ソラコム デバイス実装ガイドライン

はじめに

セルラー接続されたIoTデバイスの増加に伴い、モバイルネットワーク事業者（MNO）は、積極的に再接続を行ったり、不安定なデータ転送を行ったりするデバイスを警戒し、モバイルネットワークのパフォーマンスにますます注意を払うようになっています。

このような動作をするデバイスは、デバイスやアプリケーション自体にとって非効率であるだけでなく、セルラーネットワーク全体に過度のストレスを与えます。
MNOはネットワーク上の他のデバイスのセルラーサービスレベルの低下や通信障害を防ぐために、このような「適切でない」デバイスをブラックリストに登録したり、恒久的にブロックしたりする可能性があります。

上記の影響を受けないためには、効率的で安全で、使用するネットワークに害を与えないセルラーIoTデバイスを開発することが必要となり、そのためにはデバイス実装ガイドラインやベスト・プラクティスに従うことが重要です。

ご注意事項

本ガイドラインは、セルラー接続されるIoTデバイスを開発する際に、適切な実装を行うための注意事項とベストプラクティスを説明するものとなっており、主に GSMA TS.34 IoT デバイス接続効率ガイドラインの内容を参照しています。

本資料は弊社通信ネットワークにおいてIoTデバイスをご利用いただく際のガイドラインであり、IoTデバイスの機能や品質について保証するものではありません。本資料に記載の内容を実施したことにより、お客様又は第三者が損害を被ったとしても、ソラコムは一切の責任を負いません。また本資料の記載事項については、予告無く変更となる可能性があります。

全体的なガイドライン

一般的に、セルラー接続されるIoTデバイスやアプリケーションを設計する際には、以下を考慮する必要があります：

- 法令に適合した無線機器を利用する：無線機器は接続する各国の法令に準拠する必要があります。例えば日本においては電波法および電気通信事業法に基づく技術基準に適合する必要があります。通信モジュールは「工場設計認証（電波法第38条）」および「設計認証（電気通信事業法第56条）」を取得しているものを必ずご利用頂く必要があります。お客様製品（デバイス）においては、法令により製品または製品マニュアルに通信モジュールの技適マー

ク（直径3mm以上の大きさで表示すること）と認証番号を表記する必要があります。なお、各国に法令についてはお客様にて詳細をご確認ください。

- 標準仕様に従う実装を行う：IoTデバイスは、3GPPやGSMAなど、セルラーIoT接続の業界標準ガイドラインに従うことを推奨します。これにより、デバイスがネットワーク・インフラと適切に接続され、ネットワーク停止やその他の障害のリスクを軽減することができます。

- キャリア**IOT**（**Inter-Operability-Test**）取得の確認：利用するデバイス内のチップセット・モジュールは接続をする全ての通信キャリアのIOT取得済みであることが推奨されます。IOT未取得の場合は商用版筐体と弊社SIMを用いたフィールドテストにて、アンテナパフォーマンスや各種E2E(End to End)テストなどを十分に実施いただく事が重要となります。

- 効率的なアプリケーション・プロトコルを使用する：IoTデバイスは、MQTT、CoAP/LwM2M、HTTPなどの効率的なプロトコルを使用してデータを送信する必要があります。これらのプロトコルは、送信するデータ量を最小限に抑え、ネットワークの混雑を緩和し、ネットワークの過負荷のリスクを最小限に抑えるように設計されています。

- データ通信量を管理する：IoTデバイスは、接続・切断処理といったシグナリングを含めたデータ通信量を最小限に抑えるように設計することを推奨します。例えばeDRX/PSM等の低電力無線技術を使用したり、データ圧縮またはエンコード／デコードを行うことで、送信データ量を制限し、データ使用量を最適化することができます。

- 電力使用の最適化：電力最適化は、バッテリー駆動のIoTデバイスの寿命を延ばす鍵です。IoTデバイスは、用途や必要に応じて長時間データ転送が不要な場合はネットワークから切断するなど、電力を効率的に使用するように設計する必要があります。これにはネットワークへの影響を最小限に抑える効果もあります。

- 強固なセキュリティ対策の導入：IoTデバイスは、ネットワーク接続を悪用した不正アクセスや悪意のある攻撃を防ぐため、セキュリティを念頭に設計される必要があります。これには、強力な認証と暗号化対策の実装、セキュリティ・パッチとファームウェア・アップデートによるデバイスの定期的な更新が含まれます。

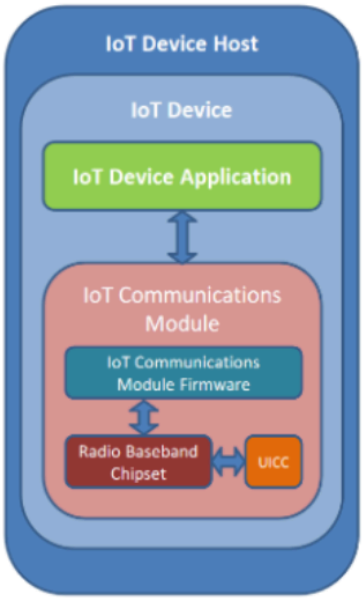
- テストと検証：IoTデバイスは標準仕様に準拠し、どのような状況に遭遇してもネットワークに害を与えないことを保証するため、導入前にさまざまな条件で徹底的にテストし、検証する必要があります。これには、相互運用性、互換性、理想的な条件と不利な条件の両方におけるパフォーマンスなど、あらゆるもののテストが含まれます。

- ネットワークの使用状況を監視する：IoTデバイスがネットワークに害を与えないように監視する必要があります。これは、ネットワーク接続頻度、データ使用量、その他のメトリクスを監視し、異常やトラヒックのスパイクを特定し、ネットワークの中断を防ぐために適切な措置を講じることにつながります。

- **FOTA**機能の統合：最後に、IoT デバイスはFirmware Over The Air (FOTA) メカニズムを使用してリモート・アップデートできるように設計される事を強く推奨します。セルラー接続されたIoTデバイスのすべてのコンポーネント（セルラーモジュール、SIM、アプリケーションを実行するマイクロコントローラー）は、FOTAを受信できる必要があります。いかなるソフトウェアもあらゆる状況で永久にエラーがないことを担保できることはありません。
  セルラーネットワークと標準仕様は継続的に進化しているため、セキュリティと正しい機能を確保し、意図せぬ通信断を防ぐためには、これらのコンポーネントのアップデートが必要になる場合があります。

デバイス・アーキテクチャ

セルラー接続された IoT デバイスを構築する方法は非常に多いため、すべてのデバイスとアプリケーションの設計要件に有効な特定の推奨事項を示すことは困難です。そのため GSMA TS.34 では、デバイス・アーキテクチャを以下のように一般化しています：



IoT Device Host -公共料金メーター、セキュリティアラームなどのIoTデバイスを含むアプリケーション固有の環境。
IoT Device - IoT Device ApplicationとIoT Communication Moduleの組み合わせ。
IoT Device Application - IoT Communication Moduleを制御し、それを介してIoTサービス・プラットフォームと相互作用する、IoTデバイスのアプリケーション・ソフトウェア・コンポーネント機能。
IoT Communication Module - 広域（セルラー）無線接続を提供する通信機器機能の総称。IoT Communication Module Firmware、Radio Baseband Chipset 、UICCで構成される。
IoT Communication Module Firmware - IoT Device ApplicationにAPIを提供し、Radio Baseband Chipsetを制御するIoT通信モジュール内の機能。
Radio Baseband Chipset -モバイル・ネットワークへの接続を提供する通信モジュール内の機能
UICC –モバイル・ネットワークが、モバイル・ネットワークへの接続やネットワーク・サービスへのアクセスのためにデバイスを認証するために使用するスマート・カード。

【図】標準的なIoTデバイスのアーキテクチャ

このような一般化されたデバイス・アーキテクチャに加えて、デバイスはしばしば次のような点でも区別されます：

- オペレーティング・システム（**OS**）駆動型デバイス
  - IoT通信モジュールは通常、オペレーティング・システム内でIoTデバイス・アプリケーションと並行して動作する専用の管理ソフトウェアによって制御されるデバイス。
- マイクロコントローラー駆動デバイス
  - IoT通信モジュールは通常、IoTデバイス・アプリケーションによって直接制御される（マイクロコントローラー独自のIPスタックまたはセルラー・モジュールに統合されたIPスタックを使用する）デバイス。
- 組み込みデバイス
  - IoTデバイス・アプリケーションが、IoT通信モジュール（SoCまたはシステムオンチップと呼ばれることもある）内の独立したアプリケーション・プロセッサ上で実行されるデバイス。

IoT通信モジュールは、セルラーネットワークへの接続に必要な3GPP標準のベースバンドスタックを処理します。しかし、IoTデバイス・アプリケーションはこれらのモジュールを制御するため、アプリ

ケーションがモジュールにセルラー・ネットワークに深刻な害を与えるような動作をさせる可能性があります。したがって、アプリケーションが適切に設計されていることが重要です。

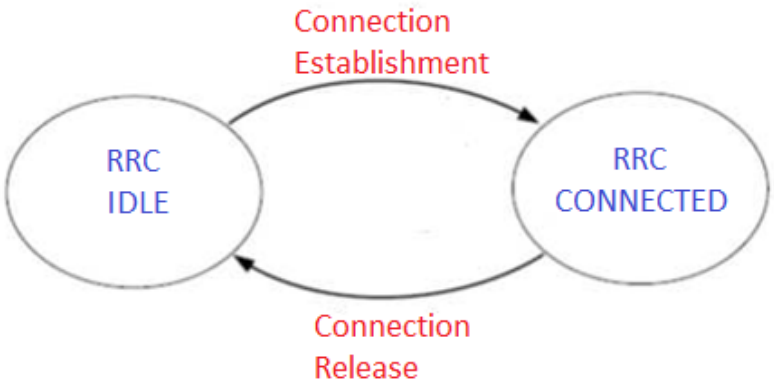
なお、セルラーネットワークに損害を与えるシナリオの中には、サイバーセキュリティ攻撃に由来するものもあるが、実際には、接続性を維持し、できるだけ速く簡単にデータを転送したいだけの機器に起因するものがほとんどです。

このような問題のあるデバイスが引き起こす最も一般的なシナリオは以下の通りです：\

- **IoT**デバイス・アプリケーションが、節電を達成するため、またはセルラーネットワーク接続をより迅速に回復するために、**IoT**通信モジュールの電源を過度にオン・オフすること。
- デバイスの再起動やデバイスのファームウェアの更新など、多数の**IoT**デバイスで同時に同じアクションを実行すると、ネットワーク接続やデータ転送が突然急増するため、セルラーネットワーク保護の観点から一種の「接続拒否」が発生する。

過度な接続や切断を避ける

過剰なオン／オフはGSMA TS.34_4.0_REQ_001に違反し、LTE標準の「常時オン、常時接続」のコンセプトに反します。LTEにおいて、IoT通信モジュールは2つのステータスしかありません：「RRC **CONNECTED**」と「RRC **IDLE**」です。



アイドルモード(RRC IDLE)では IoT デバイスの消費電力は少なくなりますが、それでも IoT 通信モジュールの電源が完全にオフの場合よりは消費電力が高くなります。しかし、LTE は、IoT 通信モジュールの電源を頻繁にオン・オフするようには設計されていません。モジュールの観点からは、オンとオフの切り替えプロセスは非常に迅速に行われますが、ネットワークの観点からは、信号メッセージが大幅に増加し、セルラーネットワークに大きな負担がかかります。

状況によっては、モジュールのスイッチを入れ直すプロセスは、アイドルモード(RRC IDLE)のまま放置するよりも、実際に多くの電力を消費する可能性があります。もちろん、オン/オフのスイッチング頻度によっては、電力を節約するためにIoT通信モジュールのスイッチを切ることが理にかなっている場合もあります。しかし、eDRX（Extended Discontinuous Reception）やPSM（Power Saving Mode）といった省電力ネットワーク機能（LTE-MやNB-IoTテクノロジー、最新のLTE cat1bisテクノロジーの機能）の導入により、IoT通信モジュールを完全にシャットダウンする必要性はさらに低くなりました。

代わりに、これらの省電力機能を有効にすることで、ネットワークに新たな負担をかけることなく、IoTデバイスの省電力化を実現することができます。

多数のデバイスの同期動作を避ける

TS.34_4.0_REQ_003および027で言及されているデバイスの同期動作は、MNOによってセルおよびコアネットワークのパフォーマンス指標を通じて検出されています。これらのは MNO にしか見えないため、特に、同じ基地局やネットワークに接続されているデバイスが数台しかない IoT デバイス開発のテスト段階では、このような問題を特定することは困難です。とはいえ、家畜のモニタリングや人口密集地でのスマートメーターなど、同期動作が起こりやすいユースケースは数多く存在します。

例えば、何百ものスマートメーターを一斉に起動する場合（例：午前4時）などは典型的なケースです。これらのメーターが同じデバイス・アプリケーション・ソフトウェアを実行している場合、同時に再接続しようとします。その結果基地局やネットワークには接続要求信号が殺到し、負荷が高まり新規接続が難しくなったり、輻輳状態になり新規接続ができなくなるなど、トラブルが長引く可能性があります。

このような事象への対処としてはネットワーク接続要求をずらすことが有効です。具体的にはデバイス・アプリケーションにランダム化タイマーを実装することが、ネットワーク接続をより長い期間にわたって分散させ、ネットワークに負荷をかけるリスクを低減するための最も価値ある設計上の考慮事項の1つです。

デバイスあたりのネットワーク接続数を減らす

TS.34_4.0_REQ_002に記載されているように、IoTデバイスとネットワーク間のネットワーク接続数を最小化することも目標の一つである。リアルタイムの追跡などのアプリケーションでは、位置情報の継続的な送信を可能にするために持続的なネットワーク接続が必要になるため、これはユースケースに大きく依存します。

しかし、非リアルタイムのユースケースの場合、開発者はアプリケーションが本当に10秒ごとに10バイトのデータを転送する必要があるのか、それとも代わりに60秒ごとに60バイトを転送することが可能なのかを検討する必要があります。このような設計上の決定は、IoTデバイスの消費電力を削減することによって、IoTデバイスにとっても、ネットワークにとっても相互に有益です。

データ転送サイズを小さくする

ネットワーク接続の最適化も、TS.34_4.0_REQ_015と密接に関連しています。
TS.34_4.0_REQ_015は、オーバーヘッドを最小限に抑え、転送するデータの総量を減らすために、データ圧縮方法を使用するよう開発者に要求しています。

開発者はまた、IoTデバイス・アプリケーションをよりインテリジェントに設計することができます。例えば、デバイスが一定間隔で8バイトのセンサーデータを送信する場合、IoTデバイス・アプリケーションは、同じデータを何度も送信するのではなく、前回の読み取りから変更された1～2バイトのセンサーデータのみを送信するように変更することができます。

エッジAIもまた、データ転送のサイズを削減できる分野です。新しいAIアルゴリズムは日々開発されており、デバイス上で直接データを処理し、アプリケーションデータのどの部分を伝送することが重要かを要約することがますます容易になっています。
【参考】バイナリパーサー：
この関数は、固定フォーマットのバイナリデータを解析し、JSONデータに変換するように設計されています。この JSON データは、Soracom Beam、Funnel、Funk、Harvest Data などの他の Soracom サービスに渡すことができます。バイナリパーサーは、デバイスがコンパクトなバイナリ形式でデータを送信できるようにすることで、データトラフィックと消費電力を削減するのに役立ちます。
データはJSON形式に変換されるため、人間の可読性を損なうことなく、クラウド側のデータの可視性を高めることができます。

アクティブ接続時間の短縮

転送するデータの頻度とサイズを減らすだけでなく、そのデータの転送にかかる時間を減らすことも同様に重要です。直感に反すると思われるかもしれませんが、IoTデバイスはより早くデータ転送を終了して省電力状態に戻ることができるため、LTEのような高速で電力消費の多い接続技術の方が、電力効率は良いが低速の技術よりも電力効率が良い場合があります。

もちろん、このことは、周波数、サイズ、継続時間にわたって損益分岐点が存在することを意味します。データサイズが小さくなるにつれて、IoTデバイスはLTE-MやNB-IoTなどのより電力効率の高い技術の恩恵を受けられる可能性があります。一方、データ転送が頻繁に行われる場合、eDRXやPSMなどの機能はそれほど効果的でない可能性があり、データサイズを大きくして転送頻度を下げると、より高速な技術が有利になる場合があります。

さまざまな消費電力テストを実施することは、伝送周波数、データサイズ、および接続時間の最適な組み合わせを選択するために不可欠です。

プロトコルを適切に選択する

多くのユースケースにおいて、開発者は、IoTデバイスが正常に動作していることを示す簡単なメッセージを定期的に送信するハートビート・メカニズムを実装する場合があります。このようなメカニズムを実装する場合、基礎となる通信プロトコルに従って実装を計画することが重要です。

例えば、TCPのようなコネクション指向のプロトコルを使用する場合、ハートビート間隔は、プロトコルのコネクションタイムアウトの直前に設定すべきです。ハートビートが停止しても、デバイスがオフラインであることを自動的に示すわけではないので、ハートビートを速く設定することは、過剰なメッセージを生成し、ネットワークのリソースを浪費する事につながります。実際は接続がタイムアウトして初めて、デバイスがオフラインであると判断できます。

これは、データが転送されていないときでも接続をオープンにしておくために TCP キープアライブを使用する場合にも適用されます。ここで、TS.34_4.0_REQ_007 は、TCP キープアライブを使用するデバイスのデフォルトのポーリング間隔を 29 分と定義しています。

プロトコルの選択は、どの接続技術を使いたいかにも大きく左右されます。最もポピュラーな例はNB-IoTで、設計上、電力効率を非常に高めるためにレイテンシーが非常に高く(概ね30秒以上)、広いカバレッジ・エリアを実現するためにデータ伝送を自動的に複数回繰り返します(これはカバレッジ・エンハンスメントと呼ばれます)。このような設計のため、TCPのようなコネクション指向のプロトコル(MQTTやHTTPのようなTCPベースのプロトコルも同様)は、タイムアウトやリトライのメカニズムがNB-IoTの待ち時間やメッセージの繰り返し動作と衝突し、何百、何千もの不要なメッセージが発生する可能性があるため、誤った選択となります。TS.34_4.0_REQ_006および031で提案されているように、NB-IoT上で通信するIoTデバイスはTCPを一切使用すべきではありません。

無線アクセス技術の動作を理解する

さらに踏み込んで、異なる接続技術間でネットワーク検索時間がどのように異なるかを知ることは、IoTデバイスを設計する際にも重要です。例えば、IoT通信モジュールがNB-IoTネットワークを見つけるのにかかる時間は、使用する帯域の大きさに応じてほぼ直線的に増加します。10MHz帯(Band 13など)のスキャンには約4秒かかるが、60MHz帯(Band 1やBand 2など)のスキャンには6倍の時間がかかります。したがって、NB-IoTバンド1、2、13を有効にしたIoT通信モジュールは、ネットワークに接続するまでにおよそ50秒間スキャンする必要があります。

これは、IoT通信モジュールが通常2〜3分以内に完全なネットワーク検索を完了し、ネットワークに接続できる2G、3G、およびLTEテクノロジーと比較しています。モジュールが3分以内に接続されることを想定し、そうでない場合は再起動するように設計されているIoTデバイス・アプリケーションでは、NB-IoTネットワークのスキャンを正常に終了できない可能性があります。

このため、NB-IoTは常設の場所に設置される機器に適していると説明されています。モバイルユースケースの機器が移動する際のように、常にネットワークを探す必要がないというのがその主な理由です。

多くの LPWAN IoT 通信モジュールが LTE-M、NB-IoT、および 2G 技術をサポートしているため、TS.34_4.0_REQ_008 および 009 で推奨されているように、接続技術に応じてタイムアウトを調整し、使用するバンド数を制限し、技術固有の動作(NB-IoT ネットワークの遅延やカバレッジ拡張など)にアプリケーションを適応させる IoT デバイス・アプリケーションを開発することが不可欠です。

通信モジュールの動作を理解する

さまざまな接続技術がどのように動作するかを理解し、IoTデバイス・アプリケーションをIoT通信モジュールとどのように連携させ、それに応じて適応させるかを設計することは重要ですが、アプリケーションは、モジュールに対して手動による制御を過度に課さないように設計する必要もあります。

例えば、すべての IoT 通信モジュールには AT+COPS コマンドが含まれており、このコマンドを使用して、モジュールに特定のネットワークへの接続を指示することができます。IoTデバイス・アプリケーションはこのコマンドを活用してネットワーク接続手順を大幅に高速化できますが、このコマンドはモジュールのデフォルト機能である、より強力な信号と優れたパフォーマンスを持つ可能性のある別のネットワークを検索して切り替える機能も無効にします。

なお、ソラコムではT&Cで特定のPLMN（通信キャリア）への恒久的な接続を担保していません。そのためグローバルローミングSIMを使用する場合、デフォルトのAT+COPS自動モードを設定することで在圏国内でのキャリアを指定しないことをソラコムでは強く推奨しています（そうしない場合、将来的にセルラーネットワークへの接続を失うリスクをお客様が抱えることとなります）。
また、IoTデバイスがATコマンドなどで事前に接続ネットワークを選択した際、選択したネットワークの信号が低い地域にある場合、IoT通信モジュールがこのネットワークに接続することを余儀なくされると、IoTデバイス・アプリケーションのデータ送信能力が損なわれるだけでなく、通信障害によってネットワークに負荷が掛かる危険性があります。

一部のIoT通信モジュールは、AT+COPSコマンドのハイブリッドバージョンをサポートしており、接続する特定のネットワークを手動で指定できますが、特定のネットワークが利用できないか、適切でない場合は、自動的に通常のネットワーク検索プロセスにフォールバックします。

IoT通信モジュールの機能と動作を知ることは、IoTデバイスがグローバルに展開されるユースケースではさらに重要になります。IoTデバイスは、より広範な接続技術、利用可能な周波数帯域、各デバイスが配置されているネットワークのスキャン条件などに対処しなければならないからです。

データ通信の失敗に正しく対処する

IoTデバイスがネットワーク通信リクエストに失敗する状況に遭遇することは避けられません。このような障害への対処方法を検討する場合、IoT デバイス・アプリケーションは、IoT 通信モジュールの通常の動作がセルラー接続を維持することであることを理解した上で設計する必要があります。TS.34_4.0_REQ_011, 012, 019, 029 で推奨されているように、開発者は IoT デバイス・アプリケーションが通信障害をどのように処理するかに注意する必要があります。

IoT デバイスアプリケーションは、失敗した通信要求を無期限に再試行すべきではありません。同様に、頻繁な再起動や再接続は避けるべきです。通信障害は、IoT デバイス・アプリケーションの別の部分（例えば、アプリケーションが DNS を解決できない、またはサーバーが一時的にオフラインであるなど）が原因である可能性があり、IoT 通信モジュールに再起動を強制することは不要であり、実施すればネットワークにさらなる負担をかけることになります。

その代わりに、デバイスは、IoT通信モジュールの手動操作に頼る前に、まず他の場所で問題をチェックし、TS.34_4.0_REQ_016、018、025で推奨されているように、データを内部にバッファリングまたは保存したり、リトライの間に指数関数的にバックオフしたり、あるいはネットワークがそれほどビジーでない「オフピーク」の時間帯や日までデータ送信を待ったりする実装を採用すべきです。

これは、他の種類の無線技術を組み込んだユースケースにも当てはまります。例えば、GNSSは位置追跡アプリケーションの一般的な選択肢であり、多くのIoT通信モジュールもセルラー接続に使用されるものと同じ受信機を使用してGNSSをサポートしています。しかし、TS.34_4.0_REQ_034に記載されているように、GNSS信号の損失がモジュールのセルラー部分の再起動につながることがあってはならず、IoTデバイス・アプリケーションは、セルラー・モジュールを再起動せずにGNSS信号を評価するために受信機を解放できるように、eDRXのような技術を活用する必要があります。

同じことが、LAN（TS.34_4.0_REQ_036）、WiFi（037）、または接続されたセンサー（038）のような他の接続を持つ IoT デバイスにも適用されます。各通信は、IoT デバイス・アプリケーションによって個別に処理され、IoT 通信モジュールも再起動することなく再起動可能でなければなりません。

最悪のシナリオでは、IoTデバイスがこうした配慮を怠り、IoT通信モジュールの標準的な動作を上書きしてしまった場合、MNOはネットワークへのさらなる被害を防ぐために、そのデバイスを全面的にブラックリストに登録する可能性があります。ブラックリストの種類によっては、モジュールまたはデバイス全体を交換する以外に回復する方法がない場合もあるため、開発と実装には細心の注意を払う必要があります。

active状態以外のSIMの接続を行わない

利用中断中（susupended）および解約済み（terminated）のステータスのSIMについてはネットワークへの接続が許容されません。このような状態でお客様製品が通信を試行した場合、ネットワークとの間で不要な制御信号のやり取りが発生し、これが繰り返される場合はネットワーク全体の負荷の増加につながります。そのため、利用中断中（**susupended**）および解約済（**terminated**）ステータスの**SIM**については**Host**側から接続要求を繰り返さないように製品側で実装をしてください。
デバイスの接続に失敗する場合は、原因コードから失敗理由を確認し、それがSIMのステータスに起因していないかを確認してください。
また、休止中（inactive）は、一時的にデバイスからのデータ通信を禁止するSIMステータスですが、長期間休止中（inactive）ステータスを維持することは、デバイスとネットワーク間の不要な制御信号を増大させますので、長期間利用予定がないSIMについては休止中（inactive）ではなく、利用中断中（suspended）を利用しください。

【参考】SIMのステータス・ユーザーコンソールでの状態表示について
ソラコムでは、SIMの状態に関して利用可能な特別な機能があります。
以下では、これらのSIMの状態について詳しく説明します。

inactive:
サブスクリプションが非アクティブに設定されている場合、データセッションはブロックされます。このステータスは、ネットワーク接続が必要ないときに、デバイスがデータを消費するのを一時的に防ぐために使用できます。
そのため、この状態ではデータ使用料は発生しないが、月額基本料金が発生します。
非アクティブ SIM のステータスは、一時的に使用することを推奨し、IoT デバイスを長期間使用する予定がない場合は、このステータスを恒久的に維持することは推奨しません。

データセッションはブロックされますが、デバイスは継続的に再接続を試みるため、過剰な電力消費が発生する可能性があります。そのため再接続の試行間隔を指数関数的に長くする、あるいは、設定された試行回数後に再接続のタイムアウトを実装するなど、適切な再接続プロセスを推奨します。

suspended:
データセッションはブロックされますが、デバイスは継続的に再接続を試みるため、過剰な電力消費が発生する可能性があります。
再接続の試行間隔を指数関数的に長くする、あるいは、設定された試行回数後に再接続のタイムアウトを実装するなど、適切な再接続プロセスを推奨します。
standby状態とは異なり、すべてのデータセッションがブロックされるため、SIMを使用するデバイスが再びネットワークに接続できるようになるには、手動でサブスクリプションを再active化する必要があります。

重要：サブスクリプションがsuspendedされている間にデバイスがネットワークに接続しようとすると、ソラコムは接続が永久に拒否されたことを通知する信号をデバイスに送信します。手動でSuspendedサブスクリプションを再active化した後、ネットワーク接続を再開するには、デバイスも手動で再起動またはパワーサイクルを行う必要があります。サブスクリプションをSuspendedステータスに変更する前に、サブスクリプションを再active化する際にデバイスを再起動するためにそのデバイスにアクセスできることを確認してください。

デバイスのシャットダウン/オフ：
同様に、サブスクリプションが現在オンラインに表示されている状態で、デバイスがネットワークから突然切断された場合（デバイスのバッテリーや電源入力を外すなど）、デバイスはデータセッションの終了をネットワークに通知する機会がありません。その結果、サブスクリプションは、デバイスの電源が切れた後、最大1時間までオンラインに表示され続ける可能性があります。

デバイス回復メカニズム

携帯電話ネットワークの一時的な問題など、いくつかの理由により、セルラーモジュールが特定のMNOネットワークを禁止PLMNリストに追加することがあります。これは、SIMカード上のファイルに過ぎません。このようなエントリーは、モバイルネットワークから特定の拒否理由がある場合にのみ書き込まれます。これは稀なシナリオであるにせよ、回復シナリオが欠けているために接続性の損失といった長期にわたる影響を及ぼす可能性があります。3GPP標準の Rel.10以降では、自動FPLMNクリアリングの手順が定義され始めています。Rel.9以下をサポートするセルラーモジュールは、リストに記載されたMNOへの再アクセスを得るためにリストを自動的にクリアする能力がないかもしれませんことに留意すべきです。

実装された回復メカニズムによっては、長期間（例えば2日間）接続がない場合や、定義されたボタンを介してトリガーされるリセット手順内で、FPLMNリストを積極的にクリアすることが望ましいでしょう。なぜなら、接続が完全に失われた場合には、OTAからリセット手順をトリガーすることは機能しないからです。
接続できるMNOネットワークが1つしかないエリアで、セルラーネットワークの問題でロックアウトされたIoTデバイスなどはその典型的なケースとなります。このケースではセルラーモジュールがこの

MNO IDをFPLMNリストに書き込んだ可能性があります。これは、許可されていないネットワークへの接続要求の量を減らすための標準的な振る舞いです。
この拒否が理由によるものであれば、ネットワークの負荷を大幅に減少させ、デバイスのリソースを節約します。このエントリーをクリアするリセット手順の存在は、デバイスを再接続するために不可欠です。

サブスクリプションコンテナ (追加サブスクリプション) の利用

OTA（Over The Air）：
OTAアップデートは、SIMカードとサブスクリプションコンテナの管理にも関連しています。Soracom Airのサブスクリプションコンテナは、ユーザーが物理的にSIMカードを交換することなく、ネットワークサービスのサブスクリプションを変更することを可能にしています。SoracomのOTAプロセスでは、新しいサブスクリプションコンテナをSIMカードに送信し、それをアクティベートしてネットワーク、サブスクリプションの詳細を変更することができます。SIMカードにアクセスできるようにするには、上記の到達可能性のセクションで述べたように、デバイスの適切なオンライン時間（サブスクリプションコンテナによるダウンロードの場合、最低5分程度を推奨）を保証する必要があります。シャットダウンが早すぎるデバイスは、必要なデータを SIM に転送するための適切な時間をまったく提供しないか、デバイスがネットワークに再び接続されるとすぐにモバイル端末の終了データが継続されるため、OTA に数日かかる可能性があります。

SORACOM

最終更新日：2024/7/04

IoTデバイス実装チェックリスト

カテゴリ	詳細	推奨の実装
技適への対応 ※日本での利用の場合	IoTデバイスは技適を取得しているか	技適取得が必須
標準仕様への対応	IoTデバイスは3GPP等の標準仕様に沿ったデバイスになっているか	3GPP,GSMAの標準的な仕様に準拠している事を強く推奨
相互接続試験（IOT）への対応	IoTデバイス、または通信モジュールは利用するキャリアのIOTを行っているか	IoTデバイスや通信モジュールは利用するMNOネットワークのキャリアIOTが行われているものを選ぶことを強く推奨
【接続・通信の分散】 ・過度な接続や切断を避ける ・多数のデバイスの同期動作を避ける ・デバイスあたりのネットワーク接続数を減らす ・データ転送サイズを小さくする ・プロトコルを適切に選択する	・通常運用時の通信時間分散が可能か ・同時再接続を想定した接続分散が可能か	・特定時間帯にデータ送信を行う場合は通信時間を分散する実装を行う ・多数のデバイスが同時に接続を試行する場合を想定し、それを分散する仕組みを実装する
【通信容量・接続回数の最小化】 ・過度な接続や切断を避ける ・多数のデバイスの同期動作を避ける ・デバイスあたりのネットワーク接続数を減らす ・データ転送サイズを小さくする ・プロトコルを適切に選択する	・デバイスのネットワーク接続回数は必要最小限か ・データ転送サイズは適切か ・プロトコルを適切に選択しているか	・不要な接続（再接続）を行わない ・要件に応じてデータ転送回数、通信容量を必要最小限にする ・ハートビートはコネクションタイムアウトの直前に行う
【接続時間の考慮】 無線アクセス技術の動作を理解する	ネットワーク検索時間を考慮しているか	ネットワーク検索時間は最大で2～3分程度かかる事があるという前提でサービス、実装を検討する（検索中に通信不可扱いで再起動を行うなどの実装を行わない）
通信モジュールの動作を理解する	・ATコマンドによるネットワーク固定がされていないか ・IoT SIMが提供する通信キャリアについては全て対応しているか	・AT+COPS自動モードを選択する（特定キャリアのみを指定する実装をしない） ・複数のネットワークに対応するIoT SIMを選択する際はそのすべてに対応したIoTデバイスの選択を行う（最低でも利用国内で2つのネットワークに対応する通信モジュール、IoTデバイスを選択、実装する）
【接続通信失敗時の挙動】 データ通信の失敗に正しく対処する	・失敗した通信要求を無期限に再試行しないか	・通信失敗時に無制限に再通信を行わない実装を行う（通信失敗時間に応じて、指数関数的に接続回数を減らす仕組を実装する）
【FOTAへの対応】 FOTA機能の統合	IoTデバイスおよび通信モジュールのファームウェア更新が行えるか	IoTデバイスに利用ネットワーク経由でのFOTA機能を実装する

【休止・解約SIMの接続停止】 active状態以外のSIMの接続を行わない	activeステータス以外のSIMでは通信を試行しないか	利用中断中（susupended）および解約済（terminated）ステータスのSIMについてはHost側から接続要求を繰り返さないように製品側で実装をする。 ※機器の電源を切るなどの対応でも可能
【電源OFF時の挙動】 その他	IoTデバイスの電源をOFFにする場合は通信ネットワークから切断を行なうか	IoTデバイスのシャットダウン時にネットワークから切断する挙動を実装する。 （少なくとも通常運用において電源をOFFにする際には接続を切断するシーケンスを実装する）
【デバイスの再起動】 その他	データ通信の失敗時やIoTデバイスの挙動が正しくない場合に再起動による復旧を可能となっているか	IoTデバイスの正常性が確認できない際にIoTデバイス自身が再起動を行う仕組みを実装する。