

CITIZEN

モバイルプリンター

iOS Bluetoothプログラミングマニュアル

Bluetoothに関する補足

Ver. 1.01

シチズン・システムズ株式会社

目次

1. Bluetooth接続とペアリング
2. XCodeのBluetooth用設定
3. 読み込み機能のプログラミング

履歴

バージョン	対象ライブラリ	説明
1.00	1.46	公開
1.01	1.60	Amd64 機能を追加 ライブラリバージョン 1.60 から各メソッドの戻り値が Int から long に変更されました。 以前のバージョンのライブラリからアップデートされる場合、Xcode が Warning メッセージを出力します。アプリケーション側のソースコードの修正をお願いします。 プリンターをエラー状態で Wi-Fi 接続を試みると、待ち状態に陥る現象を改善 openPort を 2 回実行後 closePort する接続しなくなる不具合を修正

1. Bluetooth 接続とペアリング

CMP-20/30とiOSデバイスがBluetoothの通信をするには、iOSデバイスからプリンターが発見でき、iOSデバイス上でペアリングをする必要があります。(以下の画面はiPod Touch / iOS6での例です)

1-1、設定からBluetoothを選びます。



1-2、BluetoothをONにします。

これにより周辺で見つかったBluetoothデバイスが表示されます。

CMP-20/30は、MOBILE PRINTERと表示されるので、それが見つかったら、選択します。

選択したプリンターが既にペアリングされている場合は、そのまま接続が確立します。



1-3、選択したプリンターがペアリングされていない場合、PIN入力の画面が出ます。
ここで、PINコード（初期値は0000）を入力し、ペアリングボタンを押します。



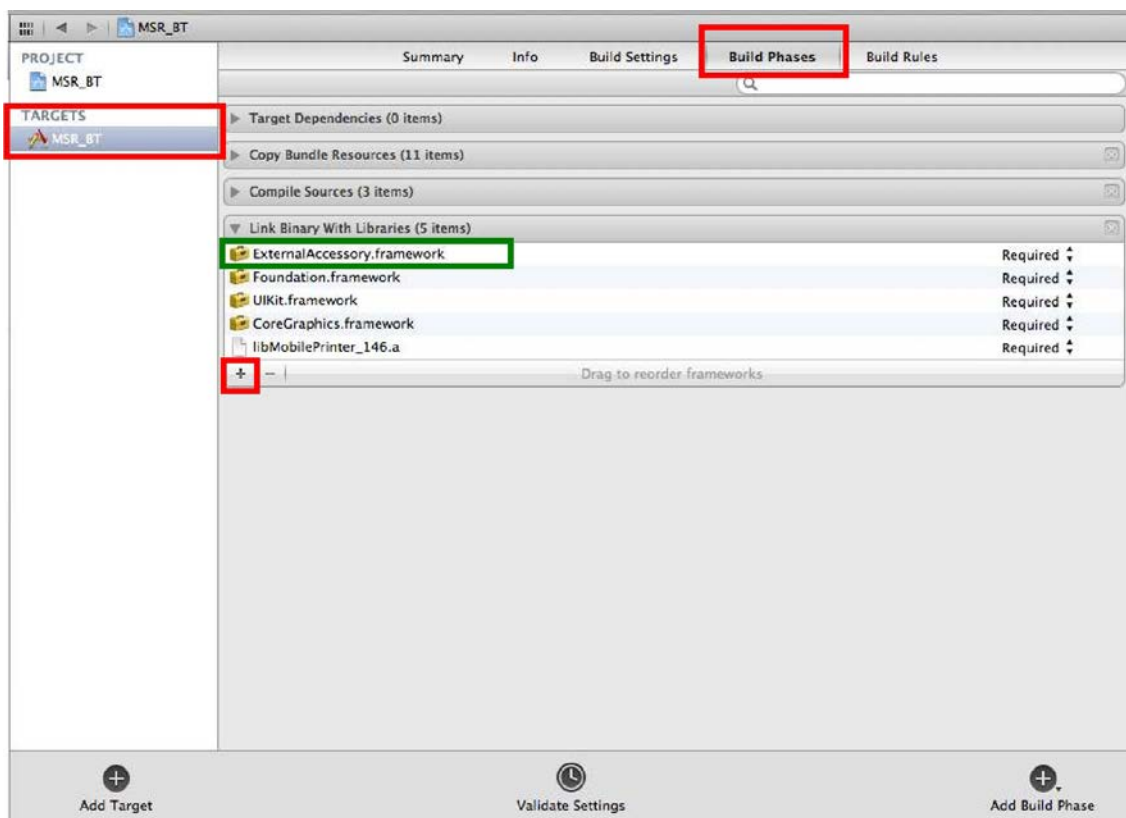
1-4、接続が確立すると、「プリンターの状態が接続されました」と表示されます。
また、通知バー上のBluetoothのアイコンがクリアに表示されます。
接続後、「Appがインストールされていません」というメッセージが出るがありますが、無視ボタンを押してください。



2. XCode の Bluetooth 用設定

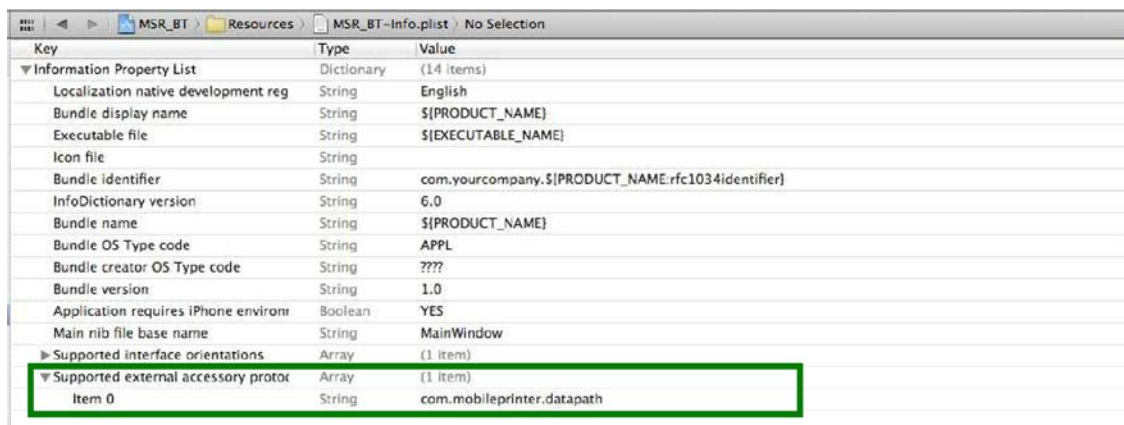
アプリにBluetooth通信機能を追加するためには、iOS用アプリ開発ツールであるXCodeのプロジェクトに External Accessory Frameworkを追加し、アプリがサポートするプロトコル名をInfo.plistファイル内で宣言する必要があります。

プロジェクトエディタ上でTARGETS内のターゲットアプリを選び、Build Phasesタブをクリックし、Link Binary With Librariesを展開し、「ExternalAccessory.framework」を追加します。



プロジェクトナビゲーター上で、[Project名]-Info.plist ファイルを開きます。

右のプロパティリスト上で行を追加 (Control + Clickから “Add Row”) し、“Supported external accessory protocols”を選択します。“Supported external accessory protocols”を開き、Item 0の値に「com.mobileprinter.datapath」と入力します。



3. データ読み込み機能のプログラミング

下記のデータ読み込み機能のサンプルをご覧ください。(printerCheck, readMSR)
詳細情報が必要な場合は、サンプルコードを参照ください。

```
#import "EABluetoothPort.h"

....

// Add Observer in initiation function [ ex: viewDidLoad ]
[[NSNotificationCenter defaultCenter] addObserver:self selector:@selector(dataReceived:)
name:EADSessionDataReceivedNotification object:nil];
[[EAAccessoryManager sharedAccessoryManager] registerForLocalNotifications];
....

// Remove Observer in termination function [ ex: didReceiveMemoryWarning ]
[[NSNotificationCenter defaultCenter] removeObserver:self];
....

// Observer
- (void) dataReceived:(NSNotification *) notification
{
    EABluetoothPort * sessionController = (EABluetoothPort *)[notification object];
    uint32_t bytesAvailable = 0;
    NSMutableData * result = [[NSMutableData alloc] init];
    while ((bytesAvailable = [sessionController readBytesAvailable]) > 0)
    {
        NSData *data = [sessionController readData:bytesAvailable]; if
        (data)
        {
            [result appendData:data];
        }
    }
    // 実行する内容 Something to do.
}
```