM の書き込みエラー
書き込み済みのアドレスに、データを重ね書きする可能性があります。同じ値は再書き込みできます。1 から 0 になる値は再書き込みできますが、0 から 1 になる書き込みはエラーとなります。
このエラーは、ダウンロードされたデータによって発生します。このエラーが発生した場合、Flash ROM を消去し、もう一度最初からダウンロードする必要があります。
- D) ブートローダはリセットされても正常に機能するか。
更新中にリセットされると、このブートローダは(1)から(4)へ遷移し、Flash ROM を消去しなおしません。消去しなおせば正常に動作するはずですが、最初から書き直す必要があります。
ブートローダが再度消去しない場合、以下の確認が必要です。これらの条件は S レコードを受信しただけでは判断できません。これらの条件に当てはまらない場合は、再度消去してください。
- 書き込み済みのデータが正しいか。
 - 新たに受信したデータと、かつて受信したデータは、同じプログラムか。
- E) いつブートローダが Flash ROM を消去すべきか。
アップグレードインジケータが"do upgrade"のとき、このブートローダ(3)はユーザプログラムとアップグレードインジケータ ([indicator]領域) を消去します。消去により、アップグレードインジケータの値は、"now upgrading"を意味する 0xFF になります。この後、アップグレードが完了するまで、ブートローダが消去することはありません。
結果として、ブートローダがリセットのたびに Flash ROM を消去することはありません。ただし、間違ったダウンロードなどによりユーザプログラム自体が機能しない場合、アップグレードできない状態になります。
ブートローダが Flash ROM を消去する条件を追加したほうがよいです。
- F) ブートローダがロード後(7) にどのように動作すべきか。
このブートローダは(8)でウォッチドッグタイマを起動して、MCU をリセットします。しかし、このとき新しいプログラムを実行することも可能です。
- G) PC 上のローダアプリケーション
PC 上に専用のローダアプリケーションを用意すれば、新しいユーザプログラムをスムーズにダウンロードできます。このようなアプリケーションには以下の機能が必要です。
- プログラムファイルが正しいか検査する。
 - アップグレードプロセスを一時停止する。
 - アップグレードプロセスを再開する。
 - レコードを再送する。
 - アップグレードの完了を確認する。

2. 機能

2.2 ユーザプログラム

0xC0000 以降にユーザプログラムを配置してください。

自身をアップグレードしようとするとき、ユーザプログラムはブートローダに格納されている `start_upgrade` 関数を呼び出します。 `start_upgrade` 関数は、ブートローダのインジケータを "do_upgrade" に変更し、ユーザプログラム（呼び出し元関数）に戻ります。ユーザプログラムは実行を続けることができます。ユーザプログラムの更新を実際に開始するためには、C17 MCU をリセットしてください。

`start_upgrade` 関数の仕様 :

関数の呼び出し方法	この関数のエントリは、ブートローダ用のベクタテーブルに格納されます (0x8000 以降)。ベクタ番号は 28 です (予約ベクタ)。この関数は割り込みハンドラではないため、通常の関数として呼び出します。 <pre>typedef void (*bootloader_entry_t)(void); extern bootloader_entry_t __START_bootloader[]; // = 0x8000 __START_bootloader[28](); // call "start_upgrade"</pre>
必要 RAM	0x0000~0x07FF (ブートローダの [fls_ram] 領域)
必要スタック領域	112 バイト
使用ペリフェラル (自己書き換えライブラリと同じ)	タイムアウトの確認に T16 Ch.1 を使用します。 SYSCLK を OSC3 4MHz にするように、CLG/OSC 設定を変更します。

2.3 PC

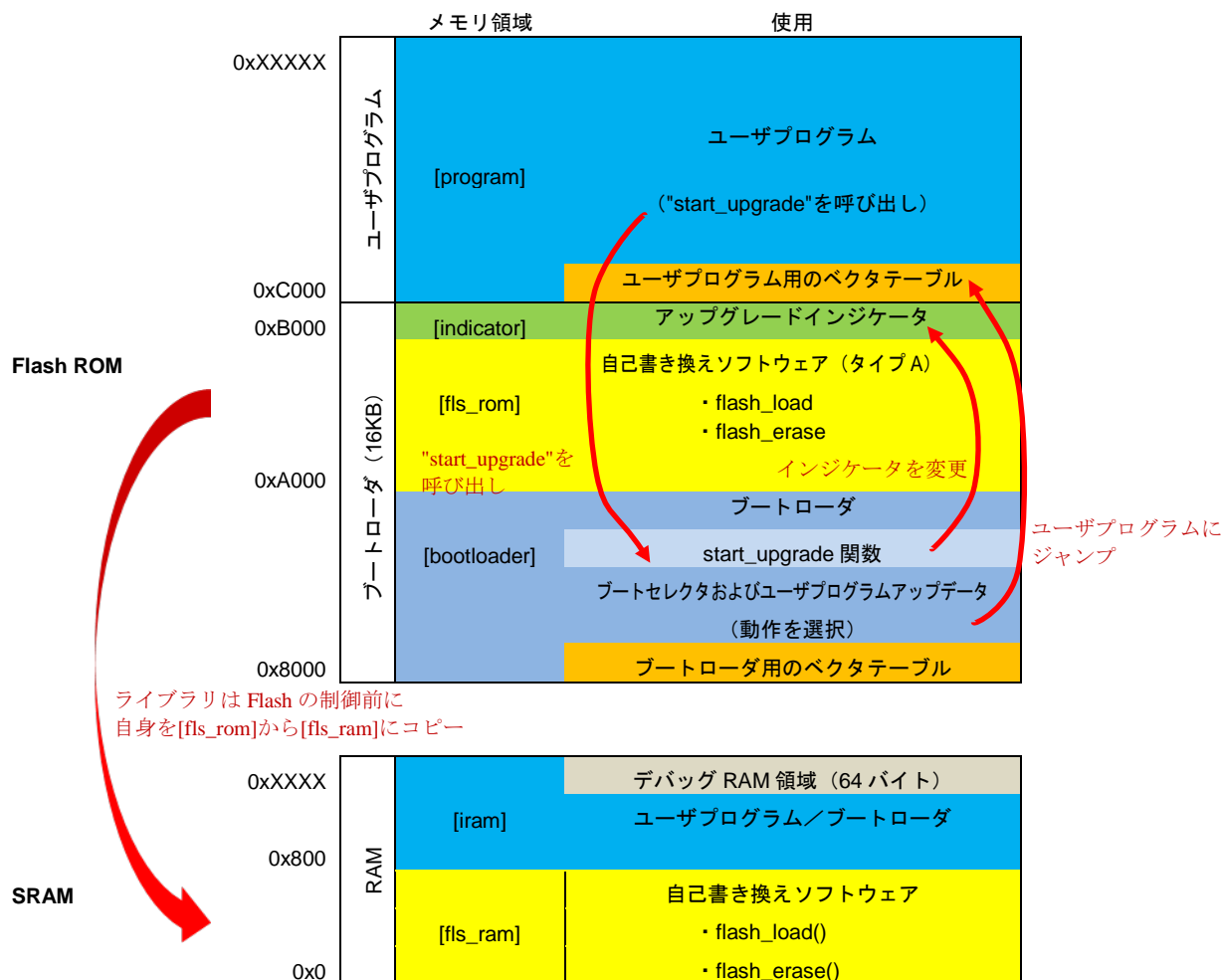
新しいプログラムをダウンロードするには、シリアルポートをサポートしたターミナルソフトウェアを使用してください。たとえばこのブートローダのテストでは「TeraTerm」を使用しています。

Tera Term : <http://ttssh2.osdn.jp/index.html.en>

PSA ファイルをダウンロードするには、以下の手順に従って TeraTerm を操作してください。

- (1) [File]メニューから [New connection] を選択し、[Serial] と使用する [Port] を指定した後、[OK] ボタンをクリックします。
- (2) [Setup]メニューから [Serial Port...] を選択します。[Serial port setup] ダイアログボックスが表示されます。
- (3) 以下のパラメータを入力します。
Port: COMx (x : 使用するポート番号)
Baud Rate: 115200 or 230400
Data: 8bit
Parity: none
Stop: 1bit
Flow control: Xon/Xoff
Transmit delay: 0msec/char, 0msec/line
- (4) [OK] ボタンをクリックします。
- (5) [File]メニューから [Send File] を選択し、PSA ファイルを指定します。「TeraTerm」は新しいプログラムをダウンロードし始めます。

3. メモリの使用



3.1 更新インジケータ ([indicator]領域)

更新インジケータのサイズは 4KB です。ただし、実際に使用されるのは 4 バイトのみで、4 バイトの値は以下の状態を示します。

状態	値
"upgrade completed" アップグレードが完了し、ユーザプログラムを実行できます。	0xAA, 0xAA, 0xAA, 0xAA
"do upgrade" ユーザプログラムがアップグレードを決定しましたが、アップグレードが開始されていません。	0x00, 0x00, 0x00, 0x00
"now upgrading" ブートローダがユーザプログラムをアップグレードしています。アップグレードは完了していません。	0xFF, 0xFF, 0xFF, 0xFF

この領域の未使用部分を使用して、アップグレードの進行状況を記録できます。

3.2 自己書き換えライブラリの予約 RAM ([fls_ram]領域)

自己書き換えライブラリはブートローダ側に含まれており、ライブラリが使用する RAM はブートローダの一部として定義されます。

3. メモリの使用

ユーザプログラム(ロードされたプログラム)がブートローダの `start_upgrade` 関数を使用する場合、ブートローダに含まれる自己書き換えライブラリが動作します。このとき、自己書き換えライブラリは作業領域として `[fls_ram]` 領域を使用します。よって、ユーザプログラムの使用する RAM が `[fls_ram]` 領域と重ならないように配置してください。

`[fls_ram]`以外の RAM は、ユーザプログラム側で使用できます。

セイコーエプソン株式会社

営業本部 デバイス営業部

東京 〒191-8501 東京都日野市日野 421-8
TEL (042) 587-5313 (直通) FAX (042) 587-5116

大阪 〒530-6122 大阪市北区中之島 3-3-23 中之島ダイビル 22F
TEL (06) 7711-6770 (代表) FAX (06) 7711-6771

ドキュメントコード : 413616000
2018年2月作成