

元祖mbedで始めるネットワーク・カメラ その②…専用ウェブ・サーバづくり

森岡 澄夫

リスト1 ウェブ・サーバ処理の基本構成

```
int main() {
    EthernetInterface eth; // ネットワーク・オブジェクト
    HTTPServer svr; // HTTPサーバ・オブジェクト

    // ネットワーク接続を確立する
    eth.init(); // IPアドレス指定, DHCPのどちらも可
    eth.connect();

    // HTTPサーバ立ち上げ
    svr.addHandler<...>(…); // 扱うリクエストの定義
    svr.start(ポート名, &eth);

    while (1) { // メイン・ループ
        svr.poll(); // HTTPリクエストを処理
        ...
    }
}
```

mbed LPC1768をネットワークに接続する

パソコンのブラウザからアクセスできるようにする

ブラウザで表示するデータを送り出す

本章では、第4章で作成したカメラをアップグレードし、ネットワークにつなげます。

カメラ用ウェブ・サーバの基本プログラム

まず、mbed LPC1768上にある画像ファイルやHTMLファイルをブラウザから見られるようにするためのプログラムを作ります。

● mbed ライブラリを利用してサッ

mbed LPC1768上でのウェブ・サーバ立ち上げは、mbed ライブラリを使うだけなので簡単にできます。ソース・コードの概要をリスト1に示します。

▶サーバの立ち上げ手順

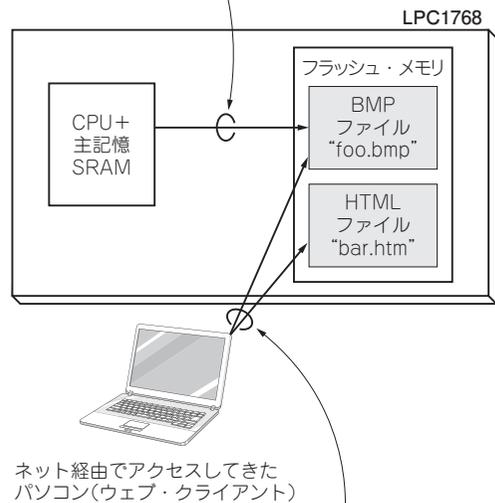
(1) EthernetInterface オブジェクトと HTTP Server オブジェクトを用意し、ネットワーク接続を確立する。

IPアドレスは直接指定することもできますし、DHCPを使うこともできます。

(2) HTTPリクエスト処理のハンドラを設定したうえで、HTTPサーバを起動する。

サーバ起動後は、メイン・ループの中で、HTTPリクエストを受信し処理するためのポーリングを繰り返します。

mbedのCPUから見たときのディレクトリ名を、LocalFileSystem 宣言時に指定する。例えば
例：LocalFileSystem local("A") で/A/に見える



ウェブ・クライアントからルートを見たとき、実際にはどのディレクトリに相当するかを、HTTPサーバ立ち上げ時に指定する。例えば、

```
HTTPFsWithRequestHandler::mount("/A/", "/");
サーバ・オブジェクト.addHandler<HTTPFsWithRequestHandler>("/");
```

とするとURL http://IPaddress/ に A中のファイルがあるように見える

図1 LPC1768内蔵フラッシュに保存した画像ファイルやHTMLファイルがウェブ・ブラウザから見られるしくみ

● 画像保存用ディレクトリをHTTPサーバのルートに指定する

プログラミング時に、ファイルを置くディレクトリ名をきちんと把握・管理する必要があります。これを整理したのが図1です。

メイン関数のグローバル変数としてファイル・システムのインスタンスを作成し、それを使って撮影画像をビットマップ・ファイルとして保存します。ウェブ・サーバのルート・ディレクトリにファイル・システムのルート・ディレクトリを指定すれば、ブラウザ