

組み込み Linux 上のソフトウェア更新方式 (3) -差分抽出方式-

深澤 司[†] 河相 英典^{††} 小谷 亮[†] 清原 良三[†]

[†] 三菱電機株式会社 情報技術総合研究所 ^{††} 三菱電機株式会社 モバイルターミナル製作所

1 はじめに

組み込み端末のソフトウェア開発では、ソフトウェアの高機能化に伴い、搭載するソフトウェアの規模が増加している。そのため、障害のない状態で組み込み端末を出荷することは困難になっている。そこで、ネットワークを利用して不具合修正を施すソフトウェア更新方式 [1] や無線パケット通信技術を利用して不具合修正を施すソフトウェア更新方式 [2] が実現されている。

組み込み端末におけるソフトウェア更新は、短時間で完了する方が望ましい。我々は、従来のモノリシックな OS においても、ROM 上における書換え量を削減することにより、ソフトウェア更新の時間を短縮してきた [3]。一方、組み込み Linux が様々な機器で用いられている。しかしながら、組み込み Linux では、ファイルシステムを利用するため、従来のソフトウェア更新方法をそのまま使用することはできない。

本論文では、組み込み Linux 端末において更新時間の短縮を図るために、ソフトウェア更新における書換え量の削減についての検討結果を報告する。

2 組み込み Linux 上におけるソフトウェア更新の時間短縮に対する課題

ソフトウェア更新時間は、サーバからの更新データダウンロード時間と更新データを適用する時間がある。組み込み端末では、リソース面 (メモリ・CPU 等) の制約が大きいために、ROM 上での書換え時間を短縮することが重要である。ここでは、組み込み端末上における書換え量削減について着目する。

ソフトウェア更新によりデータの変更、挿入、削除が生じたときには、アドレス配置の変更が発生する。配置アドレスが変更になると、変更の影響が大域的に及んでしまうために、組み込み端末における書換え量が大きくなる。書込み量の削減方法として、以下の内容について検討した。

- (1) ファイルシステム上での書換え量削減方法
 - 緩衝領域を設置
 - ROM 上におけるファイル再配置
- (2) 実行形式ファイル上での書換え量削減方法
 - 緩衝領域を設置

3 組み込み端末における書換え量削減の実現方法

3.1 ファイルシステム上での書換え量削減方法

今回は、ファイルシステムのひとつである Cramfs[5] 領域におけるソフトウェア更新を例として検討した。差分抽出方式では、新版と旧版の Cramfs イメージから差分を抽出する。Cramfs イメージは、ディレクトリツリー構造から ROM 上の配置を決定する。そのため、新規ファイルの追加位置によって、新版と旧版のファイル位置が異なり、組み込み端末における書換え量が大きくなる。

そこで、新規ファイルが追加される際に、ディレクトリツリー構造からファイルシステムのイメージ上でのファイル順序を決定し、新規ファイルを十分な緩衝領域に配置することにより書換え量を抑制する。

3.1.1 緩衝領域の設置による実現方法

図 1 に示すように、ファイルシステム上に緩衝領域を設けることにより、局所的な修正は局所にしか影響が現れないようにする。これより、組み込み端末上での書換え量を削減できる。Cramfs 上での緩衝領域の実現方法としては、次のような選択肢がある。

- (1) 1つのパーティションで一定の複数ファイル群 (以下、モジュールと呼ぶ) 毎に緩衝領域を設ける。
- (2) ファイルシステム上で、複数のパーティション毎に緩衝領域を設ける。

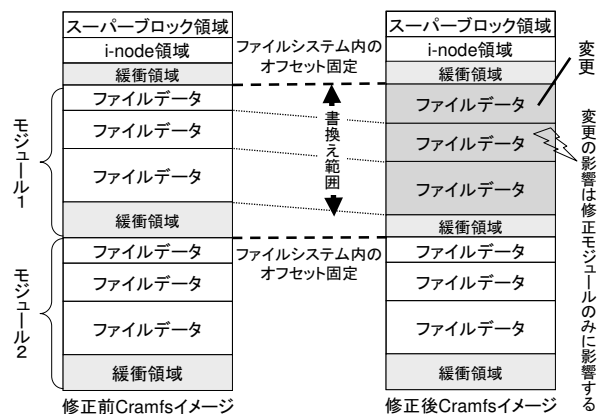


図 1: 緩衝領域を設けた書換え量削減方法

(1)、(2) は、ファイルシステム上に緩衝領域を設けることによって、更新の影響を修正モジュールだけに局所化して書換え量を削減する。(1) は、モジュールごとに緩衝領域を設けて、更新の影響を局所化している。(2) は、従来の Cramfs 作成方法を使用できるため

A Software Update Method for Embedded Linux Systems

[†] Tsukasa Fukasawa, Hideaki kawai, Akira kotani, Ryoza Kiyohara

E-mail: t-fuka@isl.melco.co.jp

Mitsubishi Electric Corp. Information Technology R & D Center ([†])

に実現は容易であるが、パーティション毎にスーパーブロック領域と i-node 領域が必要になり、ROM 消費量が大きくなる。そのため、(2) は、組み込み端末のように ROM 容量が限られている場合には好ましくない。結論として、組み込み端末における書換え量の削減方式としては、モジュールごとに緩衝領域を設ける方法が効果的である。

3.1.2 ファイルの配置による実現方法

Linux を導入することにより、組み込み端末では、ファイルシステムが存在するためにファイルの操作が容易になった。ROM 上のファイル配置を工夫することによって書換え量を削減する方式を以下に述べる。

図 2 に示すように、更新ファイル B のサイズが増加した場合には、ファイル B 以降のファイルはデータコピーを行う必要があり、更新の影響がモジュール全体に及んでしまう。そのため、組み込み端末における書換え量が大きくなる。そこで、更新ファイルを十分な緩衝領域に再配置することにより書換え量を低減する方法がある。組み込み Linux では、ファイル操作を行うファイルシステムの存在により、従来のモノリシックなソフトウェア構造のときよりもファイルの再配置操作が容易である。しかしながら、差分作成では、緩衝領域の大きさが十分でない場合に、ファイルの再配置操作を行わないようにしなければならない。結論として、新規ファイルの追加や変更ファイルサイズが増加する場合には、ファイルを再配置することにより、書換え量を削減できる。

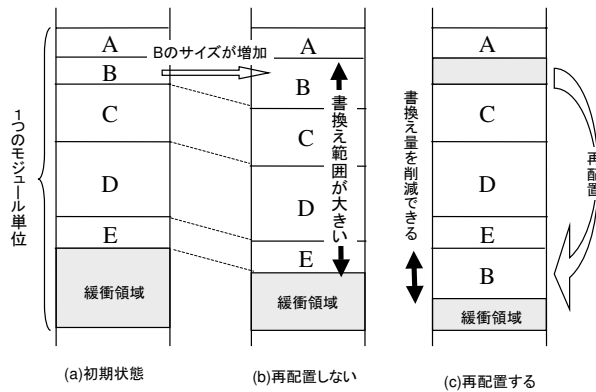


図 2: ROM 上におけるファイル再配置

3.2 実行形式のファイルにおける書換え量の削減方法

ここでは、ROM 上のファイルのサイズを考慮して、組み込み端末における書換え量の削減方法を検討する。任意に決められたモジュールサイズよりも大きな実行形式ファイルが存在した場合には、実行形式ファイルを解析して緩衝領域を設ける。

ELF(Executable and Linking Format)[4] 形式のファイルは、ELF ヘッダ、セクション、セクションヘッダテーブル、プログラムヘッダテーブルから構成される。緩衝領域をセクション毎に設けることにより、組み込み

端末における書換え量削減を実現する(図 3)。組み込み端末のように限られたメモリ領域である場合、ELF に緩衝領域を設けた場合が有効である。

よって、実行形式のファイルが任意に決められたモジュールサイズよりも大きい場合に、ELF 形式のファイルそのものに緩衝領域を設けることにより、組み込み端末における書換え量を削減できる。

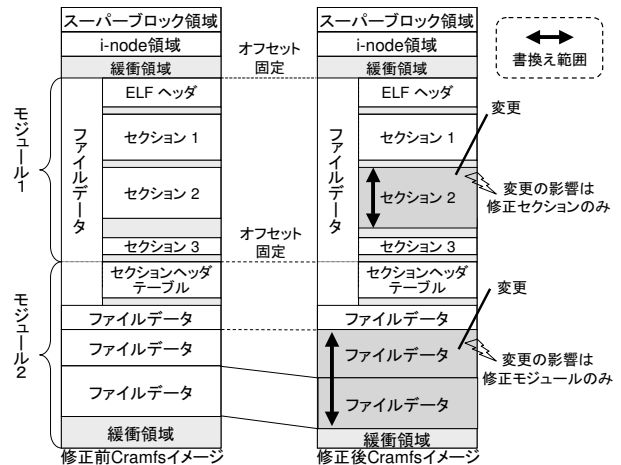


図 3: 実行形式のファイルそのものに緩衝領域を設置

4 おわりに

本論文では、組み込み Linux 端末向けソフトウェア更新における書換え量の削減方法を検討した。結果として、組み込み端末における書換え量の削減方法は、モジュール毎に緩衝領域を設けることが効果的であることがわかった。また、新規ファイルの追加や変更ファイルサイズが増加する際には、ファイルを再配置することにより書換え量を削減できることを示した。

今後の課題を以下に示す。

- 組み込み端末では、モジュール分割数を増やすと更新の影響が局所的になるが、十分なメモリが必要となるためあまり好ましくない。そこで、ファイルシステム上におけるモジュールの最適な分割数の検討が必要である。
- 修正の頻度が高いソフトウェアは、緩衝領域の前に設置することにより、書込み量の削減が期待できる。そこで、ソフトウェアの信頼度などの概念導入による緩衝領域の最適化の検討が必要である。

参考文献

- [1] BS デジタル放送の送信用受信装置、標準規格(望ましい仕様) ARIB STD-B21
- [2] 武市, 那須, 照沼他, "無線通信を利用した移動機ソフトウェアの不具合修正", 信学技報, vol.102, No.705
- [3] 清原他, 「携帯電話の SW 更新を目的としたモジュール分割に関する検討」, 情報処理学会 DI-COMO2003 4A-073
- [4] Tool Interface Standard(TIS) Committee, "Executable and Linking Format(ELF) Specification", 1995, v1.2
- [5] <http://sourceforge.net/projects/cramfs/>