



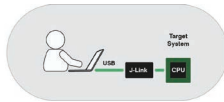
J-Link リモートアクセス

離れた場所にある開発チームをつなぐリモート対応機能

J-Link Software のリモートアクセス機能「J-Link Remote Server」を利用して開発者は、遠隔地からアクセスし、デバッグ対応を行う事ができます。リモートアクセス情報は、暗号化を有効にして、第三者からのアクセスデータの盗聴を防止することが可能です。この機能により、リモートオフィス／在宅勤務における開発業務を効率的に進める事ができます。**J-Link シリーズすべての製品で無償利用可能です。**

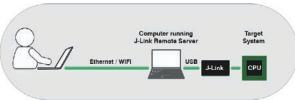
ユーザメリット

- 遠隔地／オフィス／工場にある開発ボードでリモートデバッグ対応が可能
- グローバルな開発チームで、ターゲットボードをコントロール



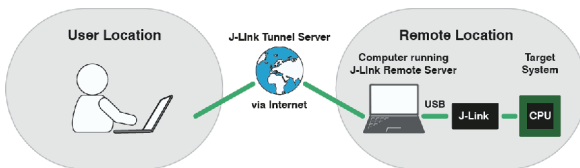
ダイレクトモード

通常のターゲットデバッグ接続方法、USB 経由で開発者様の PC へ接続します。J-Link PRO / J-Trace PRO では Ethernet 経由での接続もサポートしています。



LAN 接続モード

J-Link リモートサーバの LAN 接続により、ローカルネットワークを介したターゲット接続を行います。移動が難しい大きなハードウェア開発や複数のターゲット・J-Link を接続したテスト工程で利用頂けます。このモードでは、ファイアウォールで内部へのアクセスを許可されない限り、外部から J-Link へアクセスできません。



トンネルモード

リモートサーバのトンネルモードを利用し、開発者はインターネット経由で J-Link (および接続されたターゲットデバイス) に接続できます。これにより世界中のどこにあるターゲットデバイスへ接続・フラッシュ書込、デバッグ対応ができます。リモートオフィス・在宅勤務でのターゲットデバイス利用、数量の限られた試作基板をグローバルな拠点から早期にアクセスし、利用する事ができます。

J-Link リモートサーバは、「Flasher シリーズ」とも互換性を持っていますので、リモートプログラミング実行機能などプログラマとしての機能を制御できます。これにより、遠隔地にある工場への対応を迅速に行う事ができます。

※「Flasher シリーズ」へ「J-Flash/J-Flash SPI」を利用した制御はできません。「J-Link Commander」を利用した制御は可能です。

トンネルサーバについて

デフォルトのトンネルサーバは、SEGGER で管理しているサーバが指定されています。

このサーバで数千の接続を同時に処理できます。SEGGER のトンネルサーバは、J-Link ユーザーであれば無償で利用頂けます。

お客様のセキュリティポリシー上、このような共用サーバが許可されない場合、独自のトンネルサーバを利用し、自社のグローバルネットワーク内での接続を確立することも可能です。

! J-Link リモートアクセスのご注意事項

ダイレクトモード以外のご利用、トンネルモード・LAN 接続モードについて、無償サービス提供の扱いになり、パフォーマンス保証・不具合時対応の保証対象外となります。当社の日本語テクニカルサポートについても本無償サービスについては、対象外となりますので、お客様でご検証の上、ご利用ください。トンネルモードを介して、当社よりテクニカルサポートサービスは行っておりませんので、自社プロジェクト内でのみご利用ください。

J-Link リモートサーバ接続手順

■ J-Link の接続

ターゲットボードと J-Link を接続し、USB または LAN 経由で PC へ接続

■ J-Link Remote Server を起動

J-Link ソフトウェアインストールパッケージに含まれている「JLinkRegistration.exe」

■ J-Link Remote Server 設定

- ① J-Link connection：PC と J-Link の接続を選択
- ② Server setting：「LAN モード」または「トンネルモード」での接続を選択。お客様の独自トンネルサーバを利用する場合は、サーバ名、ポート番号を指定ください。
- ③ サーバ情報と資格情報を入力。暗号化データ転送を利用する場合は、「Encrypt transferred data」にチェックを入れてください。
※暗号化データ転送でパフォーマンスは低下します。

リモートサーバ設定完了：
クライアント未接続状態

リモートサーバ：
クライアント接続状態

■ クライアントの接続

ご利用の IDE から GUI ベースでの接続設定や J-Link Commander を利用して、リモートサーバの J-Link へ接続

```
C:\Program Files\SEGGER\J-Link\JLink.exe
No emulators connected via USB.
Do you want to connect to a J-Link via TCP/IP instead?
Connection method: [J-Link tunnel]
Identifier: may be one of the following:
Serial number (e.g. 28120000)
Hardware ID (e.g. 000000000000)
Identifier: MyLink
Tunnel server: jlink-segger.com
Port: 19020
Password: *****
[Yes] [No]
```

```
C:\Program Files\SEGGER\J-Link\JLink.exe
type "connect" to establish a target connection, "?" for help
J-Link
Please specify device / core, defaults: PROGRAMMER
Device selection string
Please specify target interface:
0: JTAG (default)
1: SWD
Please specify target interface speed [Hz], defaults: 0000 Hz
Device "RT1074AD0V" selected.
Connecting to target via net
Could not connect to remote
Please specify device / core, defaults: PROGRAMMER
Device selection string
Please specify target interface:
0: JTAG (default)
1: SWD
Please specify target interface speed [Hz], defaults: 0000 Hz
Device "RT1074AD0V" selected.
```

■ リモート接続完了後

ダイレクト接続されている J-Link と同じように操作が可能です。(動作パフォーマンスはネットワーク品質に準じます。)

IAR EWARM

GDB/Eclipse

より簡単にリモートデバッグ利用を実現するには

Ethernet 接続機能をサポートしている「J-Link PRO」、「J-Trace PRO」を利用して、VPN 接続を構築 Ethernet 接続インターフェースを持つ製品は、PC リモートサーバを使うことなく、御社内の VPN に組み込むことで、御社セキュリティポリシーに準拠した利用方法でリモート接続・デバッグを実現します。



J-Link PRO
10/100BASE LAN 搭載



J-Trace PRO for Cortex
10/100/1000BASE LAN 搭載



J-Link Software

J-Link を最大限に活用するバンドルソフトウェア

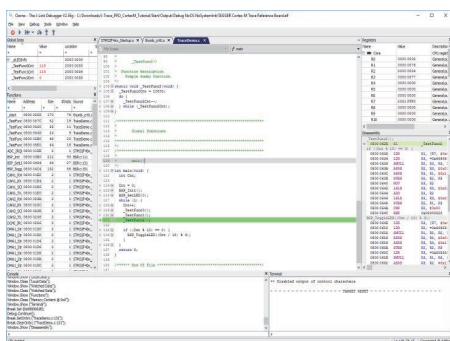


ソフトウェアデバッグ・開発支援

J-Link Debugger 「OZONE」

対応 CPU コア ARM / RISC-V

J-Link バンドルデバッグソフトウェア (J-Link PLUS 以上)



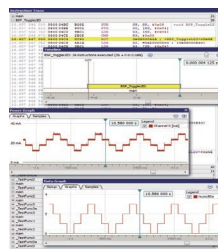
フリーの GNU コンパイラや IAR EWARM、Keil MDK など商用コンパイラから生成されたデバッグ情報含む ELF ファイルを利用してデバッグができる J-Link バンドルソフトウェアとなります。

OZONE はデバッグとしてのフル機能（命令トレース、パワグラフ、ライブウォッチ、リアルタイムターミナル I/O など）を備えています。

これまで高価なデバッグが必要であった関数プロファイリングやコードカバレッジなども J-Trace や J-Link との組み合わせで実現することができます。デバッグ人員の増減にも対応しやすく、高いコストパフォーマンスを実現します。

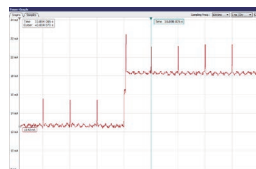
パフォーマンスアナライザ

J-Link と J-Trace PRO の機能により、ターゲットシステムのパフォーマンス分析に使用できます。J-Trace PRO と組み合わせることで、アプリケーションの実行時間を記録し分析することができます。タイムラインで視覚化してファンクションや割り込みの経路を簡単に表示することができます。J-Link の高速サンプリング技術 (RTT) により、「OZONE」はシステム変数を時系列に可視化することができます。



電力プロファイリング

ターゲットデバイスの電力消費量をデバッグセッションで記録できます。これは特定のコンポーネントを有効/無効にするときの消費電力の変化を知る必要がある場合に特に役立ちます。



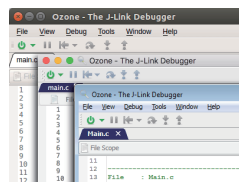
テストの自動化

デバッグのワークフローを自動化するスクリプトにより、テストを自動化して、進めることができます。デバッグのほとんどの機能をスクリプトにより制御可能で、開発生産性を向上。



マルチプラットフォーム対応

Arm / RISC-V コアに対応した「OZONE」は Windows、Mac、Linux のクライアント OS で利用することができます。利用するコンパイラもプロジェクトに合わせて設定可能なため、より生産的な開発ツールとして、利用可能です。

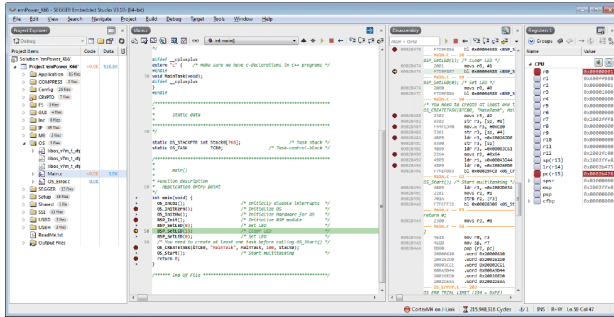


J-Link Unlimited Flash-BP

対応 CPU コア

ARM / Cortex

マイコンのブレークポイント数を無制限に設定可能な拡張機能（J-Link PLUS 以上）



「J-Link Unlimited Flash-BP」拡張機能を使うことで、ユーザーはフラッシュメモリでデバッグするとき、無制限のブレークポイントを設定できます。この機能がないデバッグでは、フラッシュに設定できるブレークポイント数は、CPU のデバッグユニットでサポートされているハードウェアブレイクポイントの数に制限されます。（ARM 7/9 では 2、Cortex-M では 4-6）

利用可能な開発ツール

SEGGER 社製： J-Link Debugger 「Ozone」 / SEGGER Embedded Studio

他社製： J-Link DLL、J-Link RDI 接続をサポートする統合開発環境（IAR EWARM / Arm MDK-ARM など）

Monitor Mode Debug

対応 CPU コア

Cortex-M

モータ制御 / 無線通信アプリケーションに最適なモニターモードデバッグ機能



J-Link モニタモードデバッグを使用すると、実行優先順位の高いコードを通常通り実行しながら、優先順位の低いアプリケーションを停止・実行することができます。これにより、開発者に新しい手法のアプリケーションデバッグ手法を提供します。通信タスクを維持したまま、割り込み、イベント、その他の信号を任意でオン/オフできるため、従来の手法ではデバッグが困難だった多様なシステム状況を開発者は把握し、デバッグする事ができます。

モニターモードでバグが有効なアプリケーション

■ PWM を利用したモータ制御アプリケーション

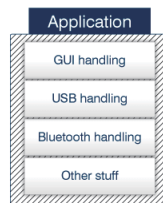
モニターモードデバッグを利用することにより、PWM 制御を続けながら、アプリケーションデバッグする事が可能です。

■ Bluetooth 通信アプリケーション

モニターモードデバッグを利用することにより、Bluetooth を「keep alive」実行続けながら、デバッグ処理を行う事ができます。

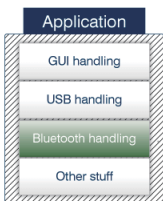
■ TCP/IP イーサネット通信アプリケーション

CPU を停止することにより、IP スタックなどが全く応答しなくなります。モニターモードデバッグでは TCP/IP パケット通信をキープアライブしながら、デバッグ処理する事ができます。



■ 通常のデバッグ

通信も停止してしまうため、通信中の挙動が見えない。



■ モニターモードデバッグ

通信部分の続けながら、デバッグ対応



利用可能な開発ツール

SEGGER 社製： J-Link Debugger 「Ozone」 / SEGGER Embedded Studio

他社製： IAR EWARM / Arm MDK-ARM

J-Link Software

J-Link を最大限に活用するバンドルソフトウェア

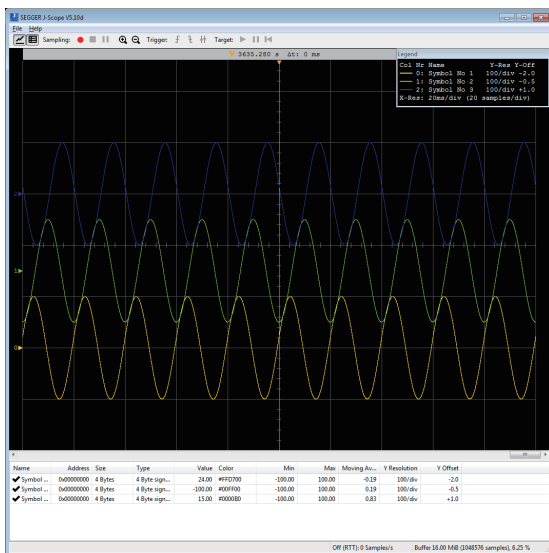
ソフトウェアデバッグ・開発支援

J-Scope

RAM モニタリング機能

対応 CPU コア

Cortex-M



ハードウェアリソース不要

「J-Scope」は JTAG 経由で必要な情報を取得します。そのため特別なハードウェアは不要、新たに情報取得のためのピンを立てる必要はありません。

高速なデータ取得

J-Link の RTT インターフェースを利用することで、最小転送周期：28.5 μ Sec (2-byte 変数 x8 (=16byte) の場合 / Cortex-M4) を実現します。

各種開発ツールと併用可能

お客様が利用している各種開発ツールと併用が可能です。



「J-Scope」はターゲットが動作している間、リアルタイムで RAM 上のデータを分析し可視化するためのソフトウェアです。複数の変数の値をオシロスコープのようなスタイルで表示できます。ELF ファイルを読み込み、視覚化する変数の数を選択することができます。

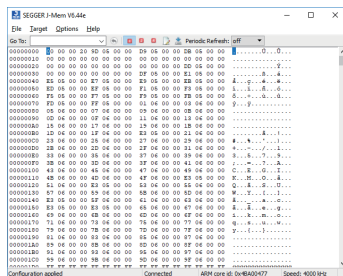
J-MEM

RAM データ表示、変更書込ツール

対応 CPU コア

すべての対応コア

- RAM 変更
- SFR 領域に書込
- RAM 領域全体に希望の値を入力
- メモリセクションを .bin ファイルに保存
- 表示されたメモリの内容の更新間隔を設定可能





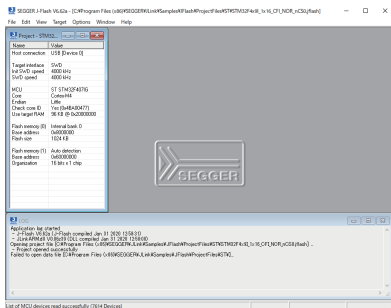
Flash 書込ツール

J-Flash / J-Flash SPI

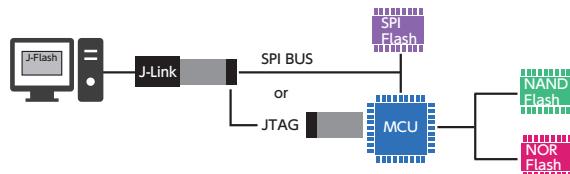
対応 CPU コア

対応する CPU コア

J-Link を利用して、内蔵/外部フラッシュに書込（J-Link PLUS 以上）



フラッシュ書込みについて、専用ツールでサポートします。マイコン内蔵、マイコン経由だけでなく、SPI バスに接続して直接 SPI フラッシュメモリに書き込み可能な「J-Flash SPI」も併せて提供。5,200 種以上のマイコン内蔵フラッシュ、300 種以上の SPI フラッシュに対応します。未対応のデバイス、専用 SoC などでも「Open Flash Loader」の仕組みを利用することで対応可能。当社で定義ファイルのカスタム対応も承ることが出来ます。



外部 NOR/SPI フラッシュ対応

JTAG からマイコン経由での書込みの他、対応する SPI フラッシュでは、SPI バスから直接書込む事が可能です。

マルチバンク対応

同じハードウェアに存在する異なるフラッシュを 1 回のセッションで書込むことができます。



⚠ J-Link BASE でも利用可能な「J-Flash LITE」

J-Flash LITE は、「J-Flash / J-Flash SPI」と書込アルゴリズムなどが異なりますので、サポート対象外・量産利用不可のソフトウェアとなります。



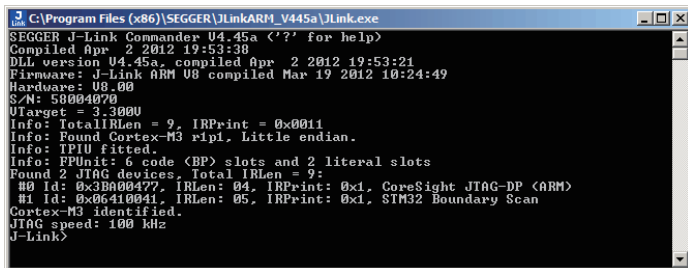
汎用ツール

J-Link Commander

対応 CPU コア

対応する CPU コア

ハードウェアデバッグ・不具合解析に利用できるコマンドラインツール



ハードウェア開発において、役立つツール「J-Link Commander」基本的なマイコンの接続確認やペリフェラルに変数を与えて、期待値通りの信号が出るかなど、ハードウェアデバッグに最適なツールです。ソフトウェア開発においてもマイコンへの接続、フラッシュへの書込など、様々な局面で開発をサポートします。J-Link の接続、動作がおかしいなどトラブルシューティングとしてもご利用頂けます。

J-Link Software 導入

J-Link を最大限に活用するバンドルソフトウェア



ダウンロード

EmbTeK www.embitek.co.jp へアクセスしてください。



J-Link 製品ページに移動

- ① トップメニュー [製品] を選択
- ② [JTAG/SWD デバッガツール] を選択
- ③ 任意の J-Link 製品を選択してください。

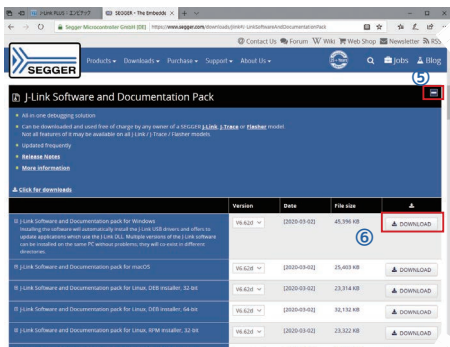
SEGGER 社ダウンロードページに移動

- ④ [J-Link 用ソフトウェア・ドライバ] をクリック



SEGGER 社 J-Link ソフトウェアダウンロードページ

- ⑤ SEGGER 社ダウンロードページで、「J-Link Software and Documentation Pack」のアイコン [+] をクリック

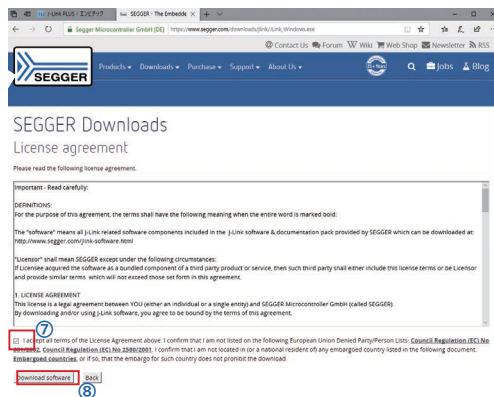


- ⑥ ご利用 OS (Windows / Mac / Linux) 用に合わせて [DOWNLOAD] ボタンをクリック

J-Link ソフトウェアダウンロード

- ⑦ 利用規約に同意する旨、チェックを入れてください。

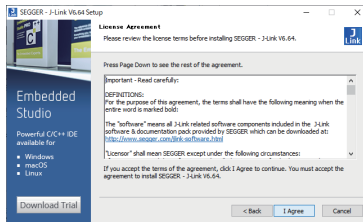
- ⑧ [Download Software] ボタンをクリック



インストール

J-Link ソフトウェアのインストール

ダウンロードが完了したら、インストーラの指示に従い、インストールしてください。



インストールフォルダ構成

インストールフォルダ

<SEGGER\JLink>

- | JFlash.exe フラッシュ書込ツール
- | JFlashLite.exe ノンサポート書込ツール（評価のみ）
- | JFlashSPI.exe SPI 書込ツール
- | JFlashSPL_CL.exe SPI 書込ツール（コマンドライン）
- | JLink.exe J-Link コマンドツール
- | JLinkConfig.exe J-Link 設定ツール
- | JLinkDLLUpdater.exe J-LinkDLL アップデータ
- | JLinkGDBServer.exe J-Link GDB サーバ
- | JLinkGDBServerCL.exe J-Link GDB サーバ（コマンドライン）
- | JLinkGUIServer.exe
- | JLinkLicenseManager.exe ライセンスマネージャ
- | JLinkRDIConfig.exe RDI コンフィグツール
- | JLinkRegistration.exe
- | JLinkRemoteServer.exe リモートデバッグサーバ
- | JLinkRemoteServerCL.exe リモートデバッグサーバ（C/L）
- | JLinkRTTClient.exe J-LinkRTT クライアント
- | JLinkRTTLogger.exe J-LinkRTT ロガー
- | JLinkRTTViewer.exe J-LinkRTT ビュー
- | JLinkSTM32.exe
- | JLinkSTR91x.exe
- | JLinkSWOViewer.exe SWO ビュー
- | JLinkSWOViewerCL.exe SWO ビュー（コマンドライン）
- | JMem.exe JMEM メモリビュー
- | JRun.exe
- | JTAGLoad.exe
- | SWOAnalyzer.exe

- | Devices デバイス定義ファイル
 - | --- AnalogDevices
 - | --- ATMEL
 - |

- | Doc マニュアル・リリースノート
 - | --- Manuals
 - | --- ReleaseNotes
- | ETC
 - | --- JFlash
 - | --- GDBServer
 - | --- RDDI

- | Samples すぐに使用可能なサンプルプロジェクト
 - | --- DCC
 - | --- GNU
 - | --- IAR
 - | --- GDB
 - | --- GDBInit
 - | --- Projects
 - | --- JFlash
 - | --- ProjectFiles J-Flash サンプルプロジェクト
 - | --- AnalogDevices
 - | --- ARM
 - | --- Atmel
 - |
 - | --- JFlashSPI
 - | --- ProjectFiles J-Flash SPI サンプルプロジェクト
 - | --- JLink
 - | --- Projects
 - | --- Scripts
 - | --- SettingsFiles
 - | --- RDI J-Flash SPI サンプルプロジェクト
 - | --- Projects
 - | --- IAR
 - | --- KEIL
 - | --- SetupFiles
 - | --- RTT
 - | --- USBDriver

最新版の「J-Link ソフトウェア」「J-Link ファームウェア」でのサポート対応となります。

サポートご依頼時については、最新版 J-Link ソフトウェアのダウンロード・インストールをお願いいたします。