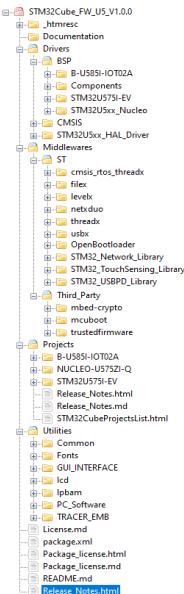




こんにちは。このプレゼンテーションでは、STM32U5 シリーズ用 STM32Cube ファームウェアパッケージについて説明します。

- **STM32CubeU5**:現時点で、マスマーケットに投入されていて、Web で入手可能な最も充実したファームウェアパッケージ

- 低消費電力機能(LPBAM を含む)
- セキュリティ TrustZone、TFM、SBSFU
- 初めて Azure RTOS をネイティブにサポート
- USBPD と USBX の完全共存
- オープンブートローダ USART、FDCAN、I2C、SPI、USB
- IoT と MXCHIP Wi-Fi のネットワークングと STM32WB BLE
- STM32CubeMX フォーマットの 204 個を含む 239 個の実行する準備が整っている例
- 新しい html 形式の readme ファイルにより改善されたドキュメント



STM32U5 シリーズ用の組み込みソフトウェア STM32Cube は、製品のマスマーケットに投入されていて Web 上でリリースされている、最も充実したファームウェアパッケージです。

改善やバグ修正をより早く反映するため、www.st.com だけでなく GitHub でも公開されています。GitHub では、ユーザの貢献や直接的なフィードバックを容易に得られるよう、プルリクエストやイシューを受け付けています。

STM32U5 製品のすべての機能について説明し、専用の例と LPBAM(低消費電力バックグラウンド自律モード)サブシステム用のヘルパユーティリティドライバとしては初めての提供を通じて、特に低消費電力機能に焦点を当てています。

セキュリティ機能には、TrustZone サポート、TFM スタック統合、および専用の SBSFU アプリケーションがあります。

このファームウェアパッケージは、初めて Microsoft® Azure® RTOS ミドルウェア(USBX、FileX/LevelX、ThreadX、NetX Duo)をネイティブにサポートしており、USBX と USBPD の完全共存を促します。

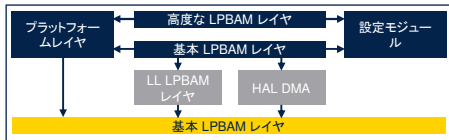
TouchSensing ライブラリやネットワークライブラリなどの社内ミドルウェアサポートに加え、STM32CubeU5 ファームウェアパッケージでは、外部ベースバンドおよび RF デバイス(B-U585I-IOT02A ディスカバリボードでサポートされている Wi-Fi 用 Mxchip および BLE 用 ST32WB)に依存する無線通信により、OpenBootloader IoT とコネクティビティも、このファームウェアパッケージの一部としてサポートしています。

各ボードに対して、サンプルとミドルウェアアプリケーションのセットが、EWARM、MDK-ARM、および STM32CubeIDE ツールチェーン用の事前設定されたプロジェクトとともに提供されます。

また、さまざまなファームウェアのディレクトリにある新しい html 形式の readme ファイルにより改善されたドキュメントについては、右の図を参照してください。

• **LPBAM サブシステムのサポートと図:**

- 基本レイヤおよび高度なレイヤで構成され、HAL ドライバに基づいた、**LPBAM** (ADC、DAC、DMA、GPIO、I2C、LPTIM、OPAMP、SPI、UART、VrefBuf) に含まれる **lps** 専用のユーティリティドライバ(最大 58 個の .c/.h ファイル)。



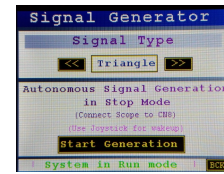
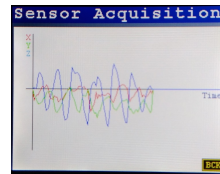
• **ドキュメント:**

- ドライバは独自のドキュメントを提供
- LPBAM 手引書(ハードウェアとファームウェアの説明を含む)
- NUCLEO-U575ZI-Q で開発された最大 9 個の実用的なユースケース

- NUCLEO-U575ZI-Q
  - Applications
    - LPBAM
      - LPBAM\_ADC\_InternalExternalChannelSwitch
      - LPBAM\_COMP\_InputSwitch
      - LPBAM\_DAC\_OPAMP\_ContinuousConversion
      - LPBAM\_DMA\_MultiQueueExecution
      - LPBAM\_I2C\_SequentialTransfer
      - LPBAM\_LPGPIO\_JOToggle
      - LPBAM\_LPTIM\_PWMGeneration
      - LPBAM\_LPUART\_TransmitReceive
      - LPBAM\_SPI\_FullDuplex\_Simplex



- STM32U575-EV の OoB デモ の 2 個の LPBAM 専用モジュール: 信号発生器およびセンサ取得



• **低消費電力機能:**

- 低消費電力モードで、サポートされている NUCLEO-U575ZI-Q、STM32U575-EV、および B-U585I-IOT02A ボード上にディスパッチされている機能を図で示す、**13 個のペリフェラルサンプル(HAL/LL)と MW アプリケーション**
- 異なるマイクロコントローラ電力モードを介して消費電力の標準値を測定する NUCLEO-U575ZI-Q ボード専用の「PWR\_ModesSelection」の例

STM32U5 シリーズには、低消費電力バックグラウンド自律モードサブシステムがあり、電力モードにかかわらず、またソフトウェアを実行していなくても、機能的かつ自律的なペリフェラルを使用できます。DMA リンクリスト転送により、さまざまな動作を連鎖させることができます。

LPBAM ユーティリティは、STM32CubeU5 ファームウェアパッケージの Utilities フォルダにあるモジュール化されたドライバのセットです。基本のレベルと高度なレベルの 2 つの抽象化をユーザに提供します。

LPBAM ユーティリティは、以下のレイヤに基づいています。

- 抽象化の最初のレベルは、基本 LPBAM レイヤと呼ばれるハードウェアに依存しないレイヤです。横断的な API を介して、シナリオ作成の細部をすべてユーザに示します。
- 抽象化の 2 番目のレベルは、高度な LPBAM レイヤと呼ばれるハードウェアに依存しないレイヤです。エンドユーザアプリケーションを構築するためにカスタマイズおよび連結可能な、事前定義されたシナリオ(基本ノードのセット)を提供します。
- プラットフォームレイヤには、実用側で使用されるデバイス固有の定数が含まれます。
- LL(ローレベル)LPBAM レイヤは、基本および高度な LPBAM レイヤによって使用されます。サポートされている各ペリフェラルに、デバイス固有のメカニズムが含まれています。

Nucleo-U575ZI-Q で開発された 9 つの実用的なユースケースにより、ユーザはこの新しいクラスのドライバに慣れることができます。

STM32U575-EV 出荷時デモで使用可能な追加の専用モジュール: 信号発生器とセンサ取得。

LPBAM のほか、複数の例に STM32U5 が提供するさまざまな低消費電力機能が実装されます。

TFM の機能	STM32CubeU5 の TF-M (新機能とSTM32CubeL5 の TF-M の比較)
TFM バージョン	TFM オープンソースバージョン TF-M v1.0-RC2(2019 年 10 月)および mcu_boot オープンソースバージョン v1.5.0(2020 年第 1 四半期)に基づく
管理されたファームウェアイメージのスロット数	SWAP モードを使用可能な 2 つのイメージのスロットまたは 1 つのイメージのスロット (内部 Flash のみ) 不変のスタンダアロンのローカル・ローダを介してダウンロードされた新しいイメージ
管理されたファームウェアイメージ数	1 つのファームウェアイメージ (セキュアパーツと非セキュアパーツの組み合わせ) または 2 つのファームウェアイメージ (1 つのセキュアファームウェアと 1 つの非セキュアファームウェア)
ファームウェアの更新タイプ	完全なファームウェアイメージの更新
ファームウェアイメージの暗号方式	AES ファームウェア暗号化を使用した ECDSA または RSA 非対称暗号化方式 (フル MbedCrypto ソフトウェアまたはミックス MbedCrypto ソフトウェアおよび DPA ハードウェア)
隔離レベル	2 つの隔離レベル: セキュア/非セキュア + 特権/非特権
セキュアサービス	完全な暗号化サービス (フル MbedCrypto ソフトウェアまたはミックス MbedCrypto ソフトウェアおよび DPA ハードウェア) 初期証明 セキュア・ストレージ (データの暗号化/整合性) 内部信頼ストレージ (データ整合性) 非特権アプリケーションサービスを統合する準備ができたアーキテクチャ
サポートされている IDE	IAR / Keil / STM32CubeIDE

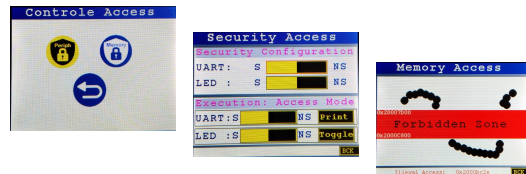


### TFM および SBSFU:

- 1 個の SBSFU および 1 個の TFM のアプリケーション
- TFM アプリケーションのマニュアル**
  - UM2851 (Rev1): Getting started with STM32CubeU5 TFM application
  - AN5447 (Rev3): Arm® TrustZone® STM32 マイクロコントローラでのセキュア・ブートおよびセキュア・ファームウェア更新ソリューションの概要

### TrustZone

- サポートされている NUCLEO-U575ZI-Q、STM32U575-EV、および B-U585I-IOT02A ボード上にディスパッチされている、7 個の専用の TrustZone の例/アプリケーション
- 1 個の STM32U575-EV の OoB デモ (エミュレーション) 専用のモジュール



4

Arm Trusted Firmware-M (TF-M) は、TrustZone 用の Arm プラットフォーム・セキュリティ・アーキテクチャ (PSA) のリファレンス実装です。

STM32Cube U5 マイクロコントローラ・パッケージには、2 つの異なる例が用意されています。

- 1 番目は、TF-M の完全実装です。
- TF-M のセキュア・ブートとセキュアファームウェア更新機能のみを実装する 2 番目のアプリケーションである STM32CubeU5 SBSFU も使用できます。

TF-M リファレンス実装では、オープンソース MCU ブートに基づいたセキュア・ブートおよびセキュア・ファームウェア更新サービスと、ユーザアプリケーションのランタイムに使用可能な一連のセキュアサービスを提供します。

この表は、セキュア・ブートおよびセキュア・ファームウェア更新アプリケーションの主な機能を示しています。

動的ファームウェア更新は、2 つのスロットとスワップ機能によりサポートされます。

不変のスタンダアロンのローカル・ローダが、暗号化されている可能性があるイメージをダウンロードします。

セキュアと非セキュア、および特権と非特権の 2 つの隔離レベルが実装されます。

セキュアサービスとは:

- セキュア・ストレージ・サービス
- 証明
- 暗号化サービス
- TF-M 監査ログ
- プラットフォームサービス

IAR、Keil、および STM32CubeIDE の 3 つの IDE 用に、7 つの専用の TrustZone のサンプルとアプリケーションを使用できます。

## STM32CubeU5

## 初めて Azure RTOS をネイティブにサポート

- **STM32CubeU5 ファームウェアに装備:**
  - Azure RTOS ミドルウェアのサポート
  - 初めての STM32CubeMX (v6.3.0 以降) でのネイティブモード
  - 合計 33 個のリリース済みアプリ
  - サポートされている IDE: IAR、STM32CubeIDE、および MDK-ARM
- **主な新機能:**
  - ネットワークライブラリの初めての統合 (WiFi + NetXDuo)
  - ThreadX MPU と TZ アプリ (MSFT からの主要な修正 CM33)
  - ThreadX LowPower アプリケーション
  - FileX の新しい BareMetal (スタンドアロン) モードとアプリケーション
  - 動的ページが搭載された WebServer
  - USBX-USBPD の初めての共存 (完全統合 + アプリケーション)
  - ユーザのより高い柔軟性を可能にする FileX/NetXDuo Low Layer ドライバの新しいアーキテクチャ

MW	App	H7	U5
Total Nbr Of Comitted Apps		27	33
ThreadX	Tx_Thread_Creation	X	X
ThreadX	Tx_Thread_MsgQueue	X	X
ThreadX	Tx_Thread_Sync	X	X
ThreadX	Tx_LowPower	-	X
ThreadX	Tx_MPU	N/A	X
ThreadX	Tx_TrustZone	N/A	X
ThreadX	Tx_FreeRTOS_Wrapper	X	X
ThreadX	Tx_CMSIS_Wrapper	X	X
FileX	Fx_SRAM_File_Edit_Standalone	-	X
FileX	Fx_USD_File_Edit	X	X
FileX	Fx_IAP	X	X
FileX	Fx_MultiAccess	X	X
FileX	Fx_DualInstance	X	X
FileX	Fx_NOR_Write_Read_File	X	X
FileX	Fx_NAND_Write_Read_File	X	X
NetXDuo	Nx_TCP_Client	X	X
NetXDuo	Nx_TCP_Server	X	X
NetXDuo	Nx_UDP_Client	X	X
NetXDuo	Nx_UDP_Server	X	X
NetXDuo	Nx_SNTP	X	X
NetXDuo	Nx_MQTT_Client	X	X
NetXDuo	Nx_WebServer	X	X
NetXDuo	Nx_Perf	-	-
USBX	Ux_Device_HID	X	X
USBX	Ux_Device_CDC_ACM	X	X
USBX	Ux_Device_MSC	X	X
USBX	Ux_Device_DFU	-	X
USBX	Ux_Device_CDC_ECM	X	X
USBX	Ux_Device_HID_CDC_ACM	X	X
USBX	Ux_Host_HID	X	X
USBX	Ux_Host_MSC	X	X
USBX	Ux_Host_CDC_ACM	X	X
USBX	Ux_Host_DualClass	X	X
USBX	Ux_Host_Composite	-	-
USBXP	USBPD_SINK_Ux_Device_HID_CDC_ACM	N/A	X
USBXP	USBPD_SRC_Ux_Host_MSC	N/A	X



Life.augmented

5

STM32CubeU5 ソフトウェア・スイートは、次を含む Microsoft Azure RTOS ミドルウェアで補完されます。

- Azure RTOS ThreadX リアルタイムオペレーティングシステム
- Azure RTOS FileXFAT ファイルシステム、フォールトトレラント
- Azure RTOS NetX/NetX Duo TCP/IP スタック
- Azure RTOS USBXUSB スタック、ホストとデバイス、および USBPD との共存

Azure RTOS ミドルウェアでサポートされている機能のリストを右の表に示します。

Azure は STM32CubeU5 でネイティブにサポートされています。 Azure RTOS が有名な STM32Cube ソフトウェア・スイートにもたらず追加の主な利点:

- 高速な性能
- 一貫した完全なソリューション
- 業界認証

- STM32U5 2M は、**USBX** と **USBPD** が完全に統合され、表示された**最初の製品**
- STM32CubeU5 のファームウェアパッケージで提供されている**2つのアプリケーション**:
  - USBPD\_SINK\_Ux\_Device\_HID\_CDC\_ACM
  - USBPD\_SRC\_Ux\_Host\_MSC
- **主な機能**:
  - **完全な電源供給およびデータ**
  - **完全に統合された ThreadX**
  - **完全検証および事前の認定テスト**



USB Type-C



USB Type-C コネクタは、ホストとペリフェラルの間の従来の USB 通信と同時に、Power Delivery プロトコルを有効にします。

STM32U5 は、Power Delivery インタフェースとホスト/デバイス USB2.0 インタフェースの両方を内蔵しています。

STM32CubeU5 は、USBX と USBPD が完全に統合された初めてのファームウェアパッケージです。USBX には、多くのクラスが装備されている次のようなホストおよびデバイスのスタックが含まれています。

ファームウェアパッケージでは、次の 2 つのアプリケーションが提供されています。

- USBPD\_SINK\_Ux\_Device\_HID\_CDC\_ACM: STM32U585xx ボード上で Azure® RTOS USBX スタックを使用する USB-PD Type-C™ Consumer および USB デバイス
- USBPD\_SRC\_Ux\_Host\_MSC に、Azure® RTOS USBX スタックの使用量のサンプルを提供します。USB Flash のリムーバブルディスクでのエニュメレーションおよび通信が可能な USB ホストのマスタストレージ(MSC)の開発方法を示します。

完全検証および事前の認定テストにより、USB データと USB Power Delivery のアプリケーションの設計が容易になります。

- **主な機能:**

- OpenBL V5.0.0
- **UART、I2C、SPI、FDCAN、USB-DFU** のサポート
- 完全な **MISRA-C 2012**
- STM32CubeProgrammer および標準システムブートローダと完全互換
- **USBX と ThreadX の初めての積分**(従来の ST USB スタックとの置換)
- **B-U585H-IOT02A** で実行中の **1 個のプロジェクト**(BSP 依存関係なし)
- **ドキュメント**: AN3154、AN3155、AN3156、AN4221、AN4286、

- **サポートされるコマンド**

- 識別コマンド
  - バージョンの取得コマンド
  - デバイス ID の取得コマンド
  - 使用可能なコマンドリストの取得コマンド
- メモリ書き込みコマンド(ユーザ Flash、SRAM、オプションバイト、OTP)
- メモリ読み出しコマンド(ユーザ Flash、SRAM、オプションバイト、OTP、SysMem)
- 書き込み保護コマンド(オン/オフ)
- 読み出し保護コマンド(オン/オフ)
- アプリケーション A へのジャンプコマンド
- Flash 消去コマンド

- **サポートされるツール**

- オープンブートローダは、既存の標準ブートローダと互換性あり
- STM32CubeProgrammer でサポートされているすべてのコマンドは、ツールで変更せずに使用できます。



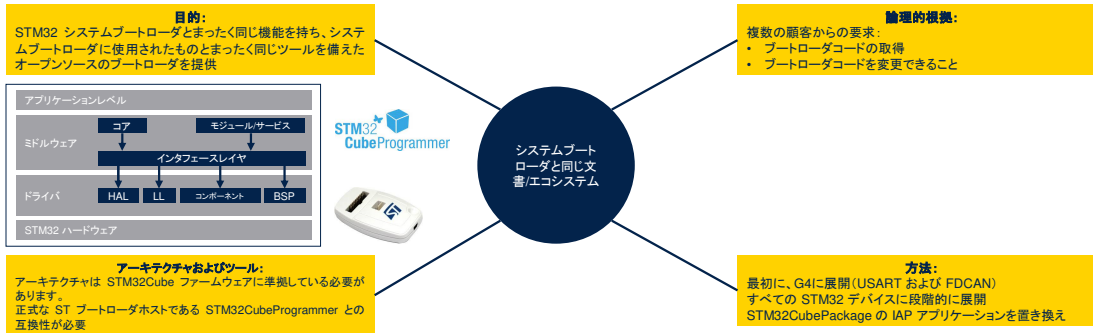
ほとんどの Flash メモリベースのシステムで重要な要件は、最終製品にインストールした後にファームウェアを更新できることです。この機能は、アプリケーション内プログラミング (IAP) と呼ばれます。オープンブートローダは、STM32Cube ファームウェアパッケージおよび GitHub で提供される IAP です。これは、独自のブートローダを構築したい顧客なら誰でも使用できるサンプルです。その主なタスクは、使用可能な通信インタフェース (USART/I2C/SPI/USB-DFU および FDCAN) のいずれかを使用して、デバッガを使わずにアプリケーションプログラムを内部ユーザメモリ (Flash/SRAM/OTP) にダウンロードすることです。USB ブートインタフェースに関しては、STM32U5 は、Azure USBX と ThreadX RTOS の統合をサポートする初めてのマイクロコントローラです。

オープンブートローダは、STM32 CubeProgrammer などのホストアプリケーションにサービスを提供して、インタフェースリンク (USART/I2C/SPI/USB-DFU & FDCAN) を介してデバイスにファームウェアをダウンロードし、このファームウェアを必要なユーザメモリにインストールします。

オープンブートローダは、異なるクロックやインタフェース設定などのハードウェアシステムの初期化を、STM32CubeFirmware HAL/LL ドライバに依存しています。

オープンブートローダは、非セキュア・ドメインの Cortex-M プロセッサによって実行されます。選択された通信リンクを通じて STM32 に渡されたコマンドのリストが、スライドの右側に表示されます。この実装では、各プロトコルの 1 つのインスタンスのみが使用され、これらのプロトコルのそれぞれに対して、1 セットのデフォルトピンが使用されます。

その後、ユーザは必要に応じてインスタンスまたはピンを変更できます。



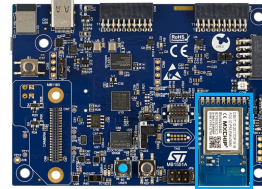
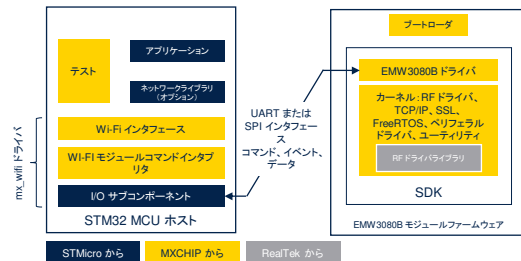
この図は、オープンブートローダの機能をまとめたものです。オープンブートローダの目的は、STM32 システムブートローダとまったく同じ機能を持ち、まったく同じエコシステムを持つオープンソースのブートプログラムを提供することです。オープンブートローダは、同じドライバとミドルウェアに基づいて、STM32Cube ファームウェアアーキテクチャに準拠しています。正式な ST ブートローダホストである STM32CubeProgrammer との互換性が保証されています。ソースコードを提供することで、顧客はブートプログラムをそれぞれのニーズに合わせることができます。オープンブートローダは、STM32CubePackage の IAP アプリケーションを段階的に置き換えます。



- **EMW3080B の主な機能:**
  - 802.11b/g/n をサポート - サポートステーション、ソフト AP、ステーション + ソフト AP
  - 最大 133 MHz の周波数と 256 KB の SRAM と 2 M Flash を備えた ARM CM4F CPU
  - ハイスピード UART、I2C、SPI、PWM、およびマルチ GPIO をサポート
  - WLAN MAC/ベースバンド/RF
- **設定:**
  - 2 つの通信モード:
    - バイパス: STM32 ホストと EMW3080B の間で交換される元データ
    - AT コマンド: EMW3080B 上で動作する TCP/IP
  - インタフェース: UART または SPI (B-U585I-IOT02A 用の SPI)
  - ホストドライバは RTOS の有無にかかわらず動作する
- **最大 7 つのミドルウェアアプリケーションと 1 つのデモ**
  - 1 つの Wi-Fi の基本動作: スキャン、ping、リモートエコーサーバ、http ダウンロードのデモを行う - 設定: **ペア OS および AT コマンド**
  - 6 つのアプリケーション: NetXDuo - Azure RTOS NetX/NetXDuo スタックの使用量 - バイパスモード
  - 1 つのデモンストレーション IOT\_HTTP\_WebServer: Web サーバは B-U585I-IOT02A で、Web クライアントはユーザのブラウザ - 設定: **ペア OS および AT コマンド**



## ソフトウェアアーキテクチャの概要



B-U585I-IOT02A ボードへの直接のはんだ付け

9

このスライドでは、STM32U5 への Wi-Fi コネクティビティを提供します。MXCHIP 製 EMW3080B デバイスは、内蔵ワイヤレスインターネットコネクティビティデバイスです。このモジュールは SPI または UART インタフェースを使用して MCU と通信します。EMW3080B モジュールは、B-U585I-IOT02A ボードの上面図に示すように、ボードに直接はんだ付けされます。次のように別のボードに付けることもできます。

- Arduino コネクタを介して任意の STM32 ボードに接続するための EXT-AT3080 シールド
- STMOD+ コネクタを介して任意の STM32 ボードに接続するための MB1400 シールド

Wi-Fi® モジュールはオペレーティングシステムを必要とせず、AT コマンドのみでコネクティビティを確立する完全に統合された TCP/IP スタックを備えています。ソフトウェアアーキテクチャは、EMW3080B モジュールと通信する mx\_wifi ドライバに依存しています。MXCHIP が開発したテストアプリケーションや、ST が開発したネットワークライブラリおよびアプリケーションを、ドライバに加えて使用することができます。7 つのミドルウェアアプリケーションと 1 つのデモにより、この Wi-Fi モジュールに慣れることができます。

### 主な機能と BLE AT のセットアップ

- B-U585I-IOT02A ボード、BLE\_AT\_Server アプリケーションを実行している BLE モジュールである STM32WB5MMG を内蔵している
- STM32U5 は、AT コマンドを使用して BLE モジュールと相互作用する
- STM32U5 は BLE\_AT\_Client アプリケーションを実行する
  - コマンドは UART リンクを介して転送される



### BLE AT - アプリケーション

- BLE アプリケーションは、「B3 user」ボタンで選択される
  - 心拍数センサ
  - P2P サーバ
- スマートフォン用アプリケーションである「ST BLE Sensor」に接続した後でユーザーができること:
  - ボード上の B3 ボタンをクリックして、スマートフォンに値を送信する
  - P2P サーバのスマートフォンのアクションからの書き込み受信時にボードの LED をトグルする
- U5 コンソールに、転送された命令やコマンドが表示される

```
TX: AT
RX: OK
TX: AT+BLE_SVC=1
RX: OK
RX: <BLE_EVT_CONN=1
A remote device is now connected
Send a BLE AT command to notify BLE application
TX: AT+BLE_NOTIF_VAL=1,2,1
RX: OK
RX: <BLE_EVT_WRITE=1,2,1
RX: <BLE_EVT_WRITE=1,2,0
RX: <BLE_EVT_CONN=0
The remote device is now disconnected
```



P2P サーバのスマートフォンアプリケーション

10

このスライドでは、STM32U5 への Bluetooth low energy technology のコネクティビティを提供するモジュールについて説明します。

STM32WB マイクロコントローラの革新的なアーキテクチャは、リアルタイム実行(無線関連のソフトウェア処理)用に最適化され、柔軟性の高いリソースの使用ならびに電源管理を可能にした、2つの完全に独立したコアに基づいています。

このモジュールは、B-U585I-IOT02A ボードに内蔵されています。

STM32U5 との通信は、UART リンク上で転送される AT コマンドに基づいて行われます。

次のアプリケーションは、この BLE モジュールの機能のデモに使用できます。

- 心拍数センサ
- ポイントツーポイントサーバ

スマートフォンにインストールされた ST BLE センサアプリケーションに接続すると、ユーザーは LED やボタンを使用して、モジュールとスマートフォンの間の通信を確認できます。

転送される命令およびコマンドが STM32U5 コンソールに表示されます。キャプチャを参照してください。

# STM32CubeU5

# デモ

## B-U585I-IOT02A Web サーバのデモ

- コネクティビティ機能 (Wi-Fi) およびセンサデータ取得に光を当てた組み込み開発による動的な Web ページの開発とナビゲーションを組み合わせる

### 動的な Web ページ

- ページコンポーネント (React JS/Bootstrap ライブラリ)
- チャートプロット (Chart JS ライブラリ)
- HTTP リクエストのエンコード (Java Script)
- HTTP レスポンスの解析とデータのプロット (Java Script)



### 組み込まれたデモ

- Wi-Fi 管理
  - MxChip WiFi ファームウェア (バイパスなし)
  - STM32 ネットワークライブラリ
  - HTTP プロトコルの通信
    - GET リクエストのレスポンス (JSON)
- 内蔵センサのデータ取得
  - 温度センサ
  - 圧力センサ
  - 湿度センサ



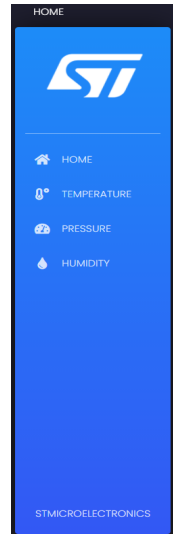
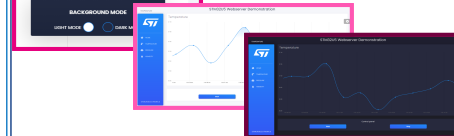
## 選択時のリアルタイムセンサデータ取得



## デバイスの互換性



## バックグラウンドモード



このスライドでは、Wi-Fi Web サーバのデモについて説明しています。

このデモファームウェアは、STM32CubeU5 マイクロコントローラパッケージに含まれています。このファームウェアでは、STM32Cube HAL を使用して HTTP リクエストを実行するための MXCHIP モジュールの使用方法を示しています。

MXCHIP モジュールと Web ブラウザ (この場合は Google Chrome™ ブラウザ) は、Web サーバを作成するために使用されます。この Web ページでは PC および電話の使用をサポートしています。

B-U585I-IOT02A ボードは、このデモでは HTTP サーバです。これには、各クライアントのリクエスト後に送信された Web ページリソースが含まれています。

B-U585I-IOT02A ボードでは、クライアントリクエストのデコード、処理、および応答が可能です。

- HTML ページ、CSS ファイル、JS ファイルなどの Web ページのリソースリクエスト
- センサ値リクエスト (温度、圧力、および湿度の値)

ユーザは、サーバの IP アドレス (ネットワークインタフェースプロセス後に提供される) をコピーしてブラウザに貼り付けることで、Web ページの閲覧を開始できます。Web ページは、クライアントサーバに読み込まれます。

この Web ページには 4 つのメインメニューがあります。

- 製品と Web ページの説明を含む Home メニュー、Temperature メニュー、Pressure メニュー、Humidity メニュー。最後に、ユーザは任意のメニューを選択して別のセンサーの取得を開始/停止できます。

## DCMI\_ContinuousCap\_EmbeddedSynchMode Example Description

- This example provides a description of how to configure DCMI peripheral in Continuous Mode and EmbeddedSynchronization mode with suspend and resume of the frame capture based on the STM32Cube HAL API when interrupts is not enabled (L21294).
- The project is targeted to run on STM32U575xx devices on STM32U575xx boards from STMicroelectronics.
- The project configures the maximum system clock frequency at 160MHz.
- The example describes how to configure the camera in continuous mode (DMA resolution, with synchronization mode set to Embedded Synchronization and to suspend/resume the capture).
- Embedded Synchronization Mode: The Synchronization of the frame is controlled by embedded codes set by the user and enabled by their dedicated LDRSRM values, these codes should be set the same way as the camera module itself should be able to support this mode.
- First push the user button to start the frame acquisition that will be displayed on the led. Then the user button is used to freeze(Suspend) and Unfreeze(Resume) the capture. The leds of the STM32U575xx EV board are used to monitor the status as following:
  - The GREEN led toggles for each new frame capture.
  - The GRP
  - The RED

## ICACHE\_Memory\_Remap Example Description

- How to encode code from a remapped region configured through the ICACHE HAL driver.
- This example describes how to remap the external SDRAM to C-bus and execute code located in this memory.
- First the code runs on the Green led from external SDRAM before remap, then it will jump to other memory remapping.
- This project is targeted to run on STM32U575xx devices on STM32U575xx boards from STMicroelectronics.
- At the beginning of the main program the HAL\_FLASH function is called to read and set the conditions, initialize the flash interface and the crystals. The SystemClock\_Config() function is used to configure the system clock by STM32U575xx devices.

## Notes

- Care must be taken when using HAL\_Delay, this function suspends all processes, then the LED register that interrupt priority you have to use HAL\_NVIC\_SetPriority.
- The application needs to ensure that the fly-back wire

## Keywords

- DCMI, Embedded
- STM32U575xx EV board's LED can be used to monitor the
- Green LED is off when no error detected
- Red LED is ON when any error occurred

## Directory of

- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap

## Hardware an

- This example runs on STM32U575xx devices without
- This example has been tested with STM32U575xx

## How to use it?

- In order to make the program work, you must do the follow

## Tx\_LowPower Application Description

The application consists of example of how to use the TX\_LowPower mode. It shows how to configure the TX\_LowPower mode, it demonstrates how to configure the TX\_LowPower mode of the TX\_LowPower mode, in addition it shows how to use the TX\_LowPower mode in the LowPower mode. The main entry function is Tx\_LowPower\_Init, it is called by the main program. The main program is Tx\_LowPower\_Init, it is called by the main program.

## Notes

- It is to ensure the TX\_LowPower mode.
- It is to ensure the TX\_LowPower mode.
- It is to ensure the TX\_LowPower mode.

## Keywords

- DCMI, Embedded
- STM32U575xx EV board's LED can be used to monitor the
- Green LED is off when no error detected
- Red LED is ON when any error occurred

## Directory of

- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap
- DCMI\_ContinuousCap

## Hardware an

- This example runs on STM32U575xx devices without
- This example has been tested with STM32U575xx

## How to use it?

- In order to make the program work, you must do the follow



スタンドアロンの html 形式の readme ファイルは、サンプルやアプリケーションが含まれるディレクトリにあります。これには、ユーザが容易にプログラムを理解し、独自のプロジェクトを迅速に開発できるよう、プログラムの目的と期待される結果が示され、また役立つ注意事項が含まれます。

# Our technology starts with You

© STMicroelectronics - All rights reserved.

ST logo is a trademark or a registered trademark of STMicroelectronics International NV or its affiliates in the EU and/or other countries.

For additional information about ST trademarks, please refer to [www.st.com/trademarks](http://www.st.com/trademarks).

All other product or service names are the property of their respective owners.



このプレゼンテーションにご参加いただき、ありがとうございました。