LoRaWAN

LoRaWAN

バージョン 1.0

低消費電力広域（LPWA）ネットワーク・プロトコルのプレゼンテーションへようこそ

LoRaWANの概要

- LoRa® テクノロジー:
  - 欧州では868MHz、北米では915MHzなど、ライセンスを必要としない1GHz未満の周波数帯を使用
  - 長距離伝送（10km超）
  - 低消費電力
- LoRaWAN®プロトコルの特徴
  - 双方向
  - 単方向または半二重（必要に応じて全二重）
  - 変調LoRa（チャープ・スペクトラム拡散）およびFSK

LoRaWANプロトコルには、無線周波数の使用に関して正式な地域仕様が存在（周波数範囲、デフォルトのパラメータ、制限事項など）

life.augmented

LoRa® テクノロジーは、多数のオブジェクトをいくつかの広帯域ネットワークに接続するためのモノのインターネットのソリューションです。
このテクノロジーは、インフラストラクチャ製品を作成するためのコンプライアンスと能力を確保するオープンなグローバル標準とされています。

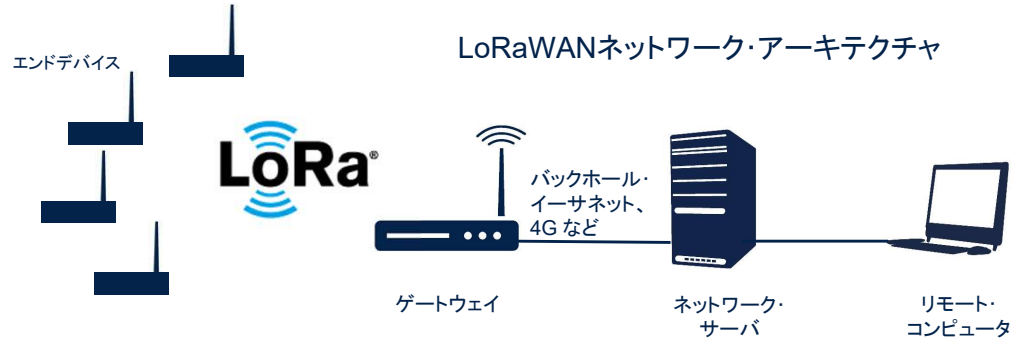
ネットワーク（1/2）

- LoRaテクノロジーは広域ネットワーク機能を提供できるので、**LoRaWAN**と呼ばれることが多い
- LoRaWANネットワークの構成要素：
  - **エンドデバイス（ノードとも呼ばれる）**：エンドデバイスは、センシングや制御を実行するLoRaWANネットワークの構成要素　通常はリモートに配置
  - **ゲートウェイ**：ゲートウェイは、LoRaWANエンドデバイスから通信を受信し、それをバックホール・システムに転送　LoRaWANネットワークのこの部分には、イーサネットやセルラーをはじめとするあらゆる有線または無線の通信リンクを使用可能　ゲートウェイは、標準的なIP接続を使用してネットワーク・サーバに接続される　データは標準プロトコルを使用するが、パブリックであるかプライベートであるかを問わずあらゆる通信ネットワークに接続できる　LoRaWANネットワークとセルラー・ネットワークとの類似性を考慮して、LoRaWANゲートウェイはセルラー基地局と共同設置されることが多い　その場合はバックホール・ネットワークの空いている容量を使用できる
  - **サーバ**：LoRaWANネットワークサーバは、ネットワークを管理する　ネットワーク・サーバは、重複パケットの排除、確認応答のスケジュール設定、データレートの調整を担当する　サーバの配備と接続の容易さを考えると、LoRaWANネットワークの導入は非常に容易である
  - **リモート・コンピュータ**：リモート・コンピュータは、エンドデバイスの動作制御やエンドデバイスからのデータ収集が可能　したがって、LoRaWANネットワークはほとんど透過的である

life.augmented

専用ネットワークや公共ネットワークを通じてセキュアで認証された方法でセンサ・データの提供やデバイスの制御を実現するために、各エンドデバイスは情報を送受信できます。

ネットワーク(2/2)



- LoRaWANネットワークの実際のアーキテクチャという観点では、エンドデバイスがスター型トポロジを構成し、ゲートウェイが透過的なブリッジを形成するのが一般的である　これらは、バックエンドの中央ネットワーク·サーバとエンドデバイスの間でメッセージを中継する
- エンドデバイスへの通信は、通常は双方向だが、マルチキャスト運用をサポートすることもでき、これはソフトウェア·アップグレードの類いや他の大量配信メッセージなどの機能に役立つ

life.augmented

4

多くの場合、LoRaWAN ネットワークはスター型トポロジで配置されます。このトポロジでは、エンドデバイスと中央ネットワーク·サーバ(バックエンドに設置)間のメッセージをゲートウェイが中継します。ゲートウェイは標準の IP 接続を通じてネットワーク·サーバに接続し、エンドデバイスはシングルホップの LoRa™ 通信または FSK 通信を使用して 1 台または多数のゲートウェイに接続します。このネットワークでは、一般的にすべての通信は双方向ですが、支配的と想定されるトラフィックは、エンドデバイスからネットワーク·サーバへのアップリンク通信です。
エンドデバイスとゲートウェイとの通信は、さまざまな周波数チャネルとデータ·レートにわたっています。

地域の概要

- 次の10の地域を定義:
  - EU868（863 ~ 870MHz）*
  - US915（902 ~ 928MHz）*
  - CN779（779 ~ 787MHz）
  - EU433（433 ~ 434MHz）
  - AU915（915 ~ 928MHz）
  - CN470（470 ~ 510MHz）
  - AS923（915 ~ 928MHz）*
  - KR920（920 ~ 923MHz）*
  - IN865（865 ~ 867MHz）*
  - RU864（864 ~ 870MHz）
- （*）LoRaWAN Certification^{CM} プログラムは、あらゆる LoRaWANネットワークでアプリケーション固有のエンドデバイスが動作することを検証するために、5つの地域の地域テストを用意
- 各地域で次の措置ができる
  - 認められた定義済みサブバンドに適合した形態で、ネットワーク・オペレータがネットワーク・チャネルを自由に使用可能
  - 物理的なビットレート（データ・レート）、Tx出力範囲、デフォルト設定を定義
  - いくつかの制限事項を規定可能（デューティ・サイクル、Dwell Time、LBT）

life.augmented

5

LoRaWAN プロトコルでは、主要な 10 地域を定義しています。世界中のさまざまな規制地域向けに、デフォルトの周波数チャネル、Tx 出力、データ・レート範囲などのデフォルト設定が用意されています。
国ごとに、統治区域ごとに効力がある規則どうしの整合性を確保するために 1 つ以上のチャネル・プランが規定されています。

LoRaWANプロトコル

LoRaWANの各クラス

- **クラス A - 双方向のエンドデバイス**: LoRaWANクラスAは双方向通信を提供 その通信を実現するために、各エンドデバイスからの送信の後、2つの短いダウンリンク受信ウィンドウが続く エンドデバイスでスケジュール設定される送信スロットは、ランダムな時間ベースによるわずかな変動を伴いながら、エンドデバイスのニーズに基づいて決まる LoRaクラスAの動作には、アップリンク送信の直後にのみサーバからのダウンリンク通信を必要とするエンドデバイス向けに、最小消費電力オプションが用意されている それ以外の時間におけるサーバからのダウンリンク通信は、スケジュール設定された次のアップリンク時間まで待機状態になる
- **クラス B - 受信スロットがスケジュールされた双方向エンドデバイス**: LoRaWANクラスBは、クラスAの機能のほか、スケジュール設定した時間に受信ウィンドウを別途開く機能を提供する ネットワークとの必要な同期をとるために、エンドデバイスは時間同期されたビーコンをゲートウェイから受信する これにより、サーバはエンドデバイスが受信するタイミングを知ることができる
- **クラス C - 最大の受信スロットを持つ双方向エンドデバイス**: LoRaWANクラスCは、ほぼ継続的に開いている受信ウィンドウを提供する エンドデバイスからの送信時にのみ、この受信ウィンドウが閉じる このタイプのエンドデバイスは、送信データよりも多い量のデータを受信する必要がある場合に適している

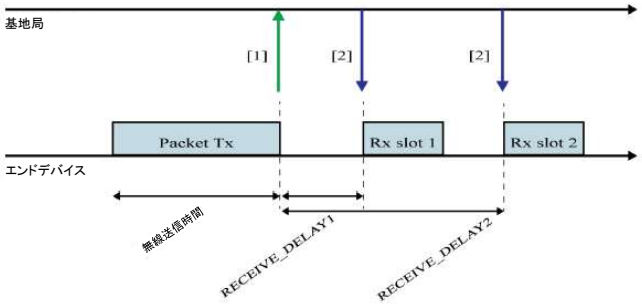
life.augmented

LoRaWAN ネットワークは、基本的な LoRaWAN（クラス A）とオプション機能（クラス B とクラス C）に分類できます。

- クラス A：クラス A のエンドデバイスでは、各エンドデバイスのアップリンク送信の後に、2 つの短いダウンリンク受信ウィンドウが続く双方向通信が可能です。
- クラス B：クラス B のエンドデバイスでは、別途スケジュール設定した受信ウィンドウによって、より多くの受信スロットを使用できます。
- クラス C：クラス C のエンドデバイスでは、ほぼ継続的に開いている受信ウィンドウを使用できます。

LoRaWANプロトコル
クラス Aの管理

- クラスA: エンドデバイスからメッセージ(センサ·データや MAC コマンドなど)を送信する場合は、1台以上のゲートウェイを介して、そのメッセージがアップリンクでネットワーク·サーバに送信される
  1台のゲートウェイから、ネットワークはダウンリンクのRx1またはRx2受信ウィンドウで応答できる(確認応答メッセージ、アクチュエータのデータ、MACコマンドなど)
  エンドデバイスは、Rx1でダウンリンク·フレームを受信した場合、Rx2を開放しない



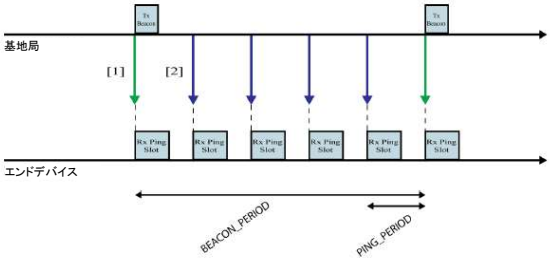
life.augmented

7

クラス A でエンドデバイスからメッセージ(センサ·データや MAC コマンドなど)を送信する場合は、1 台以上のゲートウェイを介して、そのメッセージがアップリンクでネットワーク·サーバに送信されます。
使用しているゲートウェイが 1 台であれば、ネットワークはダウンリンク Rx1 または Rx2 の受信ウィンドウで応答できます(確認応答メッセージ、アクチュエータのデータ、MAC コマンドなど)。
エンドデバイスは、Rx1 でダウンリンク·フレームを受信した場合、Rx2 を開放しません。

LoRaWANプロトコル
クラスBの管理

- クラス B: 同期した受信ウィンドウをエンドデバイスに追加
  LoRaWANエンドデバイスは、必ずクラスAでJoinし、アプリケーション層からの要求に応じてクラスBに切り替わることができる



life.augmented

8

クラス B では、同期した受信ウィンドウがエンドデバイスによって追加される。
LoRaWAN エンドデバイスは、必ずクラス A で Join し、アプリケーション層からの要求に応じてクラス B に切り替わることができます。

LoRaWANプロトコル
クラスBの管理

- クラスBエンドデバイスの同期原理
  - エンドデバイスは必ずクラスAのエンドデバイス（クラスB対応またはクラスB無効）として起動してネットワークにJoin
  - クラスBを有効にする要求は、必ずエンドデバイスのアプリケーション層から発行される
  - クラスB対応への切り替え
    - エンドデバイスはビーコンを探索してそれにロックする
      - ビーコン検出時間を短縮するために、DeviceTimeReq MACコマンドを使用可能
    - ビーコンにロックしたエンドデバイスは、受信ウィンドウ（pingスロット）を開く
      - クラスBで動作するには、pingスロット情報（周期性、データ・レート、周波数）をネットワークに公開する必要がある
    - クラスBモードになると、MACレイヤは、送信する各アップリンク・フレームのFCtrlフィールドで"Class B"のビットを1に設定する必要がある

life.augmented

クラス B 動作を実装したエンドデバイスは、固定の時間間隔で受信ウィンドウを開く必要があります。
クラス A からクラス B に切り替えるかどうかは、エンドデバイスのアプリケーション層で判断されます。
同じタイムスタンプでエンドデバイスとゲートウェイが同期した状態にするには、ビーコン・フレームが使用されます。クラス B に切り替えるには、デバイスからビーコンを 1 回以上送信する必要があります。

LoRaWANプロトコル
クラスBの管理

- クラスBモードでの動作:
  - サーバでは、PingSlotChannelReq MACコマンドを使用して、エンドデバイスのpingスロットのダウンリンクに使用する周波数またはデータ・レートを変更できる
  - エンドデバイスは、PingSlotInfoReq MACコマンドにより、そのpingスロットの周期性を変更できる
  - エンドデバイスからクラスAのアップリンク・メッセージをTx、Rx1、またはRx2の時間フレームで送信する場合は、クラスA設定が優先される　この時間フレームの間にpingスロット・ウィンドウでサーバから送信されたメッセージはすべて失われる
- クラスBのメッセージは、ユニキャスト(単一のエンドデバイスへの送信)とすることも、マルチキャスト(複数のエンドデバイスへの送信)とすることも可能
  異なるパラメータを使用してこの2つの方法を同時に有効にした場合は、pingスロット・ウィンドウの設定にどちらを使用するかをアプリケーション層で判断する必要がある

life.augmented

クラス B の設定は、ping スロットの周波数やデータ・レートなどのデフォルト設定を使用して、地域パラメータで定義します。これらの値はすべて、ネットワーク・サーバからの MAC コマンド命令によって更新できます。
また、クラス B の ping スロットはユニキャスト・ウィンドウとして使用できるほか、ネットワーク・サーバで定義されているオプションのマルチキャスト設定を使用して、マルチキャスト・ウィンドウとして使用することもできます。

LoRaWANプロトコル
クラスBの管理

- ビーコン・ウィンドウはクラスBスロットに使用できる時間間隔

BEACON_PERIOD	128秒
BEACON_RESERVED	2.120秒
BEACON_GUARD	3.000秒
BEACON_WINDOW	122.88秒

- すべてのビーコン・ウィンドウ期間のpingスロット・タイミング計算

K	0	1	2	3	4	5	6	7
ping 番号（= 2^{K}）	1	2	4	8	16	32	64	128
ping 期間（= $2^{(12-K)}$）	4096	2048	1024	512	256	128	64	32
割込み期間（秒）	122.88	61.44	30.72	15.36	7.68	3.82	1.92	0.96

- 衝突を回避するために、ビーコン周期ごとにスロット・インデックス（pingOffset）がランダムに変更される

最初のスロット	BEACON_RESERVED + PingOffset x slotLen（30 ms）
スロット 2	BEACON_RESERVED +（PingOffset + PingPeriod）x slotLen（30 ms）
スロット 3	BEACON_RESERVED +（PingOffset +（2 x PingPeriod））x slotLen（30 ms）
……	……

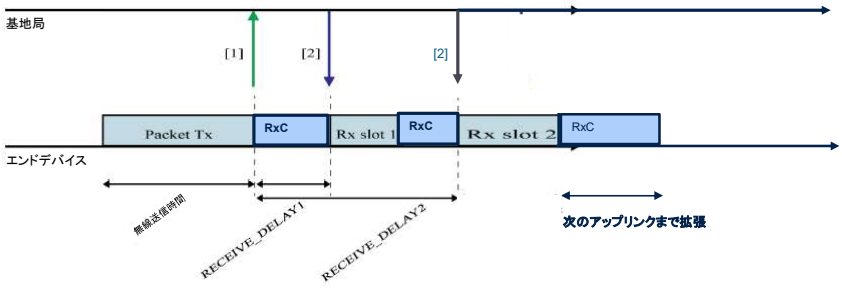
life.augmented

11

ゲートウェイで生成されたビーコンは 128 秒ごとに送信されます。クラス B モードが有効なすべてのエンドデバイスは、リッスン・ウィンドウを開いてこのフレームを受信する必要があります。エンドデバイスとネットワーク・サーバ間のクラス B 設定の定義に従い、1 秒に 1 回または 128 秒に 1 回開くように ping スロット・ウィンドウが設定されます。
また、ping スロットの計算では、擬似ランダム・オフセットを使用して受信ウィンドウを調整し、複数のエンドデバイスで衝突が発生しないようにしています。

LoRaWANプロトコル
クラスCの管理

- クラスC：同期した連続的な受信ウィンドウをエンドデバイスに追加
  LoRaWAN エンドデバイスは、必ずクラスAでJoinし、アプリケーション層からの要求に応じてクラスCに切り替わることができる



life.augmented

12

クラス C エンドデバイスには、同期した連続的な受信ウィンドウが追加されます。
LoRaWAN エンドデバイスは、必ずクラス A で Join し、アプリケーション層からの要求に応じてクラス C に切り替わることができます。

LoRaWANプロトコル
クラスCの管理

- エンドデバイスのクラスC対応/有効
  - クラスBとクラスCに対応したエンドデバイスをクラスBとクラスCで同時に有効にすることはできない
  - クラスC対応のエンドデバイスは、RxCと呼ばれるチャネル/DR パラメータの組み合わせを可能な限り使用してリッスンする
  - エンドデバイスは、クラスAの定義に従い、送信もRx1とRx2のどちらで受信もしていないときに、RxCでリッスンする必要がある
- クラスCモードで動作中のRxの優先順位
  - エンドデバイスからクラスAのアップリンク･メッセージをTx、Rx1、またはRx2の時間フレームで送信する場合は、クラスA設定が優先される この期間にRxCウィンドウでサーバから送信されたメッセージはすべて失われる
  - Rx1でダウンリンク･フレームを受信しているエンドデバイスはRx2を開かないので、そのエンドデバイスはただちに連続的なRxCを開く
- デフォルトでは、RxCパラメータは Rx2パラメータと同じである（同じチャネルと同じデータ･レート）
- クラスCのメッセージは、ユニキャスト（単一のエンドデバイスへの送信）とすることも、マルチキャスト（複数のエンドデバイスへの送信）とすることも可能　異なるパラメータを使用してこの2つの方法を同時に有効にした場合は、RxCウィンドウの設定にどちらを使用するかをアプリケーション層で判断する必要がある

life.augmented

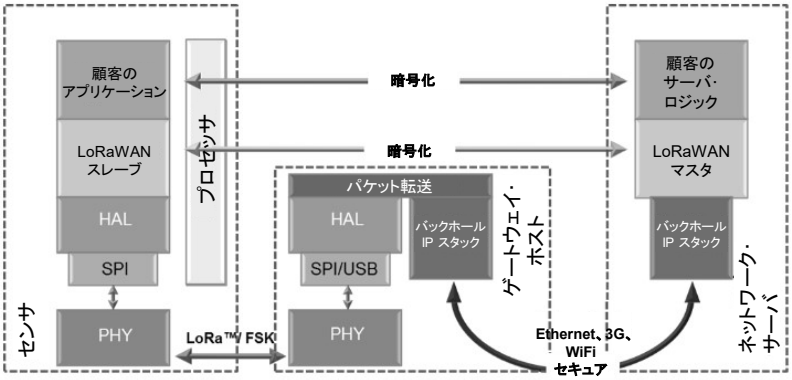
13

クラス B とクラス C で動作できるデバイスは、クラス A とクラス B またはクラス A とクラス C のどちらかでのみ同時に動作できます。
デフォルトでは、RxC の設定は Rx2 と同じパラメータで設定されます。
クラス B では RxC ウィンドウを使用して、ユニキャスト･メッセージを受信できるほか、ネットワーク･サーバで定義されているオプションのマルチキャスト設定によってマルチキャスト･メッセージを受信することもできます。

LoRaWANプロトコル

プロトコル·スタックの概要

- LoRaWANプロトコル·スタックは次の3層で定義:
  - アプリケーション層
  - MAC層(ネットワーク)
  - 物理層
- LoRaWANでは、エンドデバイスとネットワーク間のデータ交換のセキュリティを、ネットワーク·レベルとアプリケーション·レベルで確保するために、エンドデバイスごとに一意の鍵を複数実装



life.augmented

LoRaWAN プロトコル·スタックは次の 3 層で定義されています。

- アプリケーション層
- MAC 層(ネットワーク)
- 物理層

LoRaWAN では、エンドデバイスとネットワーク間のデータ交換のセキュリティを、ネットワーク·レベルとアプリケーション·レベルで確保するために、エンドデバイスごとに一意の鍵を複数実装しています。

LoRaWANプロトコル

LoRaネットワークのセキュリティ

- LoRaWANメッセージの機密性は、FRMPayloadフィールドをAES-128で暗号化することにより保護される LoRaWANは、暗号化されていないFPortフィールドを利用して、ネットワーク・サーバ宛てのMACメッセージとアプリケーション・サーバ宛てのアプリケーション・メッセージを区別する したがって、FRMPayloadフィールドでは、転送先に応じて2種類の暗号化キーが使用される
- したがって、LoRaWANのエンドデバイスは、MACメッセージ暗号化用のネットワーク・セッション鍵（NwkSKey）とアプリケーション・メッセージ暗号化用のアプリケーション・セッション鍵（AppSKey）の2つの128ビット暗号化鍵を必要とする
- **AppSKey**はエンドデバイスとアプリケーション・サーバ間で共有する暗号化鍵であり、**NwkSKey**はエンドデバイスとネットワーク・サーバ間で共有する暗号化鍵である

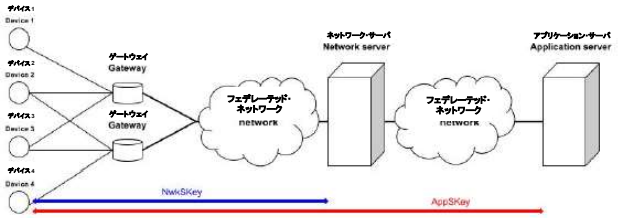


図2.5：AppSKeyとNwkSKeyの共有と使用の様子を表したLoRaWANアーキテクチャ

life.augmented

15

ネットワーク層メッセージ（MAC）とアプリケーション層メッセージを暗号化する 2 つの AES 128 ビットのセッション鍵によってこのセキュリティが実現しています。
これらの鍵は、プログラミングの段階でエンドデバイスに提供するか、Join のステップで中間鍵を使用して生成されます。

LoRaWANプロトコル

LoRaネットワークのセキュリティ

- メッセージの完全性を実現するために、AES-CMACとNwkSKeyを使用して、パケットごとにメッセージ完全性コード（MIC）が計算される　パケットの改ざんを検出するために、**FRMPayload**を暗号化した後でPHYPayload全体に対してMIC計算が実行される

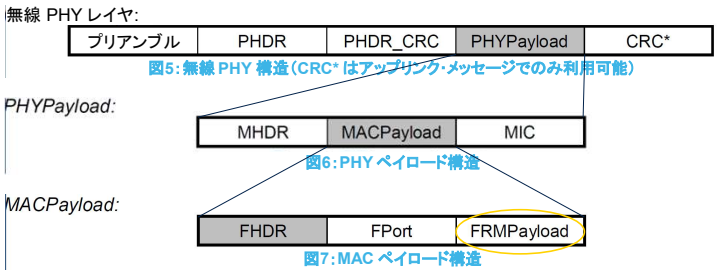
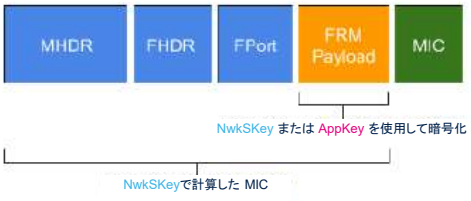


図5：無線 PHY 構造（CRC* はアップリンク・メッセージでのみ利用可能）

図6：PHY ペイロード構造

図7：MAC ペイロード構造



life.augmented

メッセージの完全性を実現するために、AES-CMAC と NwkSKey を使用して、パケットごとにメッセージ完全性コード（MIC）が計算される。FRMPayload を暗号化した後で PHYPayload 全体を対象として MIC が計算され、パケットの改ざんがないかが検査される。

LoRaWANプロトコル

LoRaネットワークのセキュリティ（コミッショニング）

- OTAA（Over-The-Air Activation）
  - OTAAのコミッショニング・パラメータ:
    - エンドデバイスのDevEUIは、WiFiのMACアドレス（エンドデバイスを一意に識別するためのID）と同様に機能する
    - AppEUI（JoinEUI）は、WiFiのSSID（ネットワーク・サーバを一意に識別するためのID）と同様に機能する
    - OTAAのAppKeyは、WiFiのパスワードと同様の機能（セキュアなアプリケーション・キー）
  - OTAAは、次のように実行される無線（over the air）メッセージによるハンドシェイクを使用
    - エンドデバイスは、ネットワーク・サーバに**join-requestメッセージ**を送信（MHDRのMTypeフィールドを 000 に設定することで指定） join-requestメッセージは暗号化されていないが**MIC**で保護される
      この場合のMICは、AES-128鍵であるアプリケーション鍵（AppKey）を使用して計算される
      join-requestの内容は次のとおり
      - グローバルに一意のエンドデバイス識別子（DevEUI）
      - アプリケーション識別子（AppEUI）
      - 2オクテットのノンス（DevNonce） DevNonceは、ネットワーク・サーバが各エンドデバイスについて記録するランダムな値
  - ネットワークへのJoinが許可されたエンドデバイスは、ネットワーク・サーバから**join-acceptメッセージ**（MHDRのMTypeフィールドを 001 に設定することで指定）を受信する

life.augmented

17

これらの鍵を取得するために、LoRaWAN のエンドデバイスをアクティブ化する方法としてOTAA と ABP の 2 種類があります。
OTAA モードでは、エンドデバイスを一意に特定する要素（DevEUI）が必要になります。
また、エンドデバイスは、AppEUI と AppKey を使用して接続先ネットワークの識別情報とネットワーク鍵を知る必要があります。
このステップは Join Request として定義されます。エンドデバイスからの要求がネットワークで受信されて認められると、エンドデバイスは、Join で作成されたセッション鍵を使用して暗号化メッセージの交換を開始できます。

LoRaWANプロトコル

LoRaネットワークのセキュリティ（コミッショニング）

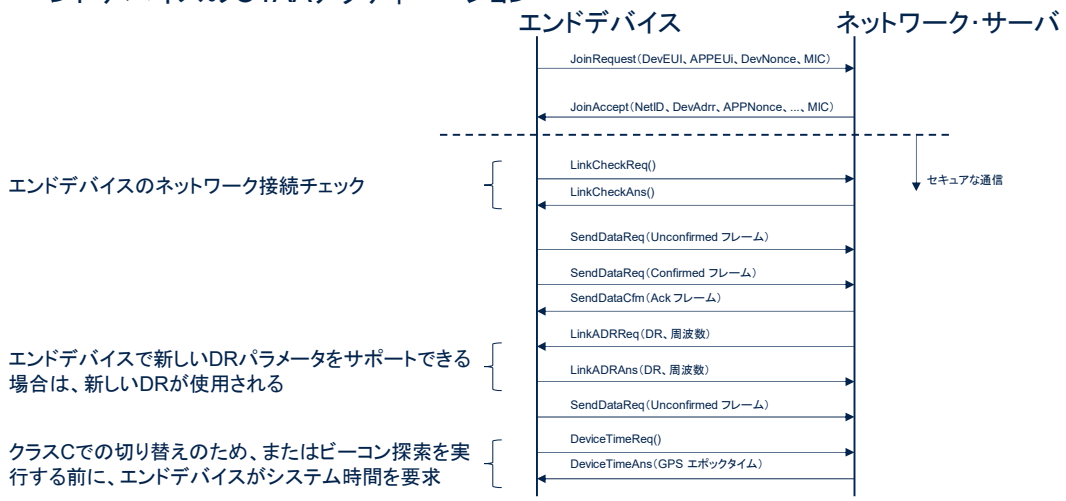
- ABP（Activation by personalization）
  - LoRaWANネットワークにJoinして鍵を共有するもう1つの方法としてABP（Activation By Personalisation）手順がある
  - ABPでは、ネットワーク・キーとアプリケーション・キーを事前に共有して使用する。これらのキーは、製品出荷時に格納される
    - デバイス・アドレス（DevAddr）
    - ネットワーク・セッション・キー（NwkSKey）
    - アプリケーション・セッション・キー（AppSKey）

life.augmented

ABP モードでは、ネットワークとエンドデバイスとの間で事前定義のセッション鍵を使用します。エンドデバイスの生産フェーズで、これらの鍵をプログラミングする必要があります。
この時点で、これらの秘密鍵を使用した暗号化メッセージの交換によって、デバイスはただちに起動できます。
まとめると、ABP 方式に比べ、OTAA 方式は実装が複雑であるが、セキュリティの水準は高い。実際、セッション鍵は、ABP では静的ですが、OTAA では Join リクエスト・セッションごとに派生されます。

LoRaWANプロトコル
メッセージ·シーケンス図

- エンドデバイスのOTAAアクティベーション



life.augmented

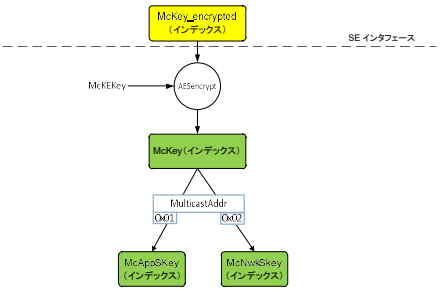
19

このシーケンス図では、MAC メッセージとアプリケーション·メッセージを通じて OTAA モードでエンドデバイスが起動する手順を示しています。
ネットワークの推奨事項と制限事項に従い、MAC メッセージを使用してエンドデバイスを設定します。

LoRaWANプロトコル
マルチキャスト鍵の派生

- McKeyからMcAppSKeyとMcNetSkeyを派生する方法
  - キー派生の説明と、ユニキャスト･キーとマルチキャスト･キーの区別
  - アプリケーション層からのマルチキャスト･セットアップで、エンドデバイスが McKey_encrypted（暗号化McKey）を受信する
  - McKeyを取得してセッション･キーを計算するフロー
    - **McKey** = aes128_encrypt(McKEKey, McKey_encrypted)
    - McRootKey = aes128_encrypt(**GenAppKey**, 0x00 | pad_{16})
    - McKEKey = aes128_encrypt(McRootKey, 0x00 | pad_{16})

      最終的に次のようにキーが得られる
    - McAppSKey = **aes128_encrypt(McKey**, 0x01 | McAddr | pad_{16})
    - McNetSKey = **aes128_encrypt(McKey**, 0x02 | McAddr | pad_{16})
    - **GenAppKey**はエンドデバイスでプロビジョニングされた新しいルート鍵（LoRaWAN V1.0.x）



life.augmented

マルチキャストは、クラス B またはクラス C を使用することで利用可能になる通信オーバーレイです。
エンドデバイスとネットワークとの間でユニキャスト暗号化メッセージを交換するためのネイティブのセッション鍵のほか、マルチキャスト･メッセージのセッション鍵があります。
アプリケーション･サーバと複数のエンドデバイスの間でマルチキャスト･グループをセットアップするステップで、これらの鍵が生成されます。
ここでは、これらの鍵を生成するためのさまざまなステップを示しています。

メディア・アクセス制御
MACメッセージのフォーマット

- **MHDR**（MACヘッダ）：次のメッセージ・タイプを指定
  - Join Request/Acceptメッセージ
  - データ・アップ/ダウン・メッセージ　MACコマンドとアプリケーション・データの両方の転送にこれらのメッセージを使用
- **FHDR**（フレーム・ヘッダ）：DevAddr、適応レート制御、フレーム・カウンタ、確認応答を記述するほか、MACコマンドがあればそれも記述
- **FPort**：アプリケーション固有であれば 1 ~ 223 の範囲の値。FRMPayloadにMACコマンドのみが記述されている場合は 0
- **FRMPayload**：データ・フレーム（AppまたはMAC）
- **FCtrl** はフレーム制御、**FOpts**はMACコマンドの転送に使用するフレーム・オプション
- メッセージ完全性コード（**MIC**）は、メッセージにあるすべてのフィールドを対象として算出（AES）

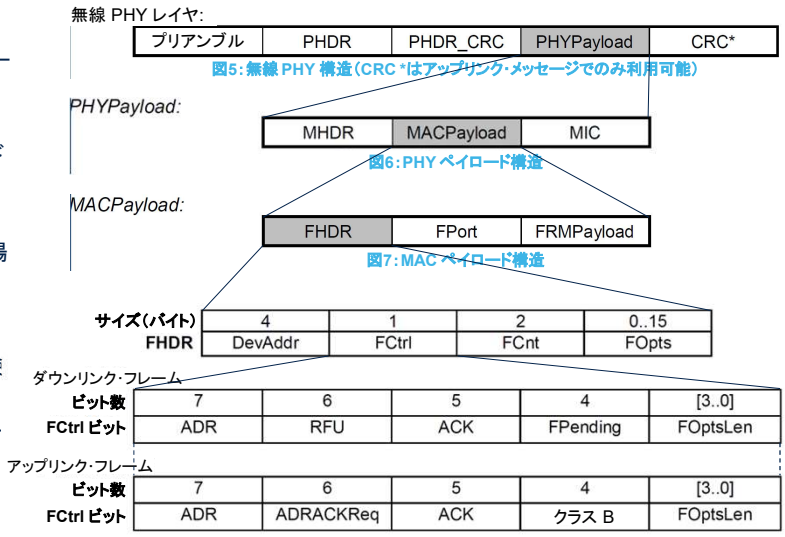


図5：無線 PHY 構造（CRC *はアップリンク・メッセージでのみ利用可能）
図6：PHY ペイロード構造
図7：MAC ペイロード構造

life.augmented

21

LoRaWAN のアップリンク・メッセージとダウンリンク・メッセージは物理ペイロードを伝送します。このペイロードは、MAC ヘッダで始まり、MAC ペイロードが続いて、メッセージ完全性コード（MIC）で終了します。
MAC ペイロードは、フレーム・ヘッダに続いて、ポート・フィールドとオプションのフレーム・ペイロードを含みます。
この MAC ペイロードは、MAC コマンドの交換と各種パラメータの制御に使用するほか、アプリケーション層からのアップリンクの場合はデータ・フレームの制御にも使用します。

メディア･アクセス制御
MACコマンド

- ネットワークを管理するために、ネットワーク･サーバとエンドデバイスのMAC層との間でのみ各種のMACコマンドを交換できる　MAC層コマンドは、アプリケーション･サーバでは認識できず、エンドデバイスで動作しているアプリケーションでも認識できない

CID	コマンド	送信元		簡単な説明
		エンドデバイス	ゲートウェイ	
0x02	***LinkCheckReq***	x		エンドデバイスで使用してネットワークとのコネクティビティを検証
0x02	***LinkCheckAns***		x	LinkCheckReq コマンドに応答。受信品質（リンク･マージン）をエンドデバイスに通知する受信信号強度推定値を格納
0x03	***LinkADRReq***		x	データ･レート、送信出力、冗長性、またはチャネル･マスクの変更をエンドデバイスにリクエスト
0x03	***LinkADRAns***	x		LinkADRReq に確認応答
0x04	***DutyCycleReq***		x	デバイスの最大アグリゲーション送信のデューティサイクルを設定
0x04	***DutyCycleAns***	x		DutyCycleReq コマンドに確認応答
0x05	***RXParamSetupReq***		x	受信スロットのパラメータを設定
0x05	***RXParamSetupAns***	x		RXParamSetupReq コマンドに確認応答
0x06	***DevStatusReq***		x	エンドデバイスのステータスをリクエスト
0x06	***DevStatusAns***	x		エンドデバイスのステータス（バッテリ･レベルと無線ステータス）を返す
0x07	***NewChannelReq***		x	無線チャネルの定義を作成または変更
0x07	***NewChannelAns***	x		NewChannelReq コマンドに確認応答
0x08	***RXTimingSetupReq***		x	受信スロットのタイミングを設定
0x08	***RXTimingSetupAns***	x		RXTimigSetupReq コマンドに確認応答
0x09	***TXParamSetupReq***		x	地域の規制に基づいてエンドデバイスの最大許容滞留時間と最大 EIRP を設定するためにネットワーク･サーバで使用
0x09	***TXParamSetupAns***	x		TXParamSetupReq コマンドに確認応答
0x0A	***DlChannelReq***		x	ダウンリンクをシフトすることによってダウンリンク RX1 無線チャネルの定義を変更

CID	コマンド	送信元		簡単な説明
		エンドデバイス	ゲートウェイ	
				アップリンク周波数からの周波数（非対称チャネルの作成）
0x0A	***DlChannelAns***	x		DlChannelReq コマンドに確認応答
0x0B to 0x0C	***RFU***			
0x0D	***DeviceTimeReq***	x		現在の GPS 時間をリクエストするためにエンドデバイスで使用
0x0D	***DeviceTimeAns***		x	ネットワーク･サーバから送信し、DeviceTimeReq リクエストに応答
0X0E to 0x7F	RFU			
0x80 to 0xFF	財産権	x	x	独自ネットワーク･コマンドの拡張のために予約済み

life.augmented

ネットワークを管理するために、ネットワーク･サーバとエンドデバイスの MAC 層との間でのみ各種の MAC コマンドを交換できます。MAC 層コマンドは、アプリケーション･サーバでは認識できず、エンドデバイスで動作しているアプリケーションでも認識できません。

メディア・アクセス制御
MACコマンド

- クラスBの仕様では以下のMACコマンドが追加される
  クラスB対応エンドデバイスは、前のスライド（クラスA仕様）で説明されているすべてのコマンドを実装する必要がある

CID	コマンド	送信元		簡単な説明
		エンドデバイス	ゲートウェイ	
0x10	***PingSlotInfoReq***	x		ネットワーク・サーバにユニキャスト ping スロットの周期性を送信するためにエンドデバイスで使用
0x10	***PingSlotInfoAns***		x	PingInfoSlotReq コマンドに確認応答するためにネットワークで使用
0x11	***PingSlotChannelReq***		x	エンドデバイスのユニキャスト ping チャネル周波数とデータ・レートを設定するためにネットワーク・サーバで使用
0x11	***PingSlotChannelAns***	x		***PingSlotChannelReq*** コマンドに確認応答するためにエンドデバイスで使用
0x12	***BeaconTimingReq***	x		非推奨
0x12	***BeaconTimingAns***		x	非推奨
0x13	***BeaconFreqReq***		x	エンドデバイスでビーコン・ブロードキャストを受信する際に想定される周波数を変更するためにネットワーク・サーバで使用するコマンド
0x13	***BeaconFreqAns***	x		BeaconFreqReq コマンドに確認応答するためにエンドデバイスで使用

life.augmented

クラス B の仕様では、ここに挙げた MAC コマンドが追加されます。
クラス B 対応エンドデバイスは、前のスライド（クラス A 仕様）で説明されているすべてのコマンドを実装する必要があります。

メディア・アクセス制御
MACコマンド

- MACコマンドは、1バイトのコマンド識別子（CID）と、それに続くコマンド固有のバイト・シーケンスで構成（このバイトは空であることも考えられる）
- クラスAのMACコマンドの例
  - LinkADRReq/Ans（ネットワーク→エンドデバイス）CID = 0x03
    - アップリンク・フレーム送信時に、データ・レート適応の実行をネットワークからエンドデバイスに要求できるようにする
      ChannelMaskフィールドはアップリンク伝送に使用可能なチャネルを提案し、Redundancyフィールドはアップリンク・メッセージごとに「NbTrans」を提案する（「Unconfirmed」アップリンク・フレームにのみ適用）
    - エンドデバイスからは、設定を受け入れたかどうかをネットワーク・サーバに応答メッセージで通知する

サイズ（バイト）	1	2	1
LinkADRReq ペイロード	DataRate_TXPower	ChMask	冗長

ビット	[7:4]	[3:0]
DataRate_TXPower	DataRate	TXPower

サイズ（バイト）	1
LinkADRReq ペイロード	ステータス

ビット	[7:3]	2	1	0
ステータス ビット	RFU	電源 ACK	データ・レート ACK	チャネル・マスク ACK

  - DeviceTimeReq/Ans（エンドデバイス→ネットワーク）CID = 0x0D
    - 現在のネットワークの時間（GPS エポック・タイム）をエンドデバイスからネットワークに要求できるようにする
      エンドデバイスでは、その内部クロックをネットワークのクロックに同期できる　クラスBビーコン取得の高速化に有用

life.augmented

MAC コマンド・リクエストは、一意のコマンド識別子である 1 バイトの CID と、それに続くコマンド固有のバイト・シーケンス（オプション）で定義します。送信するメッセージのタイプに応じて、このリクエストはエンドデバイスまたはネットワーク・サーバによって送信されます。
たとえば、LinkADR リクエストは、データ・レート、TX 出力、使用するチャネルを更新するために、ネットワーク・サーバからエンドデバイスに送信されます。
また、DeviceTime リクエストは、内部クロックをネットワーク時間に同期した状態にするために、エンドデバイスからネットワーク・サーバに送信されます。

メディア･アクセス制御
MACコマンド

- クラスBのMACコマンドの例
  - PingslotInfoReq/Ans（エンドデバイス→ネットワーク）CID = 0x10
    - エンドデバイスがそのpingスロットの周期性をネットワークに通知できるようにする
      このリクエストはユニキャストpingスロットについてサーバに通知するためにのみ使用できる　マルチキャストpingスロットはアプリケーション･サーバで定義されるので、このコマンドは使用しない　マルチキャストpingスロットの周期性は、追加のパッケージ仕様とともにMcGroupSetupReqで指定できる
  - PingSlotChannelReq/Ans（ネットワーク→エンドデバイス）CID = 0x11
    - ダウンリンクpingスロットの周波数やデータ･レートの変更リクエストをネットワークからエンドデバイスに送信できるようにする
      このリクエストは、クラスAモードのRx1ウィンドウまたはRx2ウィンドウでのみ受信できる

life.augmented

エンドデバイスがクラス B 対応であれば、オプションの MAC コマンドをいくつか使用できます。
周波数、データ･レート、周期性の更新によって ping スロットやビーコンの設定を管理する際に、このような命令が効果的です。

メディア·アクセス制御
データ·レートの適応

- 再送信メッセージに対するデータ·レートの適応
  - ネットワークに「Confirmed」アップリンク·フレームを送信したエンドデバイスは、次に続くRxスロットでFCtrlのACKビットを使用した確認応答をネットワークから受信することを想定する
    このACKビットを受信できない場合、エンドデバイスは同じデータの再送信を試みる
  - 前回とは別の周波数による再送信もできれば、異なるデータレート（前回より低いデータレートが望ましい）による再送信もできる
    送信における推奨方針は次のとおり
    - 最初の「Confirmed」アップリンク·フレームをデータレート「DR」で送信する　再送信する場合は、以下のルール·テーブルに従う
    - 8回送信してもメッセージに対する確認応答が得られない場合、MAC層はアプリケーション層に実行エラーを返す
    - 再送信のたびに、標準送信としてランダムに周波数チャネルが選択される

送信回数	データレート
1（1 回目）	DR
2	DR
3	Max(DR-1,0)
4	Max(DR-1,0)
5	Max(DR-2,0)
6	Max(DR-2,0)
7	Max(DR-3,0)
8	Max(DR-3,0)

life.augmented

26

信号に混入したノイズが多く、メッセージが失われる場合、MAC 層では、データ·レートの調整を伴う再送信手順が使用されます。
このデータ·レートは、エンドデバイスとゲートウェイ間の環境と距離に応じて最適な措置をとるために送信速度を調整するパラメータです。

LoRaWAN地域のphyLayer

デューティ·サイクル、LBT、dwell time

- LoRaWANでは、エンドデバイスとゲートウェイが通信できるように次のルールを規定
  - エンドデバイスは、送信のたびに擬似的にランダムな形態でチャネルを変更する
  - エンドデバイスは、使用するサブバンドと地域の規則を基準とした最大送信デューティ·サイクルを遵守する
  - エンドデバイスは、サブバンドと地域の規則を基準とした最大送信時間を遵守する
- デューティ·サイクルを規定した地域の規則に基づき、特定のサブバンドでフレームを送信するたびに、このサブバンドでの放射時間およびフレームのオンエア持続時間が記録される　その後の Toff 秒間は、同じサブバンドを再利用できない。特定のサブバンドを使用できない期間でも、デバイスは別のサブバンドで送信を継続できる　デューティ·サイクルの制限のためにどのサブバンドも使用できない場合、1つ以上の要素が再び使用できるようになるまで、デバイスはメッセージを送信できなくなる
- Toffは次の式で計算できる

$$Toff_{subband} = \frac{TimeOnAir}{DutyCycle_{subband}} - TimeOnAir$$

- たとえば、EU868のISMバンドでは、ETSI規則に準拠するためにサブバンドあたりのデューティ·サイクルが1%に制限されているデューティ·サイクルが1%でTx送信時間が500msであると、使用しているサブバンドを使用できなくなる時間は49.5秒になる
- これ以外に、KR920のISMバンドでは送信管理としてLBTを規定している　この管理では、周波数を使用する前にそれをリッスンすることによってトランシーバを開始する　Dwell Timeは、送信データのサイズに対する規則である

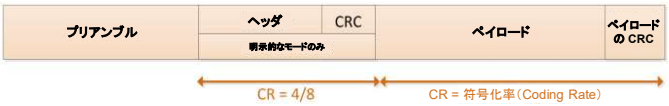
life.augmented

27

すべての LoRaWAN エンドデバイスに適用されるルールがあると同時に、規制当局によって該当の地域でのみ適用されるルールがあります。
たとえば、EU868 の ISM バンドでは、使用周波数の過負荷を防止するために、エンドデバイスからの送信に 1% のデューティ·サイクルが規定されています。
これ以外に、KR920 の ISM バンドでは送信管理として LBT を規定しています。この管理では、周波数を使用する前にそれをリッスンすることによってトランシーバを開始します。Dwell Time は、送信データのサイズに対する規則です。

LoRa物理層

- LoRaモデムでは、以下に示す物理（Phy）パケット構造を使用している



  - **プリアンブル**は、レシーバと受信信号の同期をとるために使用されている プリアンブルのフォーマットは、ISMバンドごとに固有である LoRaWANの地域パラメータ仕様では、0x34の同期ワードを含めて8つのシンボル長がデフォルトの公共LoRa変調で使用されている
  - **ヘッダ**はペイロードに関する以下の情報を提供する
    - バイト単位のペイロード長
    - 前方誤り訂正符号化率
    - オプションの16ビットCRCがペイロードにあるかどうかの指定
  - **ペイロード**は、IEEE 802.15.4勧告に準拠してMACヘッダ、MACペイロード、MAC完全性コードを収めている 想定された情報がプリアンブルにもヘッダにもない場合は、物理層からMAC層にペイロードが送信されない
  - **CRC**はアップリンク･メッセージにのみ存在する

life.augmented

LoRa パケットは、受信信号とレシーバとの同期に使用するプリアンブルのシーケンスで始まります。デフォルトでは、このパケットは 8 つのシンボル長のシーケンスです。
プリアンブルの後には、以降のペイロードに関する情報を記述したヘッダが続きます。パケットのペイロードは、特定のエラー率でコード化された実際のデータを収めた可変長フィールドです。このエラー率は、明示的なモードではヘッダで指定され、暗黙的なモードではユーザが選択します。CRC が付加されることもあります。

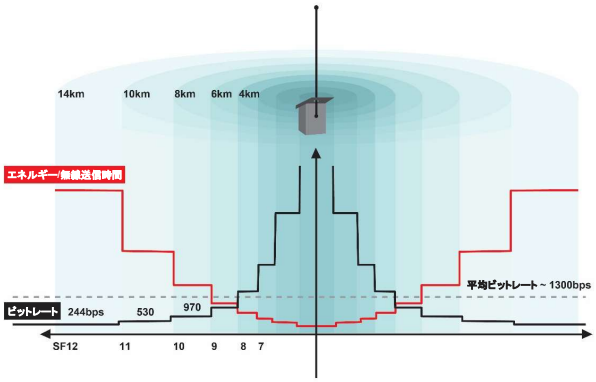
LoRa物理層

- LoRaモデムでは、スペクトラム拡散変調と前方誤り訂正の手法を使用して、無線通信の範囲を広げ、優れた堅牢性を実現している
  - 高い実効データ·レートを使用できる帯域幅（125 ~ 500kHz）により、送信時間を短縮
  - 拡散率（SF5 ~ SF12）は、情報の1シンボルあたりで送信されるビット数
  - 符号化率（4/5 ~ 4/8）は、前方誤り検出訂正（FEC）を実行するためのサイクル·エラー符号化率
- 無線送信時間を計算することでトランシーバの時間利用率が得られる

$$ToA = \frac{2^{SF}}{BW} * N_{symbols}$$

- 無線送信時間の計算機能

  https://www.loratools.nl/#/airtime



life.augmented

29

LoRa モデムの重要な特長として電磁干渉耐性の向上があります。このような優れた電磁干渉耐性があることで、周波数の使用率が高い帯域や、従来の変調方式では通信できない距離まで通信範囲を拡張するために LoRa を使用しているハイブリッド通信ネットワークで、LoRa 変調した複数のシステムの混在が容易です。
特定のアプリケーションに合わせて LoRa 変調を最適化できます。リンク·バジェット、電磁干渉耐性、周波数の占有率、公称データ·レート間のトレードオフを実現する重要な設計パラメータのそれぞれに設計段階でアクセスできるようになります。