

プロも使うYocto 開発環境で初体験! ラズパイ時代のレベルアップ! MyオリジナルLinuxの作り方

第22回 起動時間はじめての分析&短縮 [ご購入はこちら](#)

三ツ木 祐介

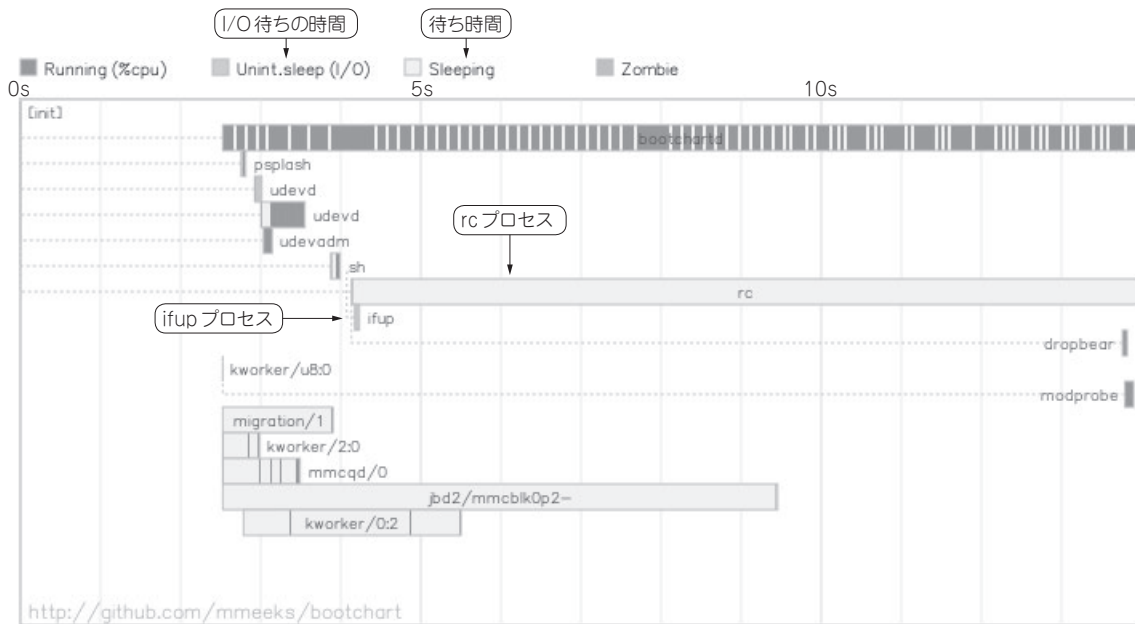


図1 ブート中にアクティブになるプロセスに注目

Linuxの起動にかかる時間を計測するブート・チャートのための環境を構築し、標準状態の起動時間を計測したのが図1です(方法は前回参照)。ブート・チャートで最初に注目すべきは、図1のrcプロセスの部分です。

起動プロセスの時間を分析

ブート・チャートを生成するbootchartdプロセスを除けば、図1ではrcプロセスが一番長く横に伸びています。しかし、ほとんどがSleeping時間、つまりほとんど待ち時間です。

さらによく見てみると、rcのサブプロセスであるifupプロセスがUninit.sleep (I/O)、つまりI/O待ちになってから、rcプロセスが待ち状態になっています。

実験1…ネットワークを無効に

ここで1つ実験をしてみます。ラズベリー・パイ3を起動し、ログインした状態で次のコマンドを実行します。

```
$ ifdown -a
$ time ifup -a
```

全てのネットワーク・インターフェースを無効にするために、ifdownコマンドを実行します。この状態でifupプロセスの実行にかかる時間を測定するためにtimeコマンドを実行します。timeコマンドでは表1に示す内容が表示されます。

実行の様子を図2に示します。この結果から、ほとんどCPU時間を使用していないにもかかわらず、実行に10秒程度の時間がかかっています。