本ドキュメントは Cypress (サイプレス) 製品に関する情報が記載されております。本ドキュメントには、「MB」から始まるシリーズ名，品名およびオーダ型格が記載されておりますが、これらはすべて「CY」から始まるシリーズ名，品名およびオーダ型格として、新規および既存のお客様に引き続き提供してまいります。

オーダ型格の調べ方について

1. www.cypress.com/pcnにアクセスしてください。
2. SEARCH PCNS フィールドに、オーダ型格などのキーワードを入力し、「Apply」をクリックしてください。
3. 該当するタイトル(Title)をクリックしてください。
4. 「Affected Parts List」ファイルを開いてください。
   当該ファイルに記載されている各種変更情報をご利用ください。

詳しいお問い合わせ先

Cypress 製品およびそのソリューションの詳細につきましては、お近くの営業所へお問い合わせください。

サイプレスについて

サイプレスは、世界で最も革新的な車載や産業機器、スマート家電、民生機器および医療機器製品向けに、最先端の組み込みシステム ソリューションを提供するリーディング カンパニーです。サイプレスのマイクロコントローラーや、アナログ IC、ワイヤレスおよび USB ベースのコネクティビティ ソリューション、高い信頼性と高性能を提供するメモリ製品は、各種機器メーカーの差異化製品の開発と早期市場参入を支援します。サイプレスは、ベストクラスのサポートと開発リソースをグローバルに提供することで、彼らが従来市場を破壊しまったく新しい製品カテゴリを歴史的なスピードで市場投入できるよう支援します。詳細はサイプレスのウェブサイト (japan.cypress.com) をご覧ください。

AN204461

FM4 無線 LAN モジュール連携による WLAN ソリューション デモ操作マニュアル

関連製品ファミリ:	シリーズ名	品種型格
	MB9B160R	MB9BF168M/N/R
	MB9B360R	MB9BF368M/N/R
	MB9B460R	MB9BF468M/N/R
	MB9B560R	MB9BF568M/N/R

本デモ操作マニュアルでは、FM4 ファミリの高速処理と SD カードインタフェース機能を活用して、無線 LAN モジュールを制御し、PC やスマートフォン・タブレット等との無線 LAN 接続を行うデモの操作について説明します。

Contents

1 はじめに 1
2 無線 LAN デモシステム 2
2.1 システム構成 2
2.2 無線 LAN とは? 3
2.3 デモシステム概要 7
3 ハードウェア概要 15
3.1 ハードウェア構成 15
3.2 詳細仕様 17
4 ソフトウェア概要 20
4.1 ソフトウェア構成 20
4.2 詳細仕様 25
4.3 デモソフトウェア 41
5 その他 45
6 改訂履歴 46
セールス, ソリューションおよび法律情報 47

1 はじめに

本デモ操作マニュアルでは、FM4 ファミリの高速処理と SD カードインタフェース機能を活用して、無線 LAN モジュールを制御し、PC やスマートフォン・タブレット等との無線 LAN 接続を行うデモの操作について説明します。

FM4 の高速処理により、FM4 と無線 LAN モジュールのみのシンプルな構成で、TCP/IP プロトコルスタック処理と Web サーバ機能を無線 LAN 接続環境で実現できます。

ネットワーク接続されていない民生機器や、有線 LAN にてネットワーク化していた産業機器を無線 LAN 化する場合に、低コストで実現するネットワーク環境整備が課題となっています。

その課題を解決するために、ネットワーク環境整備に伴う施設コストや部材コストをできる限り抑えた状態にし、アプリケーションに最適なシステム性能・機能を検討する必要があります。

本ソリューションが、この課題解決のアイデアとしてご検討いただければ幸いです。

本デモ操作マニュアルで説明する内容は、FM4 を搭載したボード(SK-FM4-U120-9B560)に、太陽誘電株式会社(以降、太陽誘電)製無線 LAN モジュール評価ボード(WBSAAVDX7)*を SD カードスロットに装着してネットワーク通信を行うソリューションです。

SK-FM4-U120-9B560 に搭載されている FM4 の型格は下記のとおりです。

・型格：MB9BF568R

*本書で使用している太陽誘電製無線 LAN モジュールの写真は、実際のものと異なります。

無線 LAN モジュールの詳細については、個別に太陽誘電株式会社にご相談ください。

本書に記載されている URL は Rev. ** 時点のものです。URL は予告なく変更されることがあります。

2 無線 LAN デモシステム

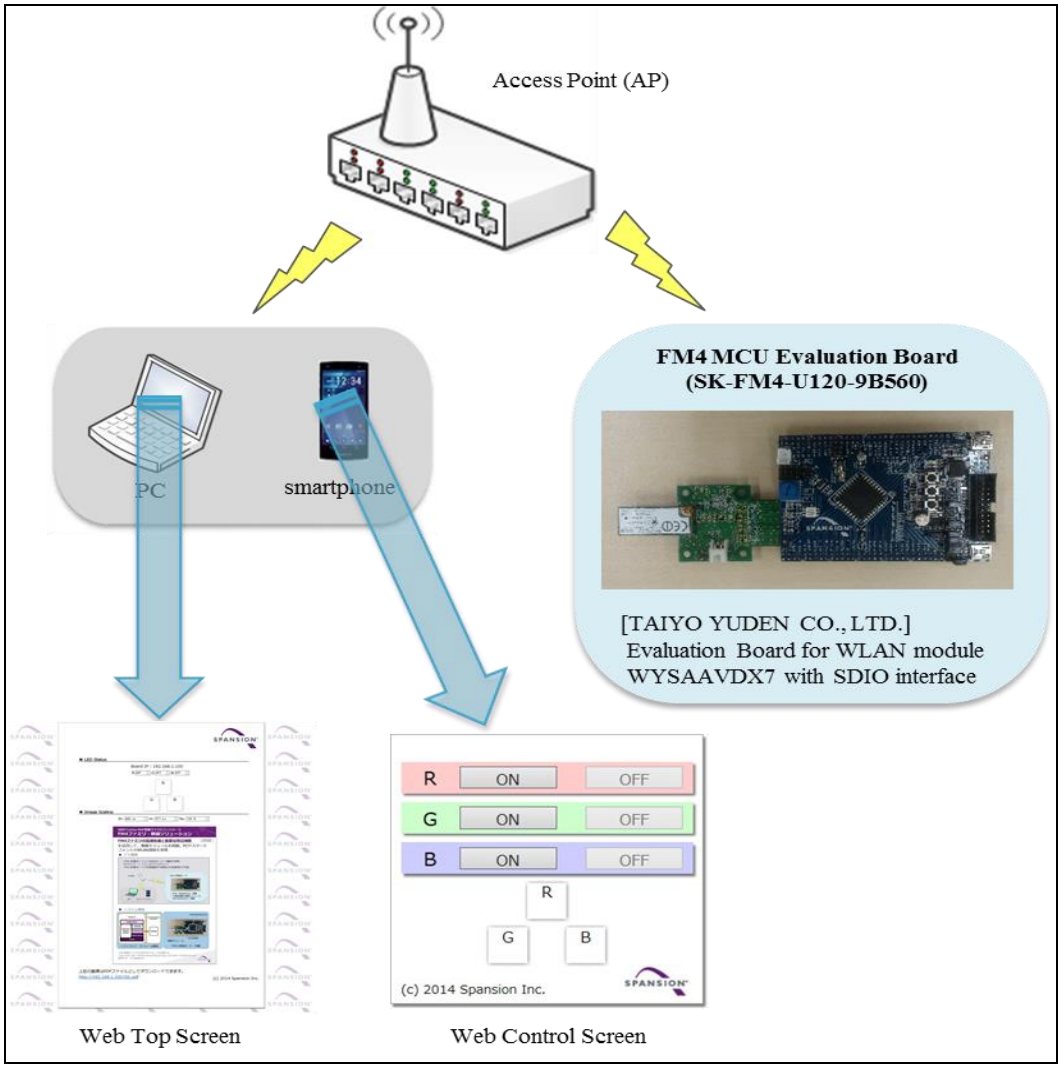
2.1 システム構成

無線 LAN デモシステムのシステム構成を以下に示します。

概要

下図のように、FM4 評価ボード＋無線 LAN モジュールと無線 LAN 機能を備えた端末(下例：FM4MCU Evaluation Board)が、アクセスポイントを介して無線 LAN で接続します。この FM4 評価ボード＋無線 LAN モジュール上で動作する Web サーバに、Web ブラウザが動作する端末(下例：PC, Smartphone)からネットワーク接続を行います。これにより、このデモシステムでは、コンテンツの表示、ファイルのダウンロード、FM4 評価ボードの制御を行えます。

図 1. 無線 LAN デモシステム構成



2.2 無線 LAN とは?

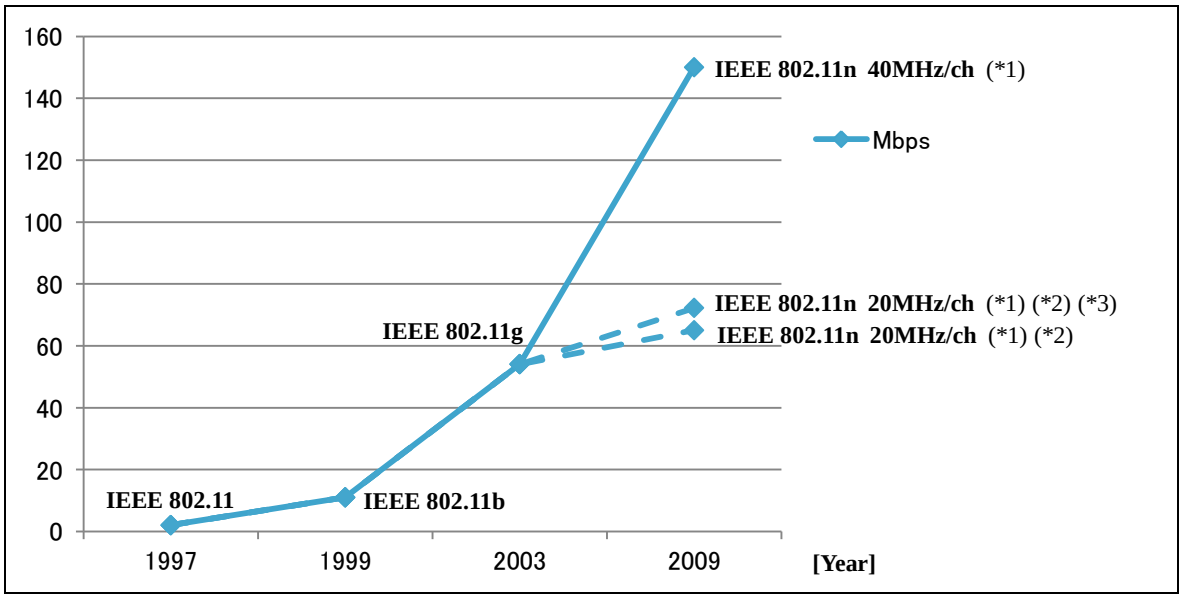
無線 LAN について説明します。

無線 LAN は、データの送受信を無線通信で行う LAN(Local Area Network)の名称です。

現在では、無線 LAN と言えば IEEE 802.11 関連規格と読み替えても過言ではないほど IEEE 802.11 関連規格の製品が普及しています。

無線 LAN の伝送速度は、技術進歩により飛躍的に向上しています。

図 2. IEEE 802.11 規格の変遷



*1: IEEE 802.11n は、MIMO(アンテナ)数により伝送速度が増加します。

*2: IEEE 802.11n 20MHz 時は GI (ガードインターバル) の値によって転送速度が異なります。

*3:本無線 LAN デモシステムでは、20MHz/ch GI=400ns の設定としております。

本無線 LAN デモシステムで対応している無線 LAN の伝送規格は以下です。

- IEEE 802.11b
- IEEE 802.11g
- IEEE 802.11n

本無線 LAN デモシステムで対応している無線 LAN のセキュリティの規格は以下です。

- IEEE 802.11 (WEP)
- IEEE 802.11i draft (WPA-PSK)
- IEEE 802.11i (WPA2-PSK)

無線 LAN は、2.4GHz 帯を利用しますが、各国によって利用可能なチャンネル数が異なります。

本無線 LAN デモシステムでは、米国等での利用を考えて、11ch を利用するように制御しています。

2.2.1 接続方式

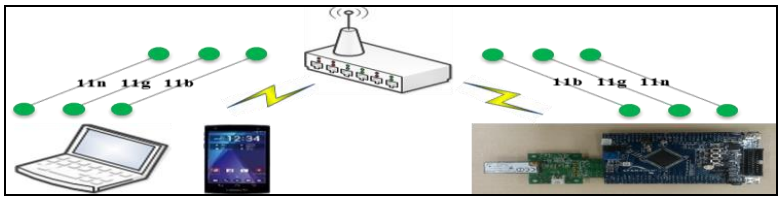
無線 LAN の接続方式として、本デモシステムで使用可能なモードを説明します。

[インフラストラクチャーモード]

アクセスポイントと呼ばれる中継機器を経由して無線 LAN 端末機器の間のデータ通信を行う方式です。

つまり、本方式では、無線 LAN 端末機器はデータ通信を行う相手の存在に関わらず、最初、アクセスポイントと接続し、無線の伝送路を確立します。

図 3. インフラストラクチャーモード



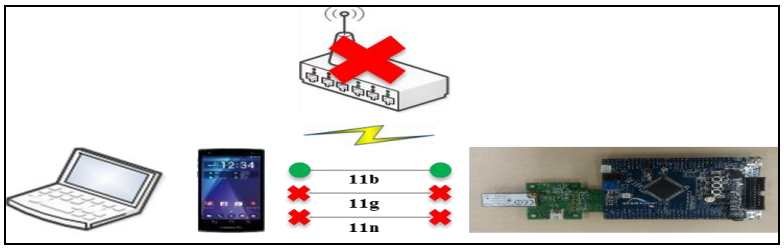
[アドホックモード]

無線 LAN 端末の機器同士で直接データ通信を行う方式です。

つまり、本方式では、アクセスポイントの設置は不要です。

このモードは、11b までの規格です。11g および 11n の規格では規定が無く、ベンダ依存の機能とした利用です。

図 4. アドホックモード



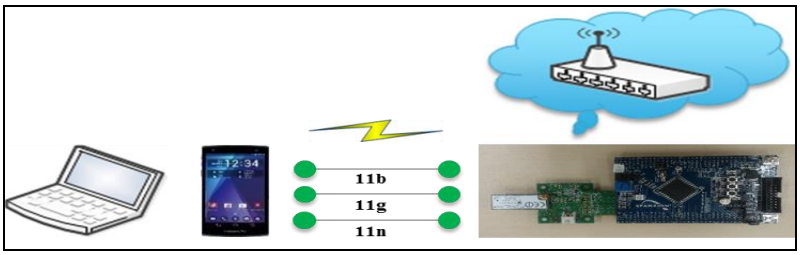
[Micro AP モード]

無線 LAN 端末機器がアクセスポイントの機能を持ち、P2P で接続して直接データ通信を行う方式です。

このモードの名称は、各社さまざまですが本書内では Micro AP モードとします。

本方式では、アクセスポイントの設置は不要です。

図 5. Micro AP モード



2.2.2 暗号方式

無線 LAN のセキュリティにおける暗号方式として、本デモシステムで使用可能なモードを説明します。

[WEP]

WEP は、暗号アルゴリズムとして RC4 をベースとした暗号方式です。現在では WEP に深刻な脆弱性が発見されているため、使用するべきではありません。

WEP には 40bit モードと 104bit モードがあり、鍵長はそれぞれ固定です。(それぞれ 5 文字、13 文字)

[TKIP]

TKIP は、暗号アルゴリズムとして RC4 をベースとした暗号方式です。WEP の脆弱性を修正していますが、暗号アルゴリズムとしては WEP と同等です。利用可能な場合は、次の項目に挙げた AES-CCMP を使用すべきです。

ユーザが指定した任意の長さのパスフレーズ (本無線 LAN デモシステムでは最大 32byte) から PMK (256bit) を生成し、鍵の素材として使用します。

[AES-CCMP]

AES- CCMP は、暗号化アルゴリズムとして AES をベースとした暗号方式です。
AES は、米国の標準暗号アルゴリズムとして規定されている方式です。

無線 LAN 機器の設定画面では、”CCMP”、”AES-CCM”、または単に”AES”と表記されることも多くあります。

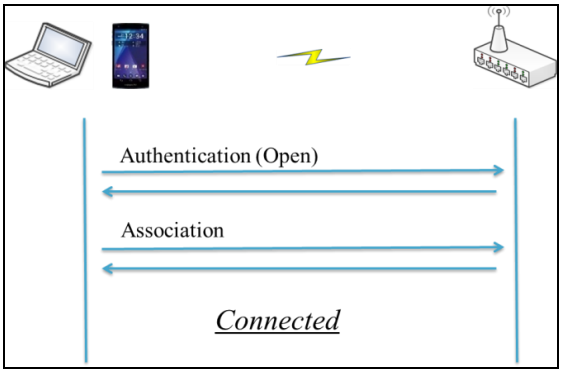
AES-CCMP でも、パスフレーズ・PMK の生成方法は TKIP と同じです。

2.2.3 認証方式

無線 LAN のセキュリティにおける認証方式として、本デモシステムで使用可能なモードを説明します。

[Open 認証 (認証なし)]

図 6. Open 認証 (認証なし)



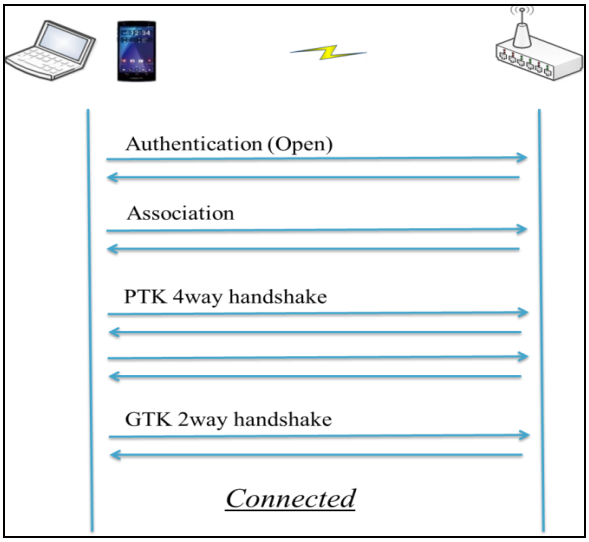
本方式は、Open System 認証とも呼ばれます。無線 LAN 機器の認証要求に対して、かならず認証成功を返却します。つまり、実質は認証なしです。

なお、暗号方式に WEP を使用する場合は、Open 認証です。古くは WEP 向けの認証方式として Shared-key 認証が規定されていましたが、本方式に中間者攻撃の脆弱性が見つかり、かえってセキュリティが低下するため、現在では認証なしが推奨されています。(ただし、WEP を使用すること自体が推奨されていません)

このため、本無線 LAN デモシステムでは、Shared-key 認証には対応していません。

[WPA-PSK]

図 7. WPA-PSK

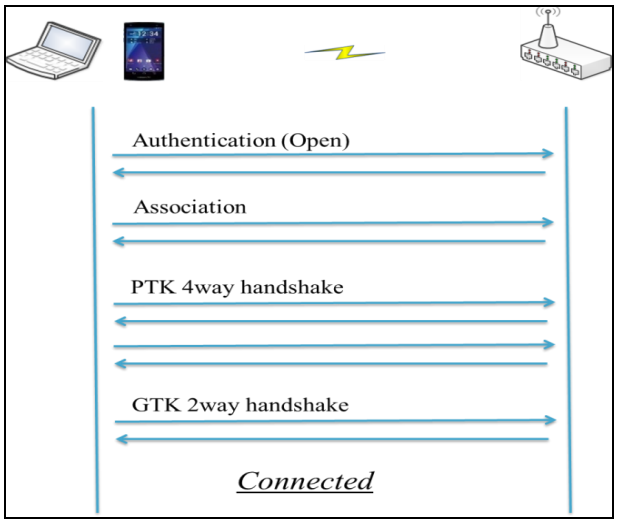


WPA-PSK は、IEEE 802.11i draft で公開された認証方式です。WPA2 が規定されるまでの暫定措置として定められた方式です。

暗号方式は TKIP / AES-CCMP から選択可能です。WPA- PSK では TKIP のサポートは必須となっています。一方、AES-CCMP のサポートは必須ではありません。

[WPA2-PSK]

図 8. WPA2-PSK



WPA2-PSK は、IEEE 802.11i で策定された認証方式です。暫定措置であった WPA- PSK を正式に規定してもので基本的には WPA-PSK と変わりませんが、定義値が変わっているため互換性はありません。

また、WPA2-PSK では AES- CCMP のサポートが必須となっており、逆に TKIP のサポートは必須ではなくなりました。

とくに対向機器の制約がなければ、WPA2-PSK + AES-CCMP の利用を推奨します。

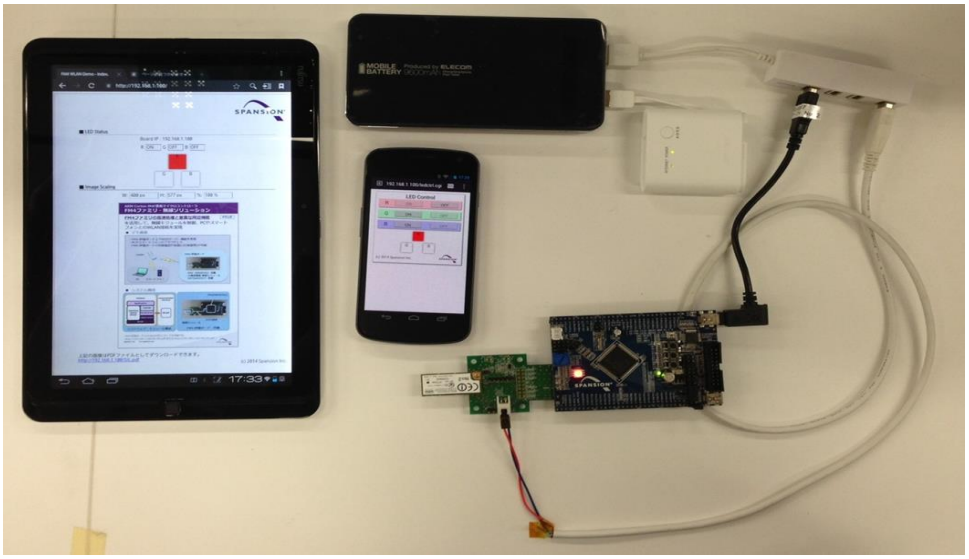
2.3 デモシステム概要

デモシステムの概要として、デモシステムの構成、設定手順、動作紹介について示します。

本章では、FM4 評価ボードにデモプログラムが書き込まれている前提で説明します。

デモシステムの構成

図 9 デモシステムの構成



機材一覧

表 1. デモシステム機材一覧

No.	名称	メーカ	数量	内容
1	FM4 評価ボード	Cypress	1	FM4 MCU と SD カードスロットが搭載された FM4 評価ボードです。(SK-FM4-U120-9B560)
2	無線 LAN モジュール評価ボード	太陽誘電	1	SDIO インタフェースを備えた無線 LAN モジュール評価ボードです。(WBSAAVDX7)
3	アクセスポイント	BUFFALO	1	DHCP サーバ機能を持つ無線 LAN アクセスポイントです。(WMR-300)
4	バッテリ	ELECOM	1	FM4 評価ボード、無線 LAN モジュールおよび、アクセスポイント用のバッテリです。(DE-M01L-3530BK)
5	USB ケーブル	-	1	FM4 評価ボードへの 5V 給電用の[USB TypeA オス]-[USB Mini- B オス]仕様 USB ケーブルです。
6	専用電源ケーブル	-	1	無線 LAN モジュールへの 5V 給電用の専用電源ケーブルです。
7	USB ハブ	-	1	FM4 評価ボード、無線 LAN モジュールへの 5V 給電用の USB ハブです。
8	Web ブラウザ端末 (Android タブレット)	-	1	デモ用コンテンツ表示用です。
9	Web ブラウザ端末 (スマートフォン)	-	1	デモ用コンテンツ表示用です。

注意事項: No.3～No.9 の機材は、特にメーカなどは問われません。

Web コンテンツ

FM4 評価ボードに設定されている Web コンテンツの URL 一覧です。

表 2. URL 一覧

ページ概要	URL
メイン	http://192.168.1.100/index.html
LED 操作	http://192.168.1.100/ledctrl.cgi
PDF	http://192.168.1.100/SIL.pdf

2.3.1 デモの事前設定

デモシステムを動作させるために、デモプログラムで既定されている無線設定と IP アドレス設定をアクセスポイントに事前に設定してください。

以下にアクセスポイントの設定と Web ブラウザ端末の設定について説明します。

＜アクセスポイントの設定＞

【無線設定】

デモプログラムの無線設定を以下に示します。

表 3. 無線設定

項目	内容
チャンネル	自動
SSID	FM4-WLAN_demo
認証	なし
暗号化	なし

この設定に合わせて、アクセスポイントの設定画面より、設定してください。

注意事項：この設定は、暗号化設定されていませんので、SSID を選択するだけで接続可能です。

【DHCP 機能設定】

Web ブラウザ端末への IP アドレスを付与するために、DHCP 機能を有効にしてください。

設定内容を以下に示します。

表 4. DHCP 機能の設定

開始 IP	台数	リース期間
192.168.1.101(※)	64 台	1 時間

※: 192.168.1.100 は FM4 評価ボードで使用しているため、除外されるように設定してください。

【IP アドレス設定】

デモプログラムでは、FM4 評価ボードの IP アドレスを 192.168.1.100 と固定しています。

そのため、アクセスポイントは FM4 評価ボードの IP アドレスに合わせたネットワークグループの IP アドレスを設定する必要があります。

各機材の IP アドレスは以下を想定しています。

表 5. IP アドレスの設定

機材名	IP アドレス
アクセスポイント	192.168.1.1
FM4 評価ボード	192.168.1.100
Web ブラウザ端末	DHCP により取得

＜Web ブラウザ端末の設定＞
DHCP 機能を有効にしてください。

2.3.2 デモの起動方法

デモの起動手順を以下に示します。

＜注意事項＞

必ず以下の手順で電源を投入してください。

手順を実施する前では、バッテリにはアクセスポイントと USB ハブは未接続状態にしてください。

合わせて USB ハブには USB ケーブルと専用電源ケーブルは未接続状態にしてください。

1. アクセスポイントの電源投入
   - バッテリを Power ON して、アクセスポイントをバッテリに接続することで電源を入れてください。
   - Web ブラウザ端末から、アクセスポイントに事前設定した SSID を検出してください。
   - Web ブラウザ端末を上記の SSID のアクセスポイントに接続してください。
2. 無線 LAN モジュールの電源投入
   - FM4 評価ボードの SD カードスロットに無線 LAN モジュールを接続してください。
   - USB ハブに専用電源ケーブルを接続してください。
   - バッテリに USB ハブを接続し無線 LAN モジュールの電源を入れてください。
3. FM4 評価ボードへの電源投入
   - USB ハブに USB ケーブルを接続した後、FM4 評価ボードの電源を入れてください。
   - 初期化中は評価ボードの LED が赤色で点灯状態になります。

     注意事項：LED が赤色で点滅している場合は、初期化が完了し、アクセスポイントを探している状態です。この状態が長く続く場合は、アクセスポイントの設定を見直してください。
   - Web サーバが起動し、Web ブラウザ端末からアクセス可能な状態になると、FM4 評価ボードの LED が緑色で点灯⇒消灯を 5 回繰り返して、消灯状態になります。
4. Web ブラウザ端末からのアクセス
   - FM4 評価ボードがアクセス可能な状態になったことを確認してから Web ブラウザ端末よりアクセスしてください。
   - アクセス可能な URL については、「表 2. URL 一覧」を参照してください。

2.3.3 デモの動作紹介

デモには以下の 3 つのデモがあります。

Demo 1 ： LED ステータス変更

Demo 2 ：画像拡大/縮小

Demo 3 ： PDF ダウンロード

各デモについて示します。

【デモ画面構成】

図 10. メイン画面

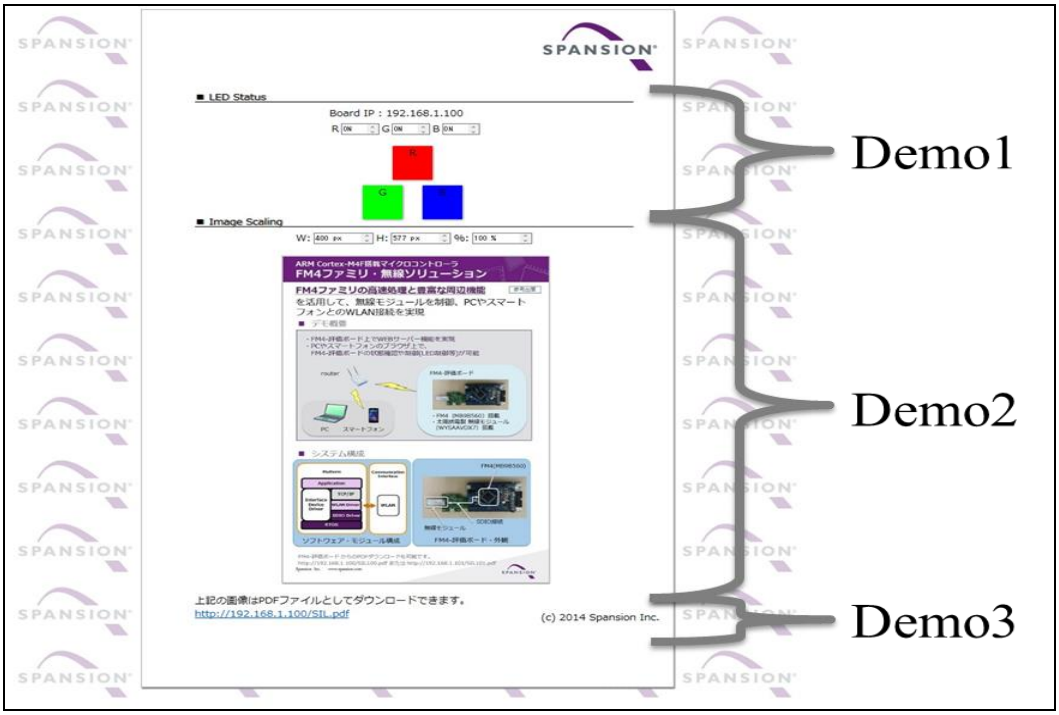
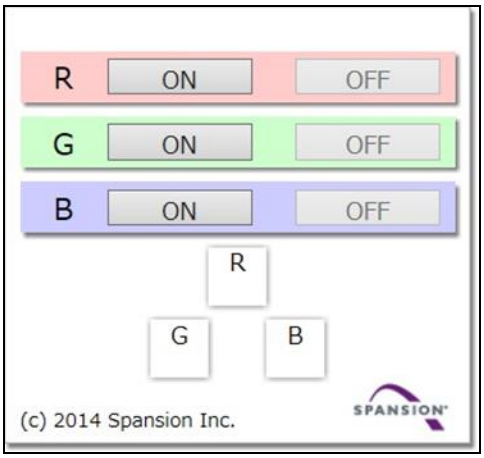


図 11. LED 操作画面



【Demo 1 ： LED ステータス変更】

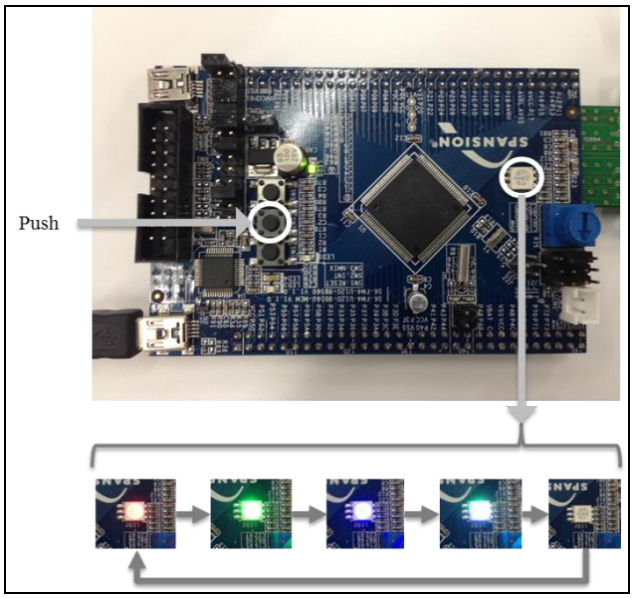
Demo 1 は、LED の点灯ステータスを WEB ブラウザ上に反映します。

LED を点灯/消灯させるには 2 つの方法で実行可能です。

1　FM4 評価ボードのスイッチによる操作

- FM4 評価ボード上のスイッチ「SW2」(3 つ並んでいるスイッチの中央) を押すたびに、以下のパターンで点灯/消灯を繰り返します。

図 12. FM4 評価ボードのスイッチによる操作

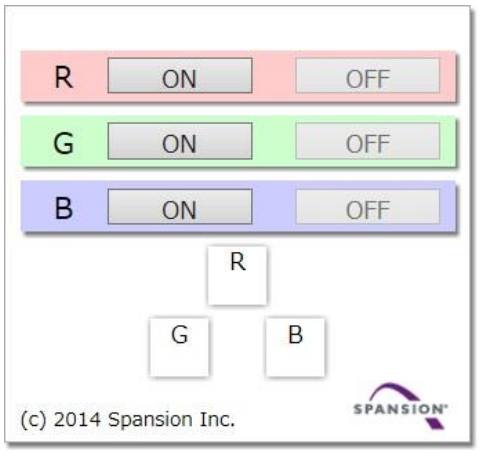


注意事項：SW2 の横にある SW1 は”Reset”です。誤って SW1 を操作してしまった場合は、起動手順に従って再度起動してください。

2 LED 操作ページによる操作

- URL 一覧の「LED 操作」ページからボードの LED を操作できます。

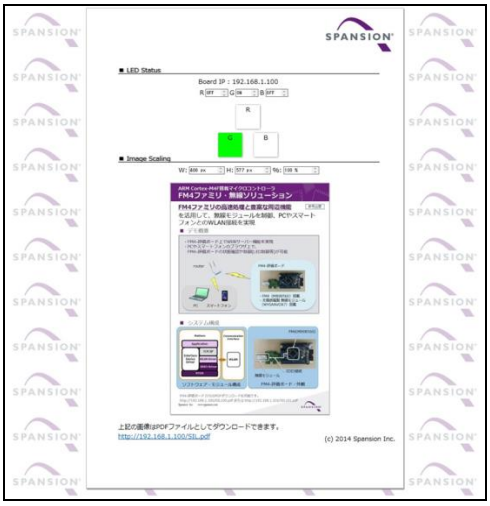
図 13. LED 操作ページによる操作



- 各色の ON のボタンをクリックすると、対応する色の LED が点灯します。(組合せは自由です。)
- 一度 ON にすると、ON のボタンは操作できなくなり、代わりに OFF のボタンが操作できるようになります。
- LED を消灯する場合は、OFF のボタンをクリックしてください。

操作による WEB ブラウザ画面例を示します。

図 14. Green を ON にした場合の画面例



【Demo 2 ：画像拡大/縮小】

Demo 2 は、デモ用パンフレットの表示をボード上のアナログ入力から画像の拡大/縮小を行います。

画像の拡大/縮小は、評価ボード上の”青色のダイヤル”を操作することで、アナログ入力値の状態に合わせてブラウザ上の画像が拡大/縮小されます。

図 15. 画像拡大/縮小の操作

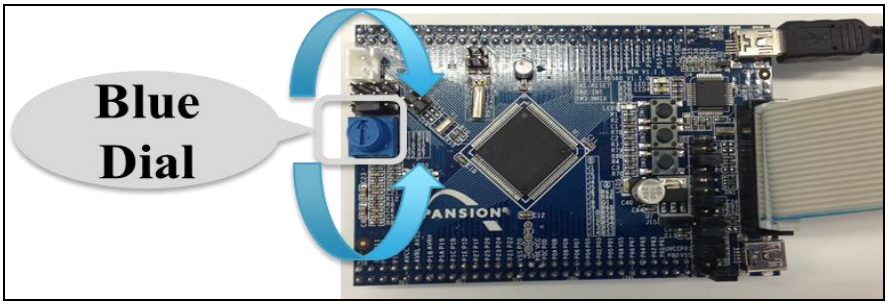
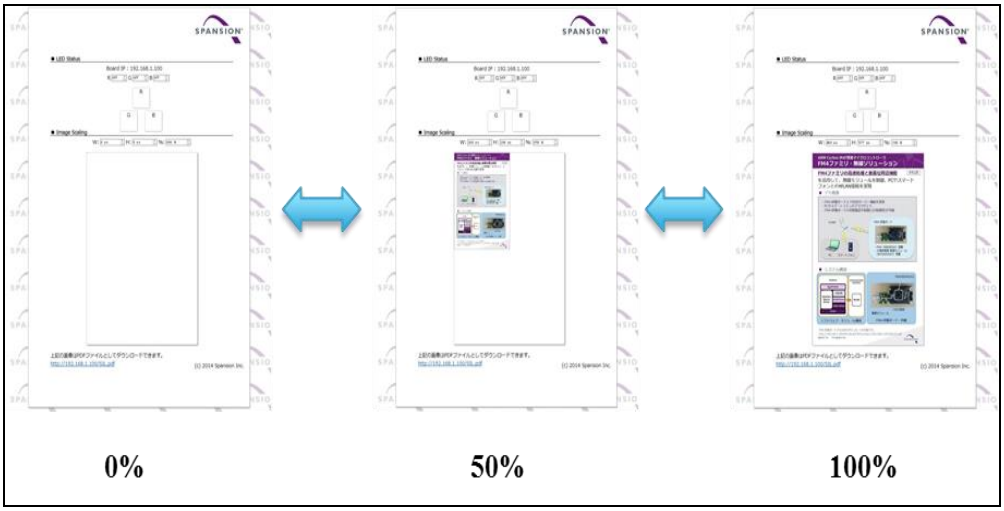


図 16. 画像拡大/縮小の画面遷移



【Demo 3 ： PDF ダウンロード】

Demo 3 は、Demo 2 で表示されている画像の PDF ファイルをダウンロードできます。

ダウンロードした PDF ファイルを閲覧するために、Web ブラウザ端末には PDF ファイルを閲覧するソフトが必要です。

Web ブラウザ端末の無線 LAN 設定については、「表 3. 無線設定」を参照してください。

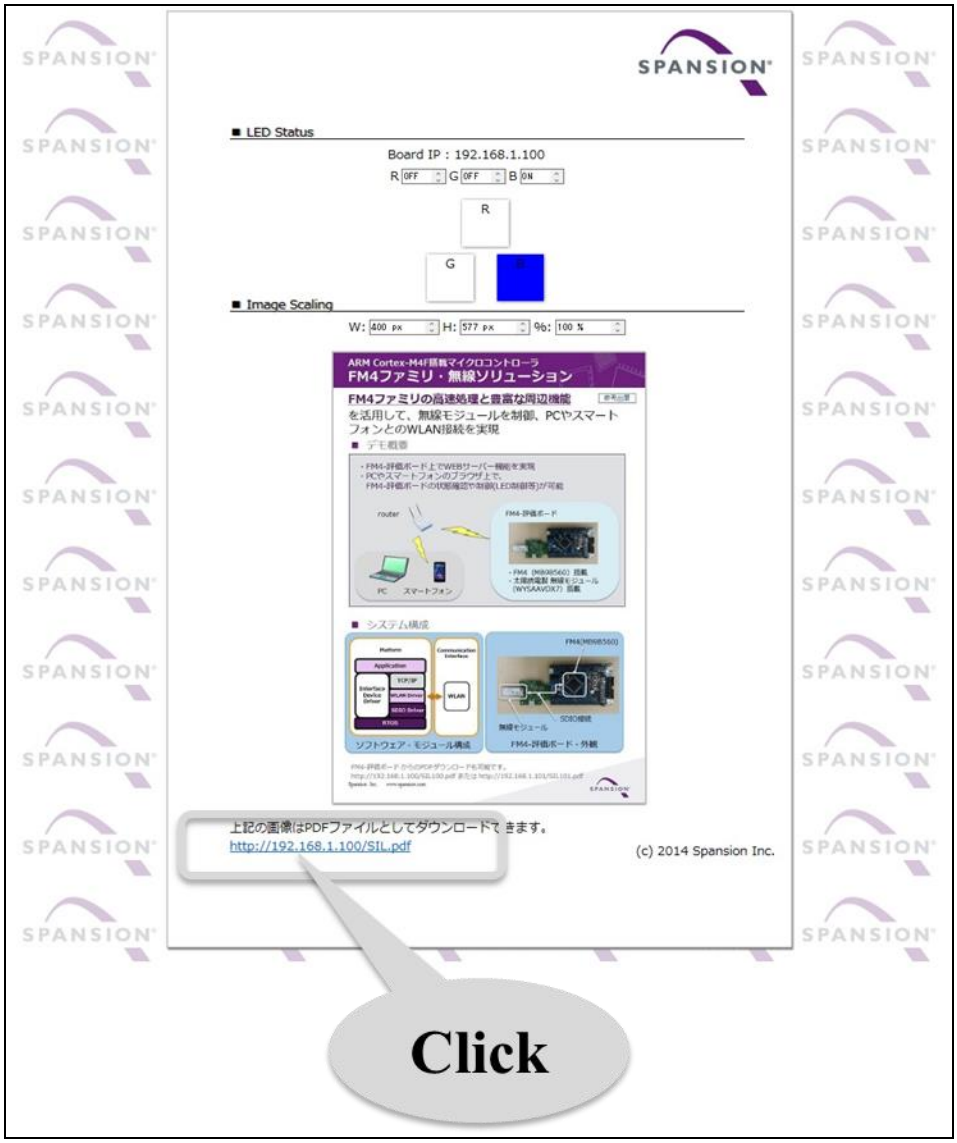
ダウンロードには 2 つの方法があります。

1 Web ブラウザでの URL を直接入力

URL 一覧の PDF ページの URL を、Web ブラウザのアドレス入力フィールドに直接入力してください。

2 デモのメイン画面からリンクをクリック

図 17. PDF ファイルのダウンロード



3 ハードウェア概要

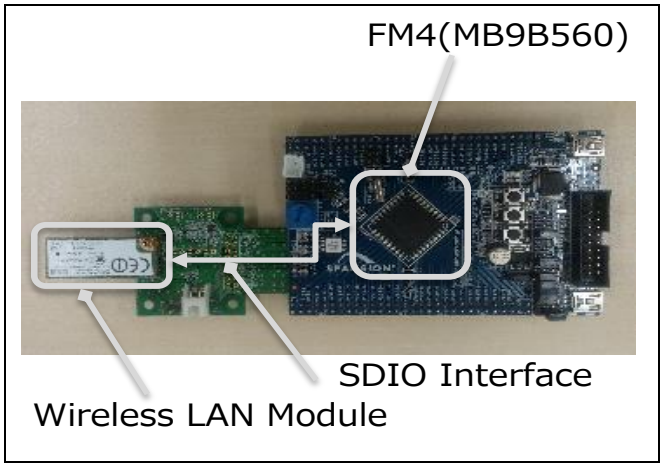
3.1 ハードウェア構成

ハードウェア構成を以下に示します。

FM4 評価ボードと無線 LAN モジュールの構成

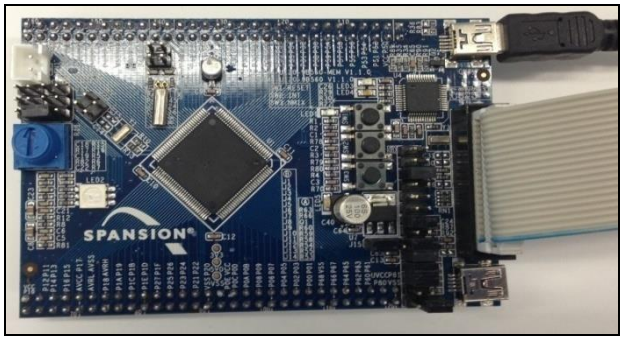
[FM4 評価ボードと無線 LAN モジュールの接続]

図 18. FM4 評価ボードと無線 LAN モジュールの接続



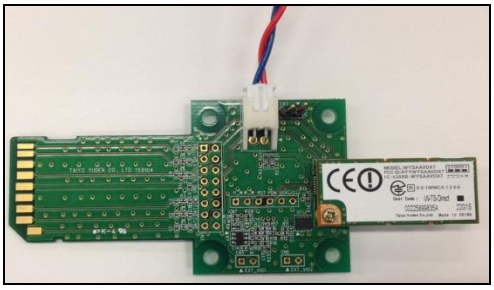
[FM4 評価ボード]

図 19. FM4 評価ボード



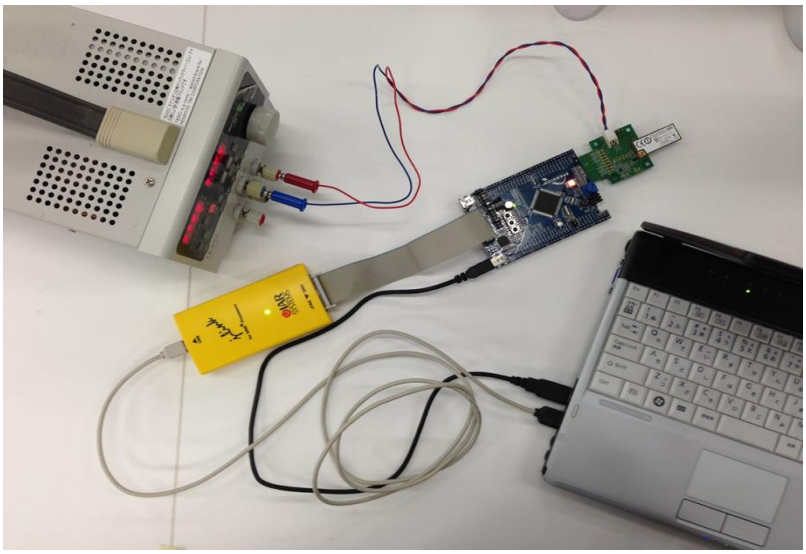
[無線 LAN モジュール]

図 20. 無線 LAN モジュール



開発時のハードウェア全体構成

図 21. 開発時のハードウェア全体構成

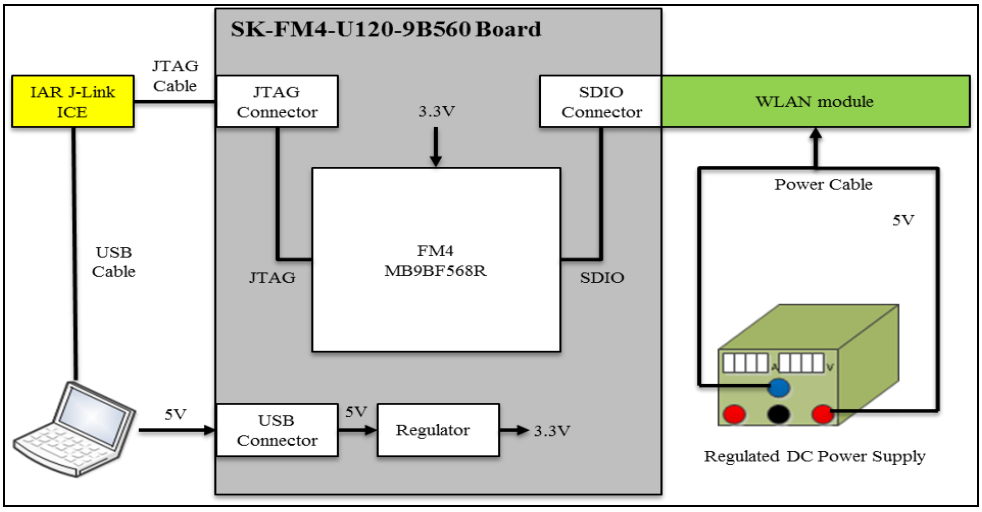


ハードウェアブロック図

FM4 と無線 LAN モジュールを連携して動作させるハードウェアについて説明します。

- WLAN module
  FM4 と無線 LAN モジュールは SDIO インタフェースで通信します。FM4 は SD カードインタフェースの SD Host 機能を使用して SDIO 仕様で動作します。
- IARJ-Link ICE
  FM4 評価ボードと IAR 製 J-Link ICE は 20pin の JTAG インタフェースで接続し、PC と IAR 製 J-Link ICE ICE は USB で接続します。
  FM4 の内蔵 Flash への書込みと、プログラムのデバッグに使用します。
  使用に際しては、IAR Embedded Workbench が必要です。
- Regulated DC Power Supply
  5V の直流電源を出力できるものを接続してください。

図 22. ハードウェアブロック図



3.2 詳細仕様

ハードウェアの詳細仕様について示します。

開発時に用意する機材一覧

表 6. 使用機材一覧

No.	名称	メーカ	型格	内容
1	FM4 評価ボード	CYPRESS	SK-FM4-U120-9B560	FM4 評価ボード本体です。
2	無線 LAN モジュール評価ボード	太陽誘電	WBSAAVDX7	無線 LAN モジュール評価ボードです。 ＜注意事項＞ このモジュールを購入すると赤と青の電源ケーブルが付属されています。
3	安定化電源	-	-	無線 LAN モジュール評価ボードへの電源を供給します。5V の直流電源を出力できるものを使用してください。 ＜注意事項＞ 安定化電源の代替として、専用電源ケーブルにより 5V の電源供給を行うことも可能です。
4	IAR J-Link ICE	IAR	J-Link-ARM	プログラムの書込みやデバッグに使用します。
5	PC	-	-	IAR Embedded Workbench が動作する PC を使用してください。 USB ポートは、最大で 3 つ使用します。 1 つ目：FM4 評価ボードの電源供給。 2 つ目：J-Link との接続。 3 つ目：無線 LAN モジュールの電源供給。
6	USB ケーブル	-	-	[USB TypeA オス]-[USB Mini- B オス]のケーブルを使用してください。 FM4 評価ボードの電源供給用です。

搭載部品

搭載部品について示します。

[FM4 評価ボード]

以下の Web サイトを参照してください。

http://www.cypress.com/products/fm4-mb9bfx6xmnr-series-motor-control-arm-cortex-m4-microcontroller-mcu-family

[無線 LAN モジュール]

以下の Web サイトを参照してください。

https://www.yuden.co.jp

購入した際に、同梱されているドキュメントを参照してください。

使用するインタフェースについて

各ボードで使用しているインタフェースについて示します。

[電源]

図 23. 電源インタフェース図

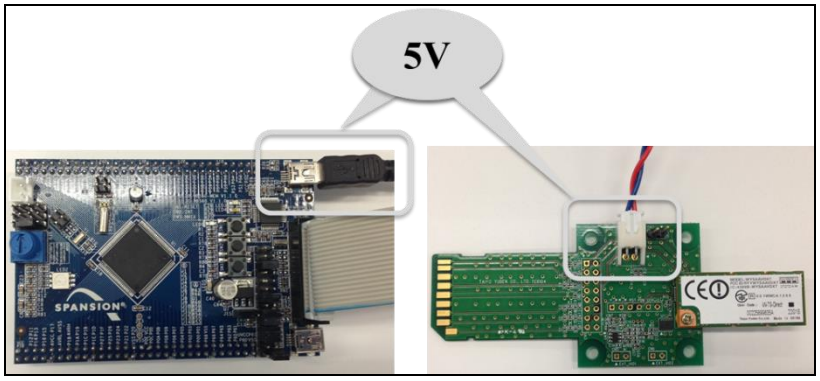
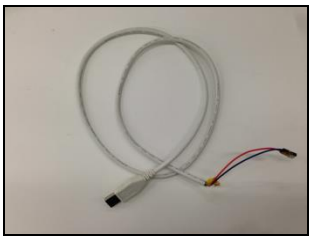


図 24. 専用電源ケーブル図



本ハードウェアでは、FM4 評価ボードと無線 LAN モジュール評価ボードに、それぞれ 5V の電源を供給する必要があります。

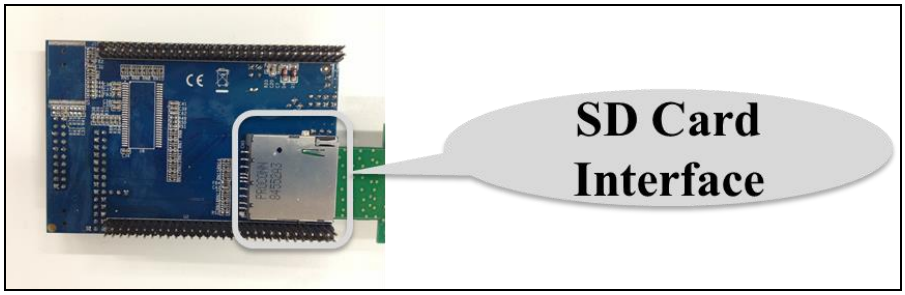
FM4 評価ボードへの電源供給は、PC と USB ケーブルを接続することで供給します。

無線 LAN モジュール評価ボードへの電源供給は、安定化電源と接続することで供給します。

電源は 5V ですので、専用電源ケーブルを作成することでも供給可能です。

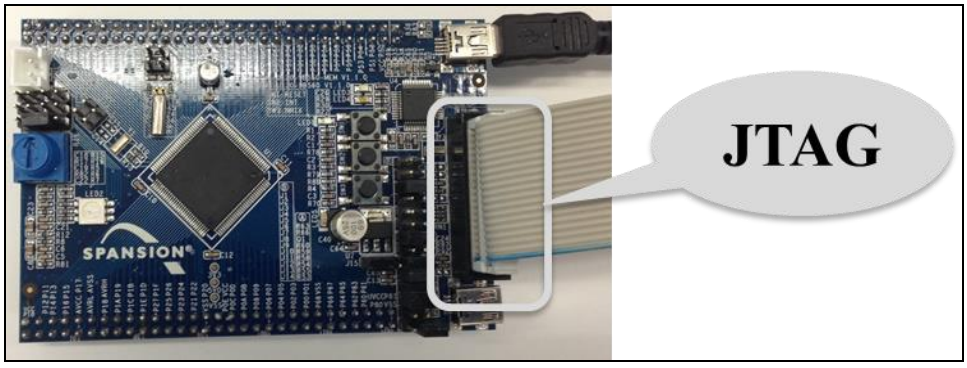
[SD カードインタフェース]

図 25. SD カードインタフェース



SD カードスロットは、プッシュオン・プッシュオフ機構のため、挿入時または抜去時に一度奥まで押し込んでください。

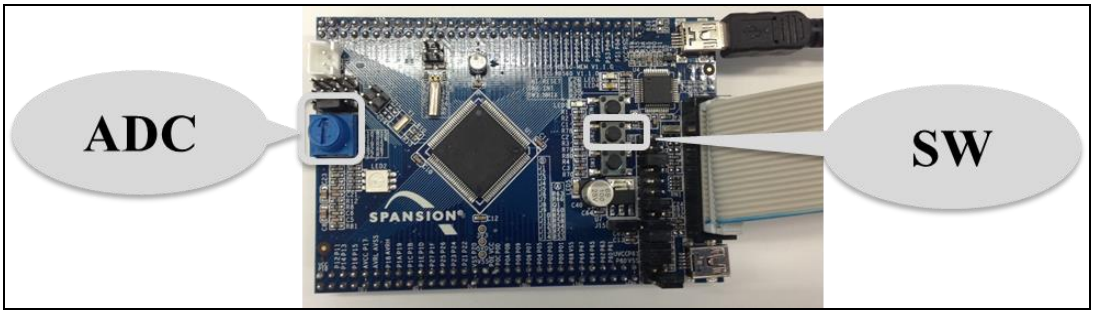
図 26. JTAG インタフェース



IAR 製 J-Link ICE と接続する 20pin の JTAG です。

[ADC/SW インタフェース]

図 27. ADC/SW インタフェース



ADC は、デモの伸縮表示で使用し、ダイヤル式です。

SW は、デモの LED 表示変更で使用し、プッシュ式です。

4 ソフトウェア概要

4.1 ソフトウェア構成

ソフトウェア構成を以下に示します。

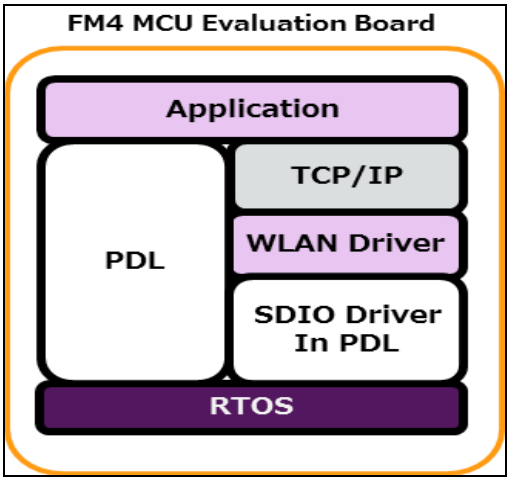
概要

本デモシステムのソフトウェアは、Real Time OS(RTOS)上で動作し、TCP/IP プロトコルスタック、無線 LAN ドライバ、FM4 ファミリ用ペリフェラルドライバをプラットフォームとして、デモアプリケーションを動作させます。

ブロック図

各種ソフトウェアの部品を階層イメージで示します。

図 28. ソフトウェア・モジュール構成



[Application]

Web サーバ機能として、lwIP-contrib 1.4.1 を使用します。

[TCP/IP]

TCP/IP プロトコルスタックとして、lwIP STABLE-1.4.1 を使用します。

[RTOS]

Real Time OS として、µT-REALOS/M4F for EWARM を使用します。

[PDL および SDIO Driver In PDL]

FM4 のペリフェラルデバイスドライバとして、PDL Project for FM4 Family ver1.0 と本デモシステム用に拡張した SDIO Driver を使用します。

[WLAN Driver]

無線 LAN モジュールと SDIO インタフェースを介して通信を行い、無線 LAN モジュールを制御します。

契約によりソースコードの開示ができないため、ライブラリ形式で提供します。

4.1.1 OS リソースについて

図 29. スタートアップとタスクの構成

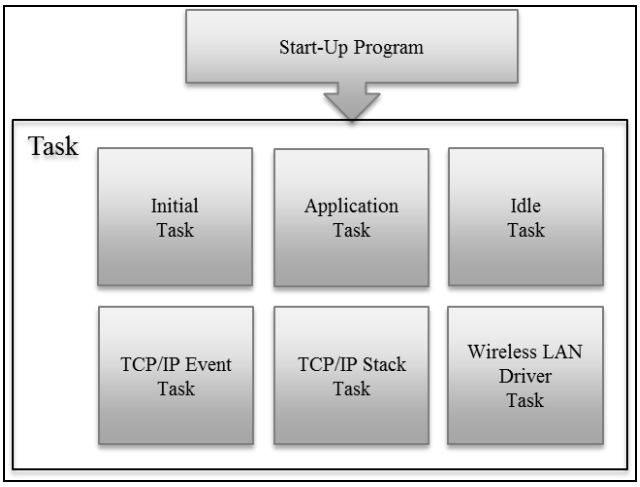


表 7. タスクについて

優先度	タスク名	スタックサイズ(Byte)	処理内容
10	Wireless LANDriver Task	4096(*)	無線 LAN ドライバのメインタスクです。 無線 LAN モジュールに対して送受信の制御を行います。
20	TCP/IP Stack Task	4096(*)	TCP/IP スタックのメインタスクです。 TCP/IP のプロトコル処理を行います。
29	TCP/IP Event Task	4096(*)	TCP/IP スタックを起動した後、無線 LAN ドライバからのイベント通知を待ち受け、対応する TCP/IP スタックの処理を起動します。
30	Application Task	4096(*)	ユーザのメインタスクです。 各種初期化処理を呼び出します。
40	Idle Task	256(*)	RTOS のアイドル処理のためのタスクで、基本的に無処理です。削除しないでください。
98	Initial Task	1024(*)	RTOS の初期起動タスクです。

*:各タスクのスタックサイズについては、最適化を実施していません。

表 8. タスク以外の OS 資源について

タスク名	利用資源
Wireless LAN Driver Task	イベントフラグ x 1、メールボックス x 1
TCP/IP Stack Task	ミューテックス(*) セマフォ(*) メッセージバッファ(*)
TCP/IP Event Task	イベントフラグ x 1
Application Task	利用資源はありません。
Idle Task	利用資源はありません。
Initial Task	利用資源はありません。

*:TCP/IP スタックタスクの利用資源は lwIP から動的に呼び出されます。そのため、動作する機能によって使用する資源数は増減します。本無線 LAN デモシステムでは、コンフィグレーションで設定している最大数で十分ですが、機能を拡張する場合は、コンフィグレーションの最大数を調整してください。

4.1.2 ファイル構成

本アプリケーションのファイル構成は下記のとおりです。

なお、一部変更していないファイルや変更ファイルが無いディレクトリについては省略しています。

```
+---app              : ユーザ追加実装ディレクトリ
  +---pdl_user.h     : FM4 PDL 設定ヘッダファイル
  +---user_main.c    : ユーザメインソースファイル
  +---user_main.h    : ユーザメインヘッダファイル
+---fm4              : FM4-PDL ディレクトリ
  +---common         : FM4-PDL の共通部ディレクトリ
    +---base_types.h : 基本型共通ヘッダファイル
    +---mb9abxxx.h   : ペリフェラル定義ヘッダファイル
    +---mcu.h        : MCU 共通ヘッダファイル
    +---system_mb9abxxx.c : システム設定ソースファイル
    +---system_mb9abxxx.h : システム設定定義ヘッダファイル
  +---library                : PDL ライブラリディレクトリ
    +---driver       : PDL コアドライバディレクトリ
      +---interrupts.c : PDL 割込みハンドラソースファイル
      +---sd         : SD カードインタフェースドライバディレクトリ
        +---sd.c     : SD カードインタフェースドライバソースファイル
        +---sd.h     : SD カードインタフェースドライバヘッダファイル
        +---sd_cfg.h : SD カードインタフェースドライバ構成ヘッダファイル
    +---highlevel    : PDL 上位ドライバディレクトリ
      +---sd         : SD 上位ドライバディレクトリ
        +---sdcard.c : SD メモリカード処理ソースファイル
        +---sdcard.h : SD メモリカード処理ヘッダファイル
        +---sdcmd.c  : SD コマンド共通処理ソースファイル
```

```
      +---sdcmd.h            : SD コマンド共通処理ヘッダファイル
      +---sdiocard.c         : SDIO カード処理ソースファイル
      +---sdiocard.h  : SDIO カード処理ヘッダファイル
+---lwip              : lwIP ソフトウェアのコア部ディレクトリ
+---lwip_contrib      : lwIP ソフトウェアの周辺部ディレクトリ
 +---apps                    : lwIP ソフトウェアのアプリケーションプロトコル部ディレクトリ
  +---httpserver_raw         : lwIP HTTP server ディレクトリ
   +---fsdata.c              : コンテンツ C 言語ソースファイル
   +---httpd_structs.h       : HTTP ヘッダ用文字列定義ヘッダファイル
   +---sample_demo_contetns        : コンテンツ C 言語ソース生成ディレクトリ
+---fs                : デモコンテンツ格納ディレクトリ
+---lwip_usr          : lwIP ソフトウェアのユーザ実装ディレクトリ
 +---ports                   : ユーザ実装ディレクトリ
  +---fm4                   : FM4 実装ディレクトリ
   +---ajax.c               : lwIP の HTTP server AJAX 処理ソースファイル
   +---app_lwip.c           : lwIP のアプリケーション実装部
   +---ethif.c              : lwIP の入出力関連の低レベル実装部
   +---sys_arch.c           : lwIP のアーキテクチャ依存の低レベル実装部
   +---include              : lwIP のインクルードディレクトリ
    +---app                 : lwIP のアプリ実装関連のインクルードディレクトリ
     +---ajax.h             : lwIP の HTTP server AJAX 処理ヘッダファイル
     +---app_lwip.h         : lwIP のアプリケーション実装部ヘッダファイル
     +---ethif.h            : lwIP の入出力関連の低レベル実装部ヘッダファイル
    +---arch                : lwIP のアーキ実装関連のインクルードディレクトリ
     +---cc.h               : lwIP のアーキテクチャ環境定義ファイル
     +---lwipopts.h         : lwIP のオプション定義ファイル
     +---perf.h             : lwIP の perf 関連ヘッダファイル
     +---sys_arch.h         : lwIP のアーキテクチャ依存の低レベル実装部ヘッダファイル
+---proj              : 開発環境プロジェクトディレクトリ
 +---IAR                          : IAR 用(JTAG)プロジェクトディレクトリ
  +---config                      : コンフィグレーションファイルディレクトリ
   +---mb9bf568.icf               : リンカ設定ファイル
   +---reset.mac                  : 起動マクロファイル
    +---flashloader               : フラッシュローダーディレクトリ
     +---FlashLoader.board        : フラッシュローダー設定ファイル指定ファイル
     +---FlashMB9B560.flash              : フラッシュローダー設定ファイル
     +---FlashMB9B560.flash              : フラッシュローダー起動マクロファイル
     +---FlashMB9B560.out         : フラッシュローダー本体
   +---wlan_release.ewd           : プロジェクト設定ファイル
   +---wlan_release.ewp           : プロジェクトファイル
```

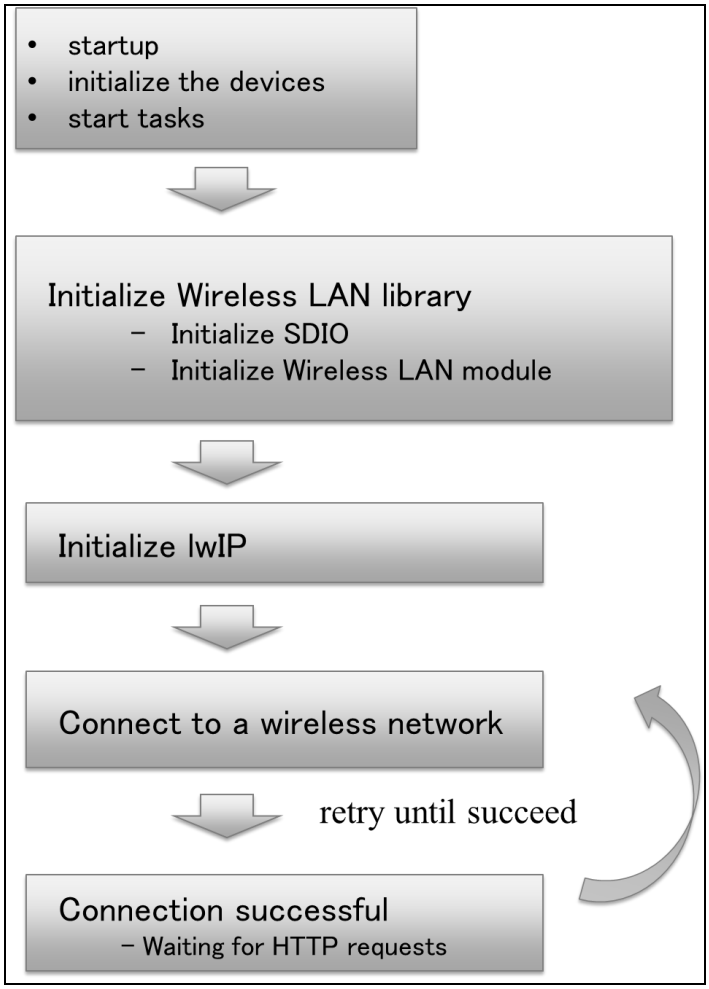
```
  +---wlan_release.eww      : ワークスペースファイル
+---utrealos          : µT-REALOS ディレクトリ
+---utrealos_usr      : µT-REALOS のユーザ実装ディレクトリ
  +---icrt0.asm             : スタートアップファイル
  +---os.c                  : µT-REALOS の低レベル実装ソースファイル
  +---os.h                  : µT-REALOS の低レベル実装ヘッダファイル
  +---startup_mb9xfxxx.s    : 割り込みベクタ定義ファイル
  +---cfg                   : µT-REALOS のコンフィグレーションディレクトリ
    +---config.a            : µT-REALOS のコンフィグレーションファイル
    +---wlancfg.tcf         : µT-REALOS のコンフィグレーション定義ファイル
+---wlanlib           : 無線 LAN ドライバライブラリディレクトリ
  +---wlan.h                : 無線 LAN ドライバライブラリヘッダファイル
  +---wlan_conf.c           : 無線 LAN ドライバコンフィグレーションソースファイル
  +---wlanlib.a             : 無線 LAN ドライバライブラリファイル
```

4.2 詳細仕様

詳細仕様を以下に示します。

動作および概略フロー

図 30. ソフトウェア概略フロー



API 仕様

<µT-REALOS >

µT-REALOS のドキュメントを参照してください。

< lwip Protocol Stack >

lwIP の公式プロジェクトが発行している lwIP に関するドキュメントを参照してください。

lwIP の公式プロジェクトの URL を以下に示します。

http:://savannah.nonqnu.org/projects/lwip/

lwIP は 2 つのパッケージで構成されています。”lwip”は公式プロジェクトで開発された TCP/IP スタックを含み、”contib”には追加コードが含まれています。

< PDL >

PDL のドキュメントを参照してください。

PDL のドキュメントは、以下の URL から Peripheral Driver Library をダウンロードした圧縮ファイル内にあります。

http://www.cypress.com/products/fm4-32-bit-arm-cortex-m4-microcontroller-mcu-families

SDIO カード対応のため、SD ドライバを拡張しています。

拡張した SD ドライバの API については「4.2.1 SDIO カード対応 SD ドライバ」で説明します。

<無線 LAN ドライバ>

無線 LAN ドライバの API については「4.2.2 無線 LAN ドライバ」で説明します。

4.2.1 SDIO カード対応 SD ドライバ

構造体定義

項目	説明
概要	コールバック設定情報
書式	typedef struct stc_sd_config_cb { func_ptr_sd_arg32_t pfnTxCallback; func_ptr_sd_arg32_t pfnRxCallback; func_ptr_sd_arg32_t pfnWakeupCallback; func_ptr_sd_arg32_t pfnErrorCallback; func_ptr_sd_arg32_t pfnCardIntCallback; func_ptr_sd_arg32_t pfnErrorResponseCallback; } stc_sd_config_cb_t;
説明	pfnTxCallback ：Tx 発生時の Callback 関数 (未使用) pfnRxCallback ：Rx 発生時の Callback 関数 (未使用) pfnWakeupCallback ：Wakeup 発生時の Callback 関数 (未使用) pfnErrorCallback ：Error 発生時の Callback 関数 (未使用) pfnCardIntCallback ：Card Interrupt 発生時の Callback 関数 pfnErrorResponseCallback ：コマンドレスポンスのエラー発生時の Callback 関数
備考	pfnCardIntCallback, pfnErrorResponseCallback 以外への設定は無効です

項目	説明
概要	SDIO カード情報
書式	typedef struct stc_sdiocard_info { en_sdio_init_type_t init_type; en_sdio_init_io_t io_init; en_sdio_init_mem_t mem_init; uint8_t flg; uint32_t OCR; uint16_t RCA; en_sdio_card_type_t card_type; } stc_sdiocard_info_t;
説明	init_type ：Sdio Initialize Type **PowerOn** Power On **ReInitMemory** Re-Init Memory **ReInitIO** Re-Init IO io_init ：Sdio IO Initialize Select **NotUseIO** Not Use IO(IO Initialize Disable) **UseIO** Use IO(IO Initialize Enable) mem_init ：Sdio Memory Initialize Select **NotUseMem** Not Use Memory(Memory Initialize Disable) **UseMem** Use Memory(Memory Initialize Enable) flg ：Initalize Status Flag OCR ：Operation Condition Register RCA ：Relative Card Address card_type ：Sdio Card Type **SDSC_MemOnly** SDSC Memory Only Card **SDHCXC_MemOnly** SDHC or SDXC Memory Only Card **SD_IOOnly** SD IO Only Card **SDSC_Combo** SDSC Combo Card **SDHCXC_Combo** SDHC or SDXC Combo Card
備考	

項目	説明
概要	DMA 転送情報
書式	typedef struct stc_sdh_dma_info { en_sd_card_type_t sdc_type; en_sdh_dma_type_t dma_type; uint32_t data_addr; uint32_t data_addr_h; en_sd_boundary_t sdma_bound; } stc_sdh_dma_info_t;
説明	sdc_type ：SD Card Type **SDH_SD_CARD** SD Memory Card **SDH_SDIO_CARD** SDIO Card dma_type ：SD Host DMA Type **SDH_SDMA** SDMA **SDH_ADMA** ADMA2 data_addr ： DMA データアドレス SDMA の場合：Buffer Address ADMA2 の場合：ディスクリプタの下位 32bit アドレス data_addr_h ： DMA データアドレス(64bit システムのための領域) SDMA の場合：不要 ADMA2 の場合：ディスクリプタの上位 32bit アドレス sdma_bound ： SDMA のためのバウンダリ指定 **BOUND_4K** **BOUND_8K** **BOUND_16K** **BOUND_32K** **BOUND_64K** **BOUND_128K** **BOUND_256K** **BOUND_512K**
備考	ADMA2 のディスクリプタについて ・1 つのディスクリプタ領域は、以下の構成です。 属性情報領域(attr)[16bit]＋データ長領域(len)[16bit]＋データ格納アドレス(addr)[32bit] ・attr はドライバ内で自動的に設定しますので、呼び出し元では 0 に設定してください。 ・ディスクリプタ領域は、link 属性でない限り連続した領域にしてください。 ・link 属性のディスクリプタを設定する場合は、len を 0 に設定してください。 このとき、addr で示される領域はディスクリプタ領域です。 ・ディスクリプタには、必ず終端判別用ディスクリプタを付加してください。 終端判別用ディスクリプタは、len を 0、addr を 0(NULL)に設定してください。 ディスクリプタ例

attr	len	addr
0	64	0x40000000
0	64	0x40000040
0	256	0x40000080
0	128	0x40000180
0	0	0

API 一覧

[SD カードインタフェースドライバ API]

API 仕様	説明
概要	SD スピードモードの High-Speed への切替
書式	void Sd_HighspeedChange (void)
入力	なし
返却値	なし
備考	初期化完了後に、カード側が High-Speed をサポートしている場合に、本 API を呼び出してください。

API 仕様	説明
概要	割込みコールバック関数の設定
書式	en_result_t Sd_SetIntCallback (stc_sd_config_cb_t* pstcSdCbCfg)
入力	pstcSdCbCfg ：コールバック設定情報のポインタ (コールバック設定情報の詳細は、構造体の説明を参照してください)
返却値	エラーコード Ok　正常終了 (本 API では常に Ok が返却されます)
備考	Sd_HostInit()を呼び出し後に、 割込みコールバックの設定が必要な場合に、本 API を呼び出してください。 pfnCardIntCallback, pfnErrorResponseCallback 以外への設定は無効です。

API 仕様	説明
概要	CardInterrupt の有効化
書式	en_result_t Sd_EnableCardInt (volatile stc_sd_t *pstcSd)
入力	pstcSd ：SD カードインタフェースレジスタ情報のポインタ
返却値	エラーコード Ok　正常終了 (本 API では常に Ok が返却されます)
備考	Sd_HostInit()を呼び出し後に、CardInterrupt を有効化する場合に、本 API を呼び出してください。 Sd_SetIntCallback で pfnCardIntCallback を設定している場合には、割込み処理終了後、本 API で有効化することで、次の CardInterrupt を受けることが可能になります。

API 仕様	説明
概要	CardInterrupt の無効化
書式	en_result_t Sd_DisableCardInt (volatile stc_sd_t *pstcSd)
入力	pstcSd ：SD カードインタフェースレジスタ情報のポインタ
返却値	エラーコード Ok　正常終了 (本 API では常に Ok が返却されます)
備考	Sd_HostInit()を呼び出し後に、CardInterrupt を無効化する場合に、本 API を呼び出してください。

[SDIO 上位レベルドライバ API]

API 仕様	説明
概要	SDIO カードの挿抜状態取得
書式	boolean_t Sdiocard_Detect (void)
入力	なし
返却値	挿抜状態 TRUE　挿入状態 FALSE　抜去状態
備考	Sd_HostInit()を呼び出し後に、本 API を呼び出してください。

API 仕様	説明
概要	SDIO の初期化
書式	int32_t Sdiocard_Init (stc_sdiocard_info_t *pstcSdiocardInfo)
入力	pstcSdiocardInfo ：SDIO カード情報へのポインタ (SDIO カード情報の詳細は、構造体の説明を参照してください)
返却値	エラーコード E_SDIO_OK　正常終了 E_SDIO_NO_CARD　エラー終了(SDIO カードが挿さっていません) E_SDIO_INIT　エラー終了 (SDIO カードの初期化に失敗しました)
備考	Sd_HostInit()を呼び出し後に、本 API を呼び出してください。

API 仕様	説明
概要	CMD52 の送受信
書式	int32_t Sdiocard_Cmd52 (stc_sdiocard_info_t *pstcSdiocardInfo, const uint32_t* pu32arg, uint32_t* pu32res)
入力	pstcSdiocardInfo : SDIO カード情報へのポインタ (SDIO カード情報の詳細は、構造体の説明を参照してください) pu32arg : SDIO CMD52 argument へのポインタ pu32res : SDIO CMD52 response 格納領域へのポインタ
返却値	エラーコード E_SDIO_OK 正常終了 E_SDIO_NO_CARD エラー終了 (SDIO カードが挿さっていません) E_SDIO_PARAMETER エラー終了 (パラメータが不正です) E_SDIO_WRITE エラー終了 (ライト動作に失敗しました) E_SDIO_READ エラー終了 (リード動作に失敗しました)
備考	Sdiocard_Init() を呼び出し後に、本 API を呼び出してください。 本 API は、SD コマンド共通処理関数の Sdcmd_SendCmd を使用して CMD52 の送受信処理を行います。

API 仕様	説明
概要	CMD53 の送受信(PIO 転送)
書式	int32_t Sdiocard_Cmd52 (stc_sdiocard_info_t *pstcSdiocardInfo, const uint32_t* pu32arg, uint32_t* pu32res, uint8_t* pu8buf)
入力	pstcSdiocardInfo : SDIO カード情報へのポインタ (SDIO カード情報の詳細は、構造体の説明を参照してください) pu32arg : SDIO CMD52 argument へのポインタ pu32res : SDIO CMD52 response 格納領域へのポインタ pu8buf : 送信データへのポイント、または受信データ格納領域へのポインタ
返却値	エラーコード E_SDIO_OK 正常終了 E_SDIO_NO_CARD エラー終了 (SDIO カードが挿さっていません) E_SDIO_PARAMETER エラー終了 (パラメータが不正です) E_SDIO_WRITE エラー終了 (ライト動作に失敗しました) E_SDIO_READ エラー終了 (リード動作に失敗しました)
備考	Sdiocard_Init() を呼び出し後に、本 API を呼び出してください。 本 API は、SD コマンド共通処理関数の Sdcmd_SendCmd/Sdcmd_TxData/Sdcmd_RxData を使用して CMD53 の PIO 転送による送受信を行います。

API 仕様	説明
概要	CMD53 の送受信 (DMA 転送)
書式	int32_t Sdiocard_Cmd53_dma (stc_sdiocard_info_t *pstcSdiocardInfo, const uint32_t* pu32arg, uint32_t* pu32res, stc_sdh_dma_info_t *pstcDmaInf)
入力	pstcSdiocardInfo : SDIO カード情報へのポインタ (SDIO カード情報の詳細は、構造体の説明を参照してください) pu32arg : SDIO CMD52 argument へのポインタ pu32res : SDIO CMD52 response 格納領域へのポインタ pstcDmaInf : DMA 転送情報へのポインタ (DMA 転送情報の詳細は、構造体の説明を参照してください)
返却値	エラーコード E_SDIO_OK 正常終了 E_SDIO_NO_CARD エラー終了 (SDIO カードが挿さっていません) E_SDIO_PARAMETER エラー終了 (パラメータが不正です) E_SDIO_WRITE エラー終了 (ライト動作に失敗しました) E_SDIO_READ エラー終了 (リード動作に失敗しました)
備考	Sdiocard_Init ()を呼び出し後に、本 API を呼び出してください。 本 API は、SD コマンド共通処理関数の Sdcmd_SendCmd_dma を使用して CMD53 の DMA 転送による送受信を行います。 本 API を使用して、ADMA2 による DMA 転送を行う場合、DMA 転送情報領域および DMA データ領域に加えて、呼び出し元でディスクリプタ領域を確保する必要があります。

4.2.2 無線 LAN ドライバ

構造体定義

項目	説明
概要	OS リソース情報
書式	typedef struct wlan_os_resource { PRI itskpri; void* stk; int32_t stksz; } WLAN_OS_RESOURCE;
説明	itskpri ：無線 LAN ドライバタスクの優先度 stk ：無線 LAN ドライバタスクのスタックアドレス stksz ：無線 LAN ドライバタスクのスタックサイズ
備考	コンフィグレーションファイル wlan_conf.c において、 変数 wlan_os_resource_init として定義してください。

項目	説明
概要	scatter/gather 型
書式	typedef struct wlan_sg { uint16_t __reserved_1; uint16_t len; void* addr; } WLAN_SG;
説明	len ：長さ addr ：アドレス
備考	送信 API で使用します。

項目	説明
概要	コールバック定義
書式	typedef struct wlan_callback { void (*Recv)(uint32_t epbuf_len); void (*Event)(uint32_t event, void* data); } WLAN_CALLBACK;
説明	Recv ：受信コールバック関数 Event ：イベントコールバック関数
備考	コンフィグレーションファイル wlan_conf.c において、 変数 wlan_callbackfunc として定義してください。

項目	説明
概要	接続コンフィグレーション
書式	#define WLAN_SSID_LEN_MAX 32 #define WLAN_KEYLEN_MAX 32 typedef struct wlan_config { uint8_t ssid[WLAN_SSID_LEN_MAX]; uint32_t ssid_len; int comm_mode; int auth_mode; uint8_t key[WLAN_KEYLEN_MAX]; uint32_t key_len; int channel; int uap_wpa_cipher; } WLAN_CONFIG;
説明	ssid ：SSID ssid_len ：SSID の長さ (0 指定で自動判定) comm_mode ：接続方式 **WLAN_MODE_ADHOC** アドホックモード **WLAN_MODE_INFRASTRUCTURE** インフラストラクチャーモード **WLAN_MODE_UAP** Micro AP モード auth_mode ：認証モード **WLAN_AUTH_MODE_OPEN** 認証なし **WLAN_AUTH_MODE_WEP** WEP (40bit / 104bit) **WLAN_AUTH_MODE_WPA_PSK** WPA/WPA2-PSK key ：鍵 key_len ：鍵の長さ（0 指定で自動判定） **WEP の場合** 5 文字 (40bit)、または 13 文字 (104bit) の鍵を指定 **WPA の場合** 任意の長さ (最大 32 文字) のパスフレーズを指定 channel ：チャンネル（0 指定で自動検出） 個別指定の場合は、1 - 11ch の範囲の値を設定してください。 uap_wpa_cipher *(Micro AP モードのみ。Micro AP モード以外では無視されます)* ：WPA 暗号方式 **WLAN_CIPHER_MIXED** AES-CCMP/TKIP mixed モード **WLAN_CIPHER_AES_CCMP** AES-CCMP
備考	コンフィグレーションファイル wlan_conf.c において、変数 wlan_conf として定義してください。 設定内容は、接続先の相手または接続元の情報を設定します。 – uap_wpa_cipher の AES-CCMP/TKIP mixed モードについて 本モードでは、Group Cipher として TKIP が使われます。つまり、AES- CCMP で接続する場合でも、TKIP のサポートが必須です。 対向機器にあわせて、以下のように設定してください。 ・TKIP のみサポートしている機器→**WLAN_CIPHER_MIXED** ・AES-CCMP のみサポートしている機器→**WLAN_CIPHER_AES_CCMP** なお、Micro AP モード以外では、WPA 暗号方式は自動的に判別できます。

項目	説明
概要	スキャン情報

書式	#define WLAN_SSID_LEN_MAX 32
	#define WLAN_BSSID_LEN 6
	
	typedef struct wlan_scan {
	uint8_t ssid[WLAN_SSID_LEN_MAX];
	uint32_t ssid_len;
	uint8_t bssid[WLAN_BSSID_LEN];
	int comm_mode;
	int auth_mode;
	int channel;
	uint8_t rssi;
	} WLAN_SCAN;
	
	#define WLAN_SCANRESULT_MAX 16
	
	typedef struct wlan_scanresult {
	int bss_num;
	WLAN_SCAN bss_info[WLAN_SCANRESULT_MAX];
	} WLAN_SCANRESULT;

説明	ssid ：SSID ssid_len ：SSID の長さ bssid ：BSSID comm_mode ：接続方式 **WLAN_MODE_ADHOC** アドホックモード **WLAN_MODE_INFRASTRUCTURE** インフラストラクチャーモード **WLAN_MODE_UNKNOWN** 不明 auth_mode ：認証モード **WLAN_AUTH_MODE_OPEN** 認証なし **WLAN_AUTH_MODE_WEP** WEP (40bit / 104bit) **WLAN_AUTH_MODE_WPA_PSK** WPA/WPA2-PSK channel ：チャンネル rssi ：RSSI bss_num ：検索結果の数 検索結果の最大数は 16 個です。それ以上の結果は打ち切られます。 bss_info ：検索結果の配列
備考	

API 一覧

API 仕様	説明
概要	無線 LAN ドライバの初期化
書式	int wlan_start (void)
入力	なし
返却値	エラーコード DRIVER_OK　　正常終了 DRIVER_ERR　　エラー終了
備考	無線 LAN ドライバの API を利用する前に、かならず最初に呼び出してください。 コンフィグレーションファイル wlan_conf.c の変数 wlan_os_resource_init の情報を利用して、OS 資源の生成を行います。

API 仕様	説明
概要	無線 LAN の接続開始
書式	int wlan_connect (void)
入力	なし
返却値	エラーコード DRIVER_OK　　正常終了 DRIVER_ERR　　エラー終了
備考	コンフィグレーションファイル wlan_conf.c の変数 wlan_conf の情報を利用して接続処理を行います。 接続処理が完了する前に API から復帰します。 実際に接続が確立したら、Event コールバック (WLAN_EVENT_CONNECT) により通知されます。 なお、接続が確立する前に対向から切断された場合など、WLAN_EVENT_CONNECT が通知されずに、WLAN_EVENT_DISCONNECT が通知されることがあります。

API 仕様	説明
概要	無線 LAN の切断処理
書式	int wlan_disconnect (void)
入力	なし
返却値	エラーコード DRIVER_OK　　正常終了 DRIVER_ERR　　エラー終了
備考	実際に切断されたら、Event コールバック (WLAN_EVENT_DISCONNECT) により通知されます。

API 仕様	説明
概要	アクセスポイント／アドホックネットワークの検索
書式	int wlan_scan (WLAN_SCANRESULT* result)
入力	result ：検索結果の格納先ポインタ
返却値	エラーコード DRIVER_OK　正常終了 DRIVER_ERR　エラー終了
備考	情報を取得したい場合に呼び出してください。 現在立ち上げられている接続先の相手を検索します。検索には時間 (1.5 秒程度) を要します。なお、16 台を超えた機器の情報は破棄されます。 本 API は Active scan を行います。scan するチャンネルの範囲は 1 - 11ch です。

API 仕様	説明
概要	通信品質の取得
書式	int wlan_get_comm_quality (int16_t* rssi, int16_t* noise_floor)
入力	rssi ：RSSI の格納先ポインタ noise_floor ：Noise Floor の格納先ポインタ
返却値	エラーコード DRIVER_OK　正常終了 DRIVER_ERR　エラー終了
備考	情報を取得したい場合に呼び出してください。 本 API の情報は接続時のみ有効です。接続していない場合、取得値は不定値です。

API 仕様	説明
概要	MAC アドレスの取得
書式	int wlan_get_macaddr (uint8_t* macaddr)
入力	macaddr ：MAC アドレスの格納先ポインタ 格納先には 6 byte の領域を確保してください。
返却値	エラーコード DRIVER_OK　正常終了 DRIVER_ERR　エラー終了
備考	TCP/IP スタックの低レベル実装から呼び出されます。

API 仕様	説明
概要	マルチキャストアドレスフィルタの設定
書式	int wlan_set_multicastaddr (uint8_t* maclist, uint32_t maclist_num)
入力	maclist ：MAC アドレスの配列へのポインタ 格納先には (6 byte * maclist_num) の領域を確保してください。 maclist_num ：maclist に含まれる MAC アドレスの数 最大数は 32 個です。
返却値	エラーコード DRIVER_OK　正常終了 DRIVER_ERR　エラー終了
備考	TCP/IP スタックの低レベル実装から呼び出されます。 登録されている MAC アドレスについては、上位で管理してください。 本 API が呼び出されると、すでに設定されている MAC アドレスの設定は全て破棄されます (上書きされます)。 なお、すべての MAC アドレスを削除したい場合は、maclist_num に 0 を指定してください。

API 仕様	説明
概要	送信関数
書式	int wlan_send_pkt (WLAN_SG* sg_buf, uint32_t total_len)
入力	sg_buf ：送信データ (scatter/gather 構造) sg_buf[0]　無線 LAN ドライバで予約 sg_buf[1]　1 つ目のデータ (物理アドレス/長さ) sg_buf[2]　2 つ目のデータ (物理アドレス/長さ) ・・・ sg_buf[N]　終端 (0 クリア必須) total_len ：sg_buf に含まれるデータの合計長 (byte)
返却値	エラーコード DRIVER_OK　正常終了 DRIVER_ERR　エラー終了
備考	TCP/IP スタックの低レベル実装から呼び出されます。 無線 LAN モジュールにデータを送信した時点で、API から復帰します。 無線 LAN モジュールにデータを送信できない場合は、データを送信できるまで API はブロッキングします。

API 仕様	説明
概要	受信関数
書式	int wlan_recv_pkt (void* epbuf, uint32_t* start_ptr_offset, uint32_t* pktlen)
入力	epbuf ：受信データの格納先 start_ptr_offset ： ep_buf の先頭から MAC フレームまでのオフセット (byte) を格納する領域 pktlen ： ep_buf に格納された MAC フレームの長さ (start_ptr_offset は含まない) を 格納する領域
返却値	エラーコード DRIVER_OK 正常終了 DRIVER_ERR エラー終了
備考	無線 LAN ドライバの受信コールバック内の専用関数です。それ以外の場所からは呼び出さないでください。 引数 epbuf には、受信コールバック関数の引数 epbuf_len で指定された長さの領域を TCP/IP スタックで確保して渡してください。 ep_buf の先頭から start_prt_offset までの領域には、無線 LAN ヘッダが格納されています。適宜、TCP/IP スタックで削除してください。

コールバック関数一覧

API 仕様	説明
概要	受信コールバック関数
書式	void (*Recv)(uint32_t epbuf_len)
入力	epbuf_len ：必要な受信バッファの長さ (byte)
返却値	なし
備考	コンフィグレーションファイル wlan_conf.c の変数 wlan_callbackfunc の情報を利用して、コールバック通知を行います。 本コールバック関数内では、wlan_recv_pkt() を除き、無線 LAN ドライバの API の呼び出しはできません。 本コールバック関数は無線 LAN ドライバのタスク上から呼ばれます。ユーザタスクへの通知を行った後はできるだけ速やかに復帰してください。 コールバック関数内で wlan_recv_pkt() を呼び出さなかった場合、本コールバック関数で通知した受信データは破棄されます。

API 仕様	説明
概要	イベントコールバック関数
書式	void (*Event) (uint32_t event, void* data)
入力	event ：イベントコード **WLAN_EVENT_CONNECT** 接続通知 **WLAN_EVENT_DISCONNECT** 切断通知 **WLAN_EVENT_MIC_ERROR** MIC Failure を検出 data ：イベントの補足情報（現在は未使用）
返却値	なし
備考	コンフィグレーションファイル wlan_conf.c の変数 wlan_callbackfunc の情報を利用して、コールバック通知を行います。 本コールバック関数内では、無線 LAN ドライバの API の呼び出しはできません。 本コールバック関数は無線 LAN ドライバのタスク上から呼ばれます。ユーザタスクへの通知を行った後はできるだけ速やかに復帰してください。 WLAN_EVENT_CONNECT と WLAN_EVENT_DISCONNECT の通知順序は、一対一に対応するとは限りません。接続が確立する前に対向から切断された場合など、WLAN_EVENT_CONNECT が通知されずに WLAN_EVENT_DISCONNECT が通知されることがあります。

4.3 デモソフトウェア

デモソフトウェアについて、以下に説明します。

4.3.1 コンフィグレーション設定について

本デモソフトウェアに関連して、修正したコンフィグレーションの項目を、以下に示します。

表 9. lwIP のコンフィグレーション

No.	コンフィグ名	設定値(*1)	備考
1	MEM_SIZE	16000	lwIP のヒープサイズ
2	MEMP_NUM_TCP_PCB	10	TCP 接続の最大数 (確立したもの)
3	MEMP_NUM_TCP_PCB_LISTEN	5	TCP 接続の同時に Listen できる最大数
4	MEMP_NUM_TCP_SEG	20	TCP セグメントの同時キューイング数
5	MEMP_NUM_IGMP_GROUP	32	IGMP マルチキャストグループの最大数
6	MEMP_NUM_TCPIP_MSG_INPKT	16	TCP/IP 受信メッセージの最大数
7	MEMP_NUM_ARP_QUEUE	5	ARP request の応答待ちキューイング数
8	PBUF_POOL_SIZE	24	固定長メモリプールの最大数
9	LWIP_MULTICAST_PING	1	マルチキャスト PING に応答する
10	LWIP_IGMP	1	IGMP モジュールの有効化
11	TCP_WND	12 (MSS)	TCP 受信ウィンドウバッファサイズ
12	TCP_SND_BUF	10 (MSS)	TCP 送信バッファサイズ
13	PBUF_POOL_BUFSIZE	1514 + 128	固定長メモリプールのブロックサイズ (無線 LAN 処理用のヘッダサイズも含む)
14	TCPIP_THREAD_STACKSIZE	(*2)	TCP/IP タスクのスタックサイズ
15	TCPIP_THREAD_PRIO	(*2)	TCP/IP タスクの優先度
16	TCPIP_MBOX_SIZE	32	TCP/IP タスクのメッセージバッファ段数

*1: 各設定値については最適化を実施していません。

*2: 「表 7. タスクについて」を参照してください。

表 10. µT-REALOS のコンフィグレーション

No.	コンフィグ名	設定値(*1)	備考
1	_KERNEL_MAX_TSK	10	タスクの最大数
2	_KERNEL_MAX_TSKPRI	100	タスク優先度の最大値
3	_KERNEL_MAX_SEM	10	セマフォの最大数
4	_KERNEL_MAX_FLG	10	イベントフラグの最大数
5	_KERNEL_MAX_MBX	10	メールボックスの最大数
6	_KERNEL_MAX_MTX	10	ミューテックスの最大数
7	_KERNEL_MAX_MBF	10	メッセージバッファの最大数
8	_KERNEL_INIT_TSKPRI	98	初期タスクの優先度
9	_KERNEL_INIT_TSKSTKSZ	0x400	初期タスクのスタックサイズ
10	_KERNEL_SYS_STKSIZE	0x400	システムスタックサイズ
11	_KERNEL_USE_IMALLOC	1	ヒープ領域の使用(メッセージバッファで必要)
12	_KERNEL_REALMEMSZ	0x1000	ヒープ領域のサイズ

*1: 各設定値については最適化を実施していません。

表 11. 無線 LAN ドライバのコンフィグレーション

No.	変数名	コンフィグ名	設定値	備考
1	wlan_conf	ssid	“FM4-WLAN_demo”	SSID
2		ssid_len	0	SSID の長さ (0 の設定は、NULL 終端により自動で計算します)
3		comm_mode	WLAN_MODE_INFRASTRUCTURE	接続方式
4		auth_mode	WLAN_AUTH_MODE_OPEN	認証モード
5		key	"password12345"	鍵
6		key_len	0	鍵長 (0 の設定は、NULL 終端により自動で計算します)
7		channel	0	チャンネル (0 の設定は、接続先の利用チャンネルに合わせて自動で識別します)
8		uap_wpa_cipher	WLAN_CIPHER_MIXED	Micro AP モード時の暗号方式
9	wlan_os_resource_init	itskpri	(*1)	無線 LAN ドライバタスクの優先度
10		stk	stk_wlan_tsk	無線 LAN ドライバタスクのスタックアドレス
11		stksz	(*1)	無線 LAN ドライバタスクのスタックサイズ
12	wlan_callback	Recv	callback_Recv	受信コールバック関数
13		Event	callback_Event	イベントコールバック関数

*1: 「表 7. タスクについて」を参照してください。

4.3.2 Web サーバについて

Web サーバの概要と構造

本デモソフトウェアの Web サーバは、lwip の Web サーバを使用しています。

Web サーバの詳細については、lwip のドキュメントを参照してください。

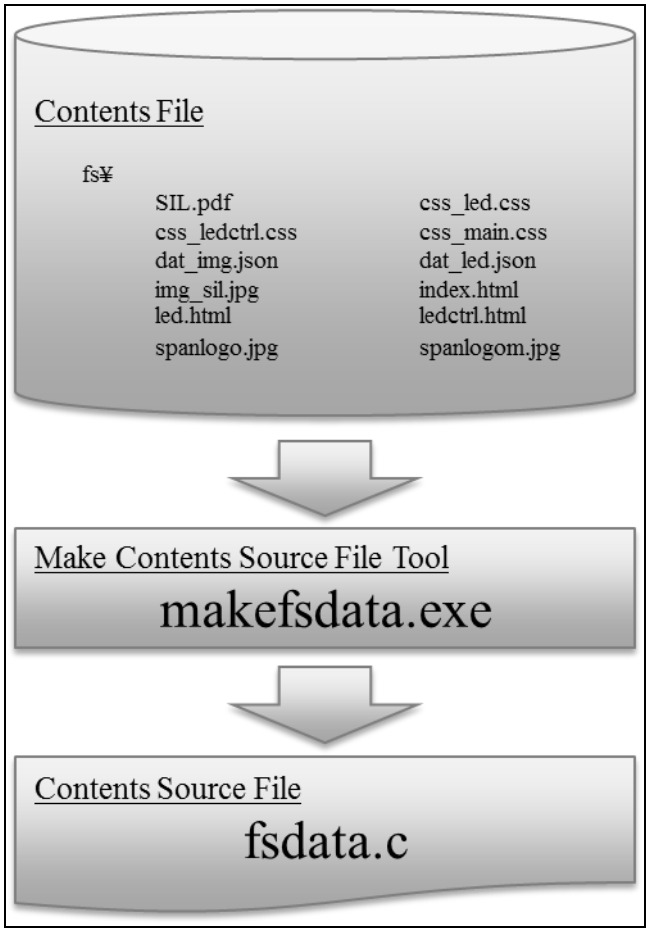
コンテンツの変更方法について

lwip の Web サーバが利用するコンテンツは、lwip より提供されるツール (makefsdata.exe) を使用することで作成できます。

makefsdata.exe が格納された同一ディレクトリに fs ディレクトリを作成、その fs ディレクトリの下に HTML ファイルなどのコンテンツを格納して、makefsdata.exe を実行します。

実行が完了すると、fsdata.c が作成されますので、そのファイルをデモソフトウェアのビルド時に、一緒にビルドすることで、コンテンツをデモソフトウェアに取り込めます。

図 31. コンテンツファイル(fsdata.c)作成動作



CGI について

cgi 処理は ajax.c 内で C 言語の関数として実装します。

cgi 処理を行う関数は以下の手順で登録してください。

<CGI 処理を行うファイル、関数の設定>

1 CGI ファイル名、関数の設定

ajax.c で定義されている変数 CGIHandlers に CGI ファイル名と対応する処理関数のポインタを設定してください。

2 Web サーバへの登録

Ajax_Init()関数内で 1. で設定したファイル名と対応関数が登録されます。

複数の CGI ファイルを登録する場合は、http_set_sgi_handlers() に与える引数を登録する CGI ファイル数に合わせて変更してください。

<CGI 処理を行う関数の実装>

設定した CGI ファイルへのアクセス時に、設定した関数が呼び出されます。

引数には与えられたパラメータが引き渡されるので、必要な処理を実装してください。

関数の戻り値には、CGI 処理後に表示するコンテンツ (HTML ファイル) のファイル名を返しますので、事前に必要なコンテンツを用意してください。

5 その他

注意事項

- ボードは各部品が露出した状態です。金属などの電気を通す物が触れると、ショートする恐れがあるため注意してください。
- 手などを接近させる際にボードに触れると感電する恐れがあるため注意してください。
- 人体や物体が静電気を帯びた状態でボードに接近した場合、ボードへ放電される可能性があります。ボード上に搭載されている各部品は、静電気に弱い部品も含まれているため注意してください。

無線 LAN デモシステムで利用する ROM/RAM サイズ

ROM サイズ	RAM サイズ	備考
772 Kbyte(*)	103 Kbyte(*)	ROM サイズの内訳 CODE : 517 Kbyte DATA : 255 Kbyte

*: 参考値です。

転送レート

以下に無線 LAN デモシステムで測定した転送レートを示します。

受信	送信	備考
19.3 Mbps (*)	18.8 Mbps (*)	測定条件 FM4 の型格: MB9BF568R (core clock : 160 MHz) プロトコル : TCP 転送サイズ : 7.3 MB 認証モード : WPA2-PSK (AES-CCMP) 接続方式 : インフラストラクチャーモード 試験ボードとアクセスポイントの間は無線接続 (IEEE 802.11n) アクセスポイントと PC の間は有線接続

*: 参考値です。電波状況等の環境によって変化します。

6 改訂履歴

文書名: AN204461 - FM4 無線 LAN モジュール連携による WLAN ソリューション デモ操作マニュアル

文書番号: 002-04462

版	ECN	変更者	発行日	変更内容
**	–	NNAK	06/27/2014	スパンションアプリケーションノート MB9B560R_AN709-00004-1v0-J をサイプレスとして登録したものです。
*A	5625223	NNAK	02/09/2017	これは英語版の 002-04461 Rev. *A を翻訳した日本語版です。
*B	5899073	YSAT	10/03/2017	Cypress の新ロゴを適用。

セールス, ソリューションおよび法律情報

ワールドワイドな販売と設計サポート

サイプレスは、事業所、ソリューション センター、メーカー代理店、および販売代理店の世界的なネットワークを保持しています。お客様の最寄りのオフィスについては、サイプレスのロケーション ページをご覧ください。

製品

ARM® Cortex® Microcontrollers	cypress.com/arm
車載用	cypress.com/automotive
クロック&バッファ	cypress.com/clocks
インターフェース	cypress.com/interface
IoT (モノのインターネット)	cypress.com/iot
メモリ	cypress.com/memory
マイクロコントローラ	cypress.com/mcu
PSoC	cypress.com/psoc
電源用 IC	cypress.com/pmic
タッチ センシング	cypress.com/touch
USB コントローラー	cypress.com/usb
ワイヤレス／RF	cypress.com/wireless

PSoC® ソリューション

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6

サイプレス開発者コミュニティ

フォーラム | WICED IOT Forums | Projects | ビデオ | ブログ | トレーニング | Components

テクニカルサポート

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2014-2017. 本書面は、Cypress Semiconductor Corporation 及び Spansion LLC を含むその子会社（以下、「Cypress」という。）に帰属する財産である。本書面（本書面に含まれ又は言及されているあらゆるソフトウェア又はファームウェア（以下、「本ソフトウェア」という。）を含む）は、アメリカ合衆国及び世界のその他の国における知的財産法令及び条約に基づき、Cypress が所有する。Cypress はこれらの法令及び条約に基づく全ての権利を留保し、また、本段落で特に記載されているものを除き、Cypress の特許権、著作権、商標権又はその他の知的財産権のライセンスを一切許諾していない。本ソフトウェアにライセンス契約書が伴っておらず、かつ、あなたが Cypress との間で別途本ソフトウェアの使用方法を定める書面による合意をしていない場合、Cypress は、あなたに対して、（1）本ソフトウェアの著作権に基づき、（a）ソースコード形式で提供されている本ソフトウェアについて、Cypress ハードウェア製品と共に用いるためにのみ、組織内部でのみ、本ソフトウェアの修正及び複製を行うこと、並びに（b）Cypress のハードウェア製品ユニットに用いるためにのみ、（直接又は再販売者及び販売代理店を介して間接のいずれかで）エンドユーザーに対して、バイナリーコード形式で本ソフトウェアを外部に配布すること、並びに（2）本ソフトウェア（Cypress により提供され、修正がなされていないもの）に抵触する Cypress の特許権のクレームに基づき、Cypress ハードウェア製品と共に用いるためにのみ、本ソフトウェアの作成、利用、配布及び輸入を行うことについての非独占的で譲渡不能な一身専属的ライセンス（サブライセンスの権利を除く）を付与する。本ソフトウェアのその他の使用、複製、修正、変換又はコンパイルを禁止する。

適用される法律により許される範囲内で, Cypress は, 本書面又はいかなる本ソフトウェア若しくはこれに伴うハードウェアに関しても, 明示又は黙示をとわず, いかなる保証 (商品性及び特定の目的への適合性の黙示の保証を含むがこれらに限られない) も行わない。適用される法律により許される範囲内で, Cypress は, 別途通知することなく, 本書面を変更する権利を留保する。Cypress は, 本書面に記載のある, いかなる製品若しくは回路の適用又は使用から生じる一切の責任を負わない。本書面で提供されたあらゆる情報 (あらゆるサンプルデザイン情報又はプログラムコードを含む) は, 参照目的のためのみに提供されたものである。この情報で構成するあらゆるアプリケーション及びその結果としてのあらゆる製品の機能性及び安全性を適切に設計, プログラム, かつテストすることは, 本書面のユーザーの責任において行われるものとする。Cypress 製品は, 兵器, 兵器システム, 原子力施設, 生命維持装置若しくは生命維持システム, 蘇生用の設備及び外科的移植を含むその他の医療機器若しくは医療システム, 汚染管理若しくは有害物質管理の運用のために設計され若しくは意図されたシステムの重要な構成部分としての使用, 又は装置若しくはシステムの不具合が人身傷害, 死亡若しくは物的損害を生じさせるようなその他の使用 (以下「本目的外使用」という。) のためには設計, 意図又は承認されていない。重要な構成部分とは, それの不具合が装置若しくはシステムの不具合を生じさせるか又はその安全性若しくは実効性に影響すると合理的に予想できるような装置若しくはシステムのあらゆる構成部分をいう。Cypress 製品のあらゆる本目的外使用から生じ, 若しくは本目的外使用に関連するいかなる請求, 損害又はその他の責任についても, Cypress はその全部又は一部をとわず一切の責任を負わず, かつ Cypress はそれら一切から本書により免除される。Cypress は Cypress 製品の本目的外使用から生じ又は本目的外使用に関連するあらゆる請求, 費用, 損害及びその他の責任 (人身傷害又は死亡に基づく請求を含む) から免責補償される。

Cypress、Cypress のロゴ、Spansion、Spansion のロゴ及びこれらの組み合わせ、WICED、PSoC、CapSense、EZ-USB、F-RAM、及び Traveo は、米国及びその他の国における Cypress の商標又は登録商標である。Cypress の商標のより完全なリストは、cypress.com を参照のこと。その他の名称及びブランドは、それぞれの権利者の財産として権利主張がなされている可能性がある。