

プロも使う Yocto 開発環境で初体験! [ご購入はこちら](#)

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23回 より便利な systemd による起動時間の計測

三ツ木 祐介

リスト1 systemdの設定

```
DISTRO_FEATURES_append = " systemd"
VIRTUAL-RUNTIME_init_manager = "systemd"
DISTRO_FEATURES_BACKFILL_CONSIDERED = "sysvinit"
VIRTUAL-RUNTIME_initscripts = ""

DISTRO_FEATURES_append = " pam"
```

前回までsysvinitの環境でbootchartを使用して起動時間を測定しました。連載第20回(2018年8月号)で触れたように、ラズベリー・パイの標準OSであるRaspbianをはじめ、最近のLinuxディストリビューションでは、INITプロセスにsystemdを使用することが増えています。

sysvinitでは設定されたラン・レベルによって、必要なプロセスをrcスクリプトと呼ばれるシェル・スクリプトによって起動します。単にスクリプトを実行していただけなので、プロセスの起動はシーケンシャルに行われます。

systemdでは、INITプロセスで行う必要がある処理をユニットという単位で分割し管理します。各ユニットには依存関係や前後関係が記述でき、それがない場合にはできるだけ並列に処理されるようになっています。

起動されたプロセスはsystemdのプロセスによって管理され、ログの収集や各プロセスの監視なども行われます。システムの都合上、止まてはいけなサービスなどはあらかじめ監視しておき、予期せず終了した場合には再起動できるようにもなっています。

このようにsystemdを使用すると、sysvinitを使用するよりも便利なが多くなっています。

Yoctoで作成したLinuxではデフォルトでsysvinitですが、systemdを使用することもできます。

起動時間を確認できる systemd-bootchart

systemdにはブート・チャートの機能として、systemd-bootchartがあらかじめ組み込まれています。

リスト2 systemd-bootchartの設定

```
IMAGE_INSTALL_append = " systemd-bootchart"
CMDLINE_append = " initcall_debug printk.time=y
                  init=/lib/systemd/systemd-bootchart"
```

sysvinitで使用したbootchart2とは別物であるため、残念ながらsysvinitで得られたものと全く同じ情報を得ることはできません。そのため起動時間について、sysvinitの環境と単純に比較することは難しくなっています。

● Yoctoでsystemd-bootchartを使えるようにする

ここではまず、Yoctoでsystemd-bootchartを使えるようにしていきます。

local.confにリスト1の内容を記述し、YoctoのINITプロセスをsysvinitではなくsystemdになるようにします。

さらにsystemd-bootchartを使用するためにリスト2の内容も追記します。

この状態でBitBakeを実行し、作成したイメージでラズベリー・パイを起動します。このとき、イーサネット・ケーブルは接続しないでおきます。

● サービス状態の確認

ログイン後、systemctlコマンドでsystemd-bootchartサービスの状態を確認します(図1)。/run/log/bootchart-20180807-0112.svgに画像ファイルが保存されています。

● 起動時間の確認

/run/logはtmpfsでマウントされているため、電源を切るとデータは消えてしまいます。そこで、/home/rootにコピーしてSDカードに保存されるようにします。

```
$ cp /run/log/bootchart-20180807-0112.svg /home/root
```

このSDカードをPCに挿入して、保存した画像ファ