Best practices for cellular IoT development

nWP-044

White Paper

1159720_178 / 2022-03-28

NORDIC
SEMICONDUCTOR

Contents

Revision history

Date	Description
2022-03-28	First release

1 Introduction

This document introduces the main aspects and decisions you need to consider before and during your development phase of a low-power cellular *Internet of Things (IoT)* product.

The sections are structured and ordered chronologically to follow a natural way of designing a cellular IoT device. The first sections cover the initial design decisions you might need to make to get the best possible baseline of your design. This includes overviews about the nRF9160 *System in Package (SiP)*, modem radio, and radio technologies. The importance of network coverage and access is also discussed.

Then, high-level topics such as IP, security, and application protocols are presented. The sections discuss IP protocol choices and recommendations, different security protocols, and available application protocols you can use.

Lastly, the document gives information about certificates, third-party modules from Nordic partners, and support channels.

2 Hardware architecture

The nRF9160 *SiP* has a dedicated application processor, *Long-Term Evolution (LTE)* modem, and memory tightly integrated into a *System on Chip (SoC)* with unified power and clock management.

Compared to traditional cellular *IoT* architecture which has a separate MCU, cellular modem, and serial memory interface, the integrated SoC reduces the communication time between devices, the power used in external interfaces, and the total hardware complexity.

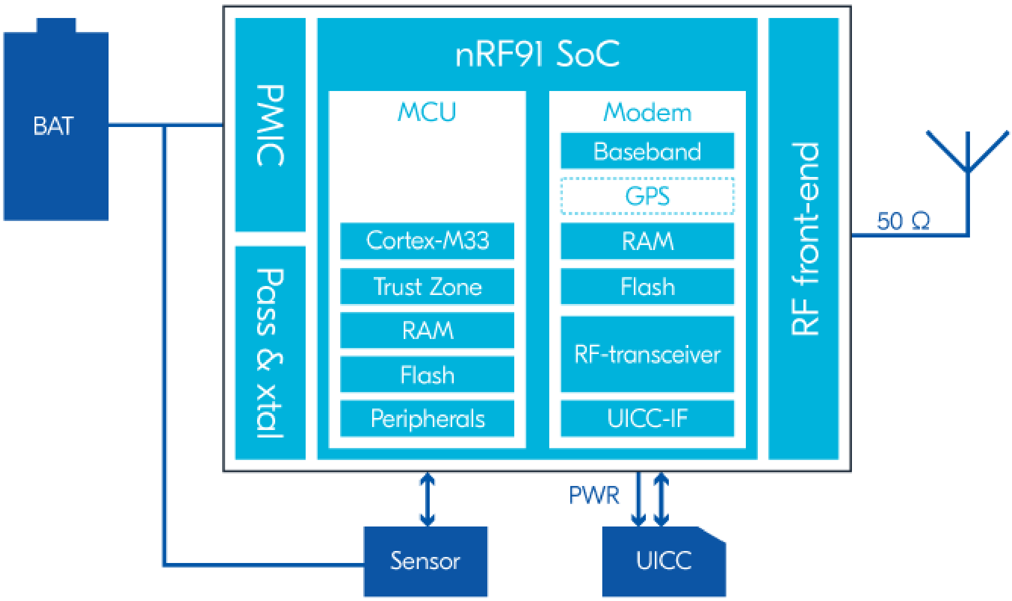

Figure 1: nRF9160 SiP overview

You can have everything running on nRF9160 as a single-chip implementation or you can add nRF9160 as a coprocessor in a larger or more specialized design, where nRF9160 handles communication and low-power or system tasks while the main or specialized CPU sleeps.

You can also use nRF9160 only as a modem if you have a design with older cellular modem technologies (2G or 3G) and want to quickly switch over to *LTE-M* or *Narrowband Internet of Things (NB-IoT)*. Running the Serial LTE Modem application on the application processor enables you to communicate with the nRF9160 modem using only AT commands. This lets you introduce the technology into your designs, and allows you at a later stage to migrate your whole application into the nRF9160's application space to optimize your design for low power, device size, Bill of Materials (BOM), support, and supply chain.

Some considerations for hardware architecture are:

- Are you going to utilize the nRF9160 SiP as a single chip implementation?
- How important is BOM, device size, and low power for your application?

3 Low-power application design through radio focus

Even with a very low-power wireless device, wireless connection usage still has a major impact on your product. Total power consumption and battery life are impacted by your technology choice (*LTE-M* or *NB-IoT*), transport and application protocols you use, and how often you use the link and the amount of data you transfer each time.

An important decision point for your application is how frequently it should communicate with the cloud. Every connection to the cloud spends time and power on overhead. For example, tasks the *LTE* modem must do without providing real value for your application. Every time you send or receive data, some functions with associated overheads are triggered. The overheads are:

- Time and power to initiate the modem
- Time and power to connect to the cellular network
- Time and power to connect to the cloud services
- Headers and security in your chosen application protocol

The nRF9160 *SiP* was designed from the ground up with the first two factors listed above in mind. However, your choices on the last two factors also have a major power consumption impact and need your attention.

LTE offers direct connection to the Internet in a global, secure, and mostly reliable way. This inherently leads to more overhead compared to simpler, local networks. See the LPWANs bridge the IoT divide blog for more context around cellular positions in *Low-power Wide Area Network (LPWAN)*s. By making correct and relevant decisions early in your product design, you can avoid pitfalls and inferior decisions.

If the value in your application includes real-time and low latency delivery of data, that can justify the cost of frequently setting up a connection to transfer small amounts of data. However, frequent sensing of data may not be viable due to the incurred battery size or cost, and the cost of ownership from overhead in both power consumption and data usage.

The key point is to value the data you send versus the cost of sending it. If the value of your application does not always rely on low latency delivery, you can transform your power budget and extend the battery lifetime by leveraging the local processing capacity in nRF9160. Local processing can be anything from a simple collection of data over time and bulk transfers, to local processing or decision making all the way to local machine learning or AI.

Internet protocols were originally made for transferring large amounts of data between computers, so the cost per bit transferred is much lower if you can send larger amounts each time you take the cost of connecting to the cloud, spreading the overhead cost on more bits.

Most IoT applications have a mix of critical or low latency data like alerts or alarms that needs to be sent immediately, and data collected and processed over time that can be sent in larger bulks. If the historical data can be processed locally to reduce the size, then you can also increase the value of each bit as well as lengthening the intervals between transfers. This is the key to achieving a low-power wireless product.

The first and most important decision you need to make is how to use your wireless link:

- What data and in which operational modes do you have latency requirements?
- How much and for how long can you store data locally before sending in bulk?
- How can you process data locally to increase the value of the data to exchange?

These considerations form a foundation for the choices you need to make in the following sections.

4 LTE technology

nRF9160 supports *LTE-M* (*Cat-M1*) and *NB-IoT* (*Cat-NB1*/*Cat-NB2*). Both technologies are used for low-power cellular *IoT* but have some significant differences based on particular use cases.

Features	LTE-M	NB-IoT	
Also known as	LTE CAT-M1, *Enhanced Machine Type Communication (eMTC)*	LTE Cat-NB1 (3GPP rel 13)	LTE Cat-NB2 (3GPP rel 14)
Bandwidth	1.4 MHz	200 kHz	200 kHz
Max throughput (DL/UL)	300/375 kbps	30/60 kbps	127/169 kbps
Latency	50-100ms	1.5-10s	
Typical range	<11 km	<15 km	<15 km
Mobility/cell reselection	Yes	No	Limited
Roaming	Yes	Limited	Limited
Deployment density	Up to 50 000 per cell	Up to 50 000 per cell	Up to 50 000 per cell
Battery lifetime	Up to 12 years[1]		

Table 1: LTE technologies

The biggest difference between LTE-M and NB-IoT is the offered data rate or how much time you spend transferring the data. During this time, the radio is powered on and you consume the current specified in product specifications. With similar peak currents, a technology offering higher data rates always spends less power, because it spends less time transmitting and receiving. Conversely, a lower data rate of lower bandwidth technologies generally has better range and coverage, meaning these technologies are able to maintain the link under more difficult conditions (for example, underground parking spaces).

Note: An nRF9160 product in good to reasonable radio conditions using LTE-M consumes less power than a product using NB-IoT.

LTE-M is the winner in terms of low latency, so applications that require lower latency than LTE-M should be considered. It is also the preferred choice for moving applications because of the support for mobility. This means that the device has automatic cell handover when it moves, compared to NB-IoT where the device loses connection if it goes out of a cell, and needs to renegotiate with another cell. This can have a high cost of power in the long run.

Besides differences in features and data rate differences, network operators also support different technologies around the world, so you must know which technology is supported in the countries or regions where you plan to deploy the product. Most operators support both, so this is not a major issue. But there are areas, like mainland China, where LTE-M is not supported. See the Mobile IoT Deployment Map for more information.

It may be difficult to decide which parameter is the most important (for example, bandwidth or coverage, mobility/roaming or not). It may not only be different for each product you design but also for each product you ship depending on where it is installed (country, operator used, outdoor, indoor, underground). For mobile (moving) products this may also change depending on the location at a specific

[1] An application optimized for power consumption.

time. Therefore, the most popular variant of our nRF9160 SiP is the SICA variant that supports both LTE-M and NB-IoT. When using this variant, the decision is made by your application depending on the environment that your particular product is installed in, or even operate in with real-time switching between the technologies. This means you can effectively use NB-IoT for day-to-day operations, but for *Firmware-Over-The-Air (FOTA)* updates you switch to LTE-M to get the benefit of the higher data rates.

Some considerations when choosing the LTE technology are:

- Is your device going to be stationary and does not require low latency?
- Is your device going to be buried underground, or in concrete buildings?
- Is the technology supported in the location where you plan to deploy the product?

In most cases we see that LTE-M is the preferred technology if you do not need the high penetration feature and lower bandwidth that NB-IoT offers.

5 Network coverage and SIM cards

You need to determine early in the project if you are launching a global or region-specific product.

A common issue when developing a cellular *IoT* application is not being able to connect to the network, even if *LTE-M* or *NB-IoT* is supported in the area according to the Mobile IoT Deployment Map. This is usually because the *Subscriber Identity Module (SIM)* card does not have a *LPWAN* subscription or it is not supported by the *Mobile Network Operator (MNO)* in that area. For example, using a roaming SIM card that does not have an agreement set up with that MNO. It is important to note that MNOs differentiate between standard IoT 2-4G SIMs and LPWAN SIMs, where the latter is needed for LTE-M/NB-IoT connections. An example is the two versions of Vodafone Global IoT SIM cards, 4G with IMSI 20404 and LPWA with IMSI 901288.

You need to use the correct SIM card to get connected. Nordic Semiconductor ships all their nRF9160 *Development Kit (DK)*s and Nordic Thingy:91™ prototyping platforms with an iBasis SIM roaming card that can be used in these areas. If your country is not on this list, you would either need to get a local SIM card or a roaming SIM card from another provider.

5.1 Roaming versus local SIM cards

Roaming *SIM* cards can access different networks around the world if the SIM manufacturer has a roaming agreement with the network outside their local network.

This means that the network looks at the device as roaming in their network and often have some different configurations compared to a local device. For example, many roaming SIM cards lack agreements around the world to access the important *Power Saving Mode (PSM)* and *Extended Discontinuous Reception (eDRX)* features. You must check with your SIM card provider if these features are available. If your device cannot get in to PSM or use eDRX intervals in a network, you need to make sure your application turns the modem completely OFF if the device is going to sleep for a long time.

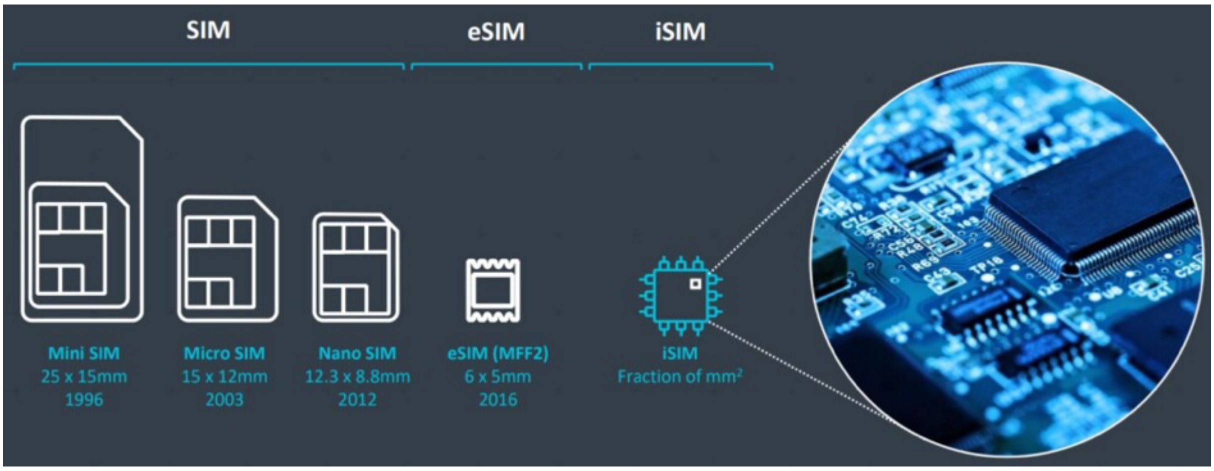

Figure 2: SIM card form factors

See this blog post to learn more about the different types and form factors of SIM cards.

5.2 SIM card power consumption

SIM cards can have a big impact on the average current consumption on your designs.

Historically, SIM cards were designed for devices with big batteries and current consumption optimization was typically not considered during development. However, in ultra-low power devices like nRF9160, we often see SIM idle currents that are 10 times larger than the *PSM* floor current of nRF9160.

Current measurements are easy with the accessible pins on the nRF9160 DK. We recommend that you ask your SIM card provider about this and gather how much current their SIM card is consuming in *clock stop mode*.

Our advanced modem automatically optimizes the power consumption of the SIM card, but you can further improve this if the SIM card has the *UICC suspend-resume* feature enabled (specified in *3GPP TS 31.102 chapter 5.1.11*). You can request this when ordering SIM cards to ensure that the SIM card does not use unnecessary current when not in use.

Some considerations when choosing a local or roaming SIM are:

- Is your device going to be moving between countries?
- Do you need to use PSM and *eDRX* intervals to save power?
- Are you going to have a SIM socket or use an *Embedded SIM (eSIM)* in your design?

6 IP transport options

You can choose the transport protocol to use for transferring data.

The protocols are:

- *Transmission Control Protocol (TCP)/Internet Protocol (IP)*
- *User Datagram Protocol (UDP)*/IP
- *Non-IP Data Delivery (NIDD)*

The protocols differ in terms of their properties and the overhead associated with sending data over the air. This affects both the power consumption of your device and the data costs on the cellular network.

Option	Advantages	Disadvantages
TCP/IP	Data retransmissions Congestion control In-order delivery Error detection	Slow handshake Larger overhead Repetition if data is not received Not suited for NB-IoT
UDP/IP	No handshake needed Better suited for low-power devices	Not guaranteed delivery Not all major cloud vendors support UDP yet
NIDD	Removes the IP overhead Network optimized	Not many networks support it Only supported on NB-IoT Not many cloud vendors support it

Table 2: Transport protocols

Compared to TCP/IP, UDP/IP is better suited for low-power devices because of the difference in overhead and the handshake part is not needed. Based on your specific use case of application it is good to know which route you should take. For example, consider if data should be acknowledged at the destination or is power consumption more important.

In NIDD, the IP protocol is not used as a transport layer which removes big parts of the overhead. NIDD is the most power efficient protocol, but has major disadvantages presented in the table above.

Some considerations when choosing which IP protocol to use are:

- Is ultra-low power and data cost a high priority in your design?
- Do you know which IP protocols your cloud service supports?

7 Security protocols

You can use *Transport Layer Security (TLS)* for *TCP* and *Datagram Transport Layer Security (DTLS)* for *UDP* as security protocols. Adding security to TCP or UDP increases the overhead of the data sent over the air.

However, there are other ways to add security to your protocols without adding too much overhead. For example, you can set up a secure channel by authenticating the device with a *Pre-shared Key (PSK)*. This saves both power consumption and data costs, while still having extra security in data transfers.

An important aspect in *IoT* devices is that they are secured against any malicious attacks. Arm® TrustZone® and Arm CryptoCell™ 310 are available in the nRF9160 *SiP* to securely store data and keys, and to decrypt and encrypt your data. Nordic also has also Trusted Firmware-M support which is the reference implementation of the *Platform Security Architecture (PSA) IoT Security Framework*.

The nRF9160 modem normally handles DTLS/TLS security. This can be handled by the application processor using mbedTLS instead, if there are some TLS Cipher suites or features that you want to use that are not supported by the modem.

For *FOTA* updates, you must sign your firmware images before going into production. This ensures that your device only updates with the firmware images that you provide.

Some considerations when adding security to your device are:

- Is ultra-low power and data cost high priority in your design?
- Are you going to store keys onto your device?
- Make sure to sign your firmware images before going into production.

8 Application protocols

You can use raw application protocols like *UDP* or *TCP* for communications, or you can add additional overhead but also get the benefits by using application protocols like *Constrained Application Protocol (CoAP)*, *Lightweight M2M (LWM2M)*, *Message Queueing Telemetry Transport (MQTT)* or *Hypertext Transfer Protocol Secure (HTTPS)*. Check which protocols are supported by your cloud providers to narrow down your options.

nRF9160 supports UDPs (CoAP and LWM2M) and TCPs (MQTT and HTTPS). In some cases, TCPs can cause issues if you are using *NB-IoT* because it is not required for NB-IoT to support TCP according to the 3GPP specification. If you need to use TCP, it is important to check your NB-IoT network by field testing or asking the carrier for that specific network about support.

MQTT is a widely supported protocol by cloud vendors and is already used by many IoT applications. Unfortunately, it is not a very power-efficient protocol because of the added overhead. HTTPS is a well-established standard but is not the most suitable protocol for a low-power device because of its large overhead. However, it is an effective protocol for *FOTA* updates.

The UDP based protocols are preferred when it comes to optimized power consumption, and the current lack of support by some cloud vendors is slowly but surely getting addressed. LwM2M has gained attention lately because it is specifically designed to reduce power and data usage on low-power devices. It is a popular protocol used for device management alongside CoAP to handle other data communication.

For possible data protocols, you should consider if you can optimize the data sent over the network by switching data protocols. For example, using CBOR instead of JSON reduces the amount of data that is sent, but this is dependent on the cloud site having support for it.

See our webinar on Cloud connectivity and protocols for the Internet of Things for more information on data protocols and transport protocols that you should consider in your designs.

Some considerations when selecting application protocols are:

- Is ultra-low power and data cost a high priority in your design?
- Do you know what is supported on your cloud side?

9 Cloud services and connections

The types of data exchanged between a device and the cloud are application, user, or device management data.

Application or user data are processed measurements gathered from the sensor data by the device. For example, processed data that has gone through a machine learning model.

For sensor applications, you should transfer large amounts of data over the air rarely instead of small amounts often to save on power and network data costs. nRF9160 can provide the application with a collection of the connectivity statistics. This enables tracking of the amount of data transmitted to or received by the device, and helps you in optimizing the network traffic. *MNO*s also provide detailed per-SIM data usage statistics, often in real-time, which allows you to monitor data usage without needing to implement reporting in the firmware.

Device management data refers to the state and configuration of the device. This includes battery level, device health, location, device behavior configuration (for example, switching sensitivity on a motion sensor), and *FOTA* updates. FOTA images are 2-3 times larger in size compared to the other control messages. To limit the time the radio needs to be powered on, it is recommended to suspend all other operations and use all resources on the firmware update. It is also highly recommended to use the *LTE-M* technology (if available) when doing FOTA to benefit from the higher throughput of the technology. Use the connection parameters evaluation feature in nRF9160 to check the quality of your connection before doing a FOTA to avoid unnecessary high-power consumption.

For a low-power device, it is important to restrict how many cloud services you are connected to at a single time to avoid the added overhead of connecting to several cloud connections. You should strive to limit the number of cloud connections from your device to one and instead focus on cloud-to-cloud services. This means you can have a secure connection to your specific cloud solution, and then gather other cloud services (for example, location services, FOTA services) from nRF Cloud through REST APIs, so you get the benefits from the services as well as the power saving/network costs on the device. In other words, let the cloud handle connections to several cloud services, not the constrained device.

Include the LwM2M carrier library in your application if your product will be certified with these carriers.

Some things to consider when looking at cloud services and connections are:

- How often do you need updates from your device?
- Can you switch to LTE-M when doing FOTA?
- Can you access the cloud services through cloud-to-cloud connections?

9.1 Remote observability

During development and production, it is extremely useful to remotely monitor your fleet of devices through remote debugging, fleet management, monitoring, and *Over-the-Air (OTA)* updates. If you find any malfunctioning devices, you can remotely debug and update those devices with new firmware.

We recommend our solution partner Memfault for this specific purpose. Nordic developers get access to up to 100 devices for free when they Memfault register. To implement Memfault in your application, check the Memfault library documentation and the nRF9160: Memfault sample for more details. This does not take a lot of flash space, and it can save costs to remotely debug and monitor devices instead of going out in the field.

You should consider how important it is that your devices are robust and running stable.

9.2 Support for modem trace collection

During firmware development it is crucial to be able to debug any network related issues and take modem traces from your custom design. This can provide valuable insight to yourself or the Nordic support team if you encounter any issues.

10 Local application processing

In constrained cellular IoT application designs, it is important to note that using the radio link for communication can be costly on your power budget if it is used too often.

Therefore, you need to consider how to you use your link. A rule of thumb is to process as much as you can on the application processor, and then send only the essential information as rarely as you can to optimize for low power.

To reduce the radio link usage, you can use different power saving techniques. For example, using *PSM* or increased *eDRX* intervals depending on how you want to use your radio link. If your device is rarely going to send anything to the cloud, it can be beneficial to use PSM to avoid the power consumption penalty of reattaching before sending compared to turning the modem completely off. Because the nRF9160 hardware is optimized for low power, you can leverage our features by following our Power optimization guide.

We have an Online Power Profiler tool to estimate power consumption for a device. After your initial estimations, we recommend using the Power Profiler Kit II to measure the current consumption. During development, it is important to test your design on live networks to debug and improve power consumption. It is also useful to verify supported network configurations in your location, which will reveal the device configurations needed to extend the battery life of your product as much as possible.

Edge computing means doing the computational processing on the *edge* and limiting the data sent over the air by sending only the preprocessed data. The introduction of TinyML now enables machine learning models and training running on the constrained devices. Because the nRF9160 SiP is supported on the Edge Impulse platform, it is possible to train the model using the vast computation power of the cloud and then moving the trained model into your application to run it on the edge device. This means that you can capture a lot of data, run it in your machine learning model, and get back the relevant result. For example, you have a scenario where you constantly send accelerometer data (x,y,z data plots) from a sensor over the air, which could draw ~60 mA and then be processed on the cloud. Compare this to locally processing the same data and just sending a small information event over the air when something interesting is happening, which would draw ~3 mA instead. An example of this information sent to the cloud is `Pump has stopped working, please inspect locally`.

We recommend using the nRF9160 DK for evaluating and developing your applications, and following the Getting started with nRF9160 DK guide.

Some considerations about local application processing are:

- How much of your data can be processed on the device?
- Have you considered the use of PSM or eDRX to save power?
- Can the data capture be analyzed by a machine learning model?

See the Improving Energy Efficiency for Mobile IoT white paper for more information on the low-power features of *LTE-M* and *NB-IoT*.

11 Antenna design recommendations

For the best possible antenna performance, it is important that the antenna design has been properly arranged.

nWP033 - nRF9160 Antenna and RF Interface Guidelines gives recommendations on how to achieve the best possible performance. If good *Global Navigation Satellite System (GNSS)* and *LTE* performance is important in your application, it is crucial to have a good antenna design as early as possible.

We have several antenna manufacturers as design partners, and we recommend that you contact them early in the hardware design process to get the best possible performance and support.

We also recommend that you learn how you can use the GNSS on the application side for best performance.

12 Certification

See nRF9160 certifications to get an overview of all the nRF9160 *SiP* certifications.

We also recommend reading this blog on how to certify your cellular IoT device.

The following are typical certifications you need depending on where the product is going to be shipped:

- *Global* – GCF and PTCRB are global certifications designed to ensure compatibility with the *LTE* 3GPP specification so that the device can communicate with the *E-UTRAN Node B (eNB)*s using *LTE-M* or *NB-IoT*.
- *Regulatory* – The regulatory certifications are regional in nature and are required in order to use radio equipment in a region.
- *MNO* – Some MNOs have their own certification programs but most MNOs only require GCF or PTCRB certifications. Before selecting a network SIM card, check if they require network certification.

You can also simplify your design and certification process by selecting an end-device certified modem as described in this blog. See the full list of third party end-device certified modules here.

13 Nordic partners

To help you bring successful products to market, Nordic has established several partnerships with leading solution providers. These companies can help you with your product journey, by providing products, services, and solutions in relation to the Nordic portfolio.

See Nordic Partners if you need help with any part of the process.

14 Conclusions and next steps

Going through the sections in this guide and considering the questions asked based on your specific application increases the chances of successfully developing a low-power, small form factor cellular device.

To start developing, we recommend getting an nRF9160 DK and reading Getting started with nRF9160 DK. We also have the Nordic Thingy:91, a prototyping platform with its own battery, multitude of sensors, and a small form factor. Nordic Thingy:91 is great for making prototypes and showcasing your proofs of concepts.

If you have any technical questions before, during, or after your development, contact our Technical Support team at Nordic DevZone where you can create a public or private ticket. You can also find extensive blogs and guides made by and for developers.

Glossary

AT command

A command used to control the modem.

Cat-M1

LTE-M User Equipment (UE) category with a single RX antenna, specified in 3GPP Release 13.

Cat-NB1

NB-IoT *UE* category with 200 kHz UE bandwidth and a single RX antenna, specified in 3GPP Release 13.

Cat-NB2

An upgraded version of *Cat-NB1*, specified in 3GPP Release 14.

Datagram Transport Layer Security (DTLS)

A communications protocol providing security to datagram-based applications by allowing them to communicate in a way designed to prevent eavesdropping, tampering, or message forgery.

Development Kit (DK)

A hardware development platform used for application development.

Constrained Application Protocol (CoAP)

A specialized web transfer protocol for use with constrained nodes and constrained networks in the Internet of Things.

E-UTRAN Node B (eNB)

The hardware that is connected to the mobile phone network that communicates directly wirelessly with mobile handsets (*UE*), like a base transceiver station in GSM networks. Also known as *Evolved Node B*.

Embedded SIM (eSIM)

A form of programmable *SIM* that is embedded directly into a device.

Enhanced Machine Type Communication (eMTC)

A low-power wide area network radio technology standard developed by 3GPP to enable a wide range of cellular devices and services.

Extended Discontinuous Reception (eDRX)

A method to conserve the battery of an *IoT* device by allowing it to remain inactive for extended periods.

Firmware-Over-The-Air (FOTA)

A firmware update performed remotely over the air (OTA).

Global Navigation Satellite System (GNSS)

A satellite navigation system with global coverage. The system provides signals from space transmitting positioning and timing data to GNSS receivers, which use this data to determine location.

Hypertext Transfer Protocol Secure (HTTPS)

An extension of the Hypertext Transfer Protocol (HTTP) used for secure communication over a computer network.

Internet of Things (IoT)

Physical objects that are embedded with sensors, processing ability, software, and other technologies that connect and exchange data with other devices and systems of the Internet or other communications networks.

Internet Protocol (IP)

The network layer communications protocol in the Internet protocol suite for relaying datagrams across network boundaries. Its routing function enables internetworking, and essentially establishes the Internet.

Lightweight M2M (LWM2M)

A device management protocol developed by OMA SpecWorks designed for sensor networks and the demands of a machine-to-machine (M2M) environment.

Long-Term Evolution (LTE)

A wireless broadband communication standard for mobile devices and data terminals, based on the GSM/EDGE and UMTS/HSPA technologies.

Low-power Wide Area Network (LPWAN)

A type of wireless telecommunication wide area network designed to allow long-range communications at a low bit rate among things, such as sensors operated on a battery.

LTE-M

An open standard that is most suitable for medium throughput applications requiring low power, low latency, and/or mobility, like asset tracking, wearables, medical, POS, and home security applications. Also known as Cat-M1.

Message Queueing Telemetry Transport (MQTT)

A lightweight, publish-subscribe network protocol that transports messages between devices.

Mobile Network Operator (MNO)

A provider of wireless communications services that owns or controls all the elements necessary to sell and deliver services to an end user.

Narrowband Internet of Things (NB-IoT)

A narrowband technology standard with longer range, lower throughput, and better penetration in, for example, cellars and parking garages compared to LTE-M. NB-IoT is most suitable for static, low throughput applications like smart metering, smart agriculture, and smart city applications. Also known as *Cat-NB1* or *Cat-NB2*.

Non-IP Data Delivery (NIDD)

A technology that transmits data to *IoT* without using the *IP*.

Over-the-Air (OTA)

Refers to any type of wireless transmission.

Power Saving Mode (PSM)

A feature introduced in 3GPP Release 12 to improve battery life of *IoT* devices by minimizing energy consumption. The device stays dormant during the PSM window.

Pre-shared Key (PSK)

A password authentication method, a string of text, expected before a username and password to establish a secured connection. Also known as a shared secret.

Subscriber Identity Module (SIM)

A card used in *UE* containing data for subscriber identification.

System on Chip (SoC)

A microchip that integrates all the necessary electronic circuits and components of a computer or other electronic systems on a single integrated circuit.

System in Package (SiP)

Several integrated circuits, often from different technologies, enclosed in a single module that performs as a system or subsystem.

Transmission Control Protocol (TCP)

One of the main protocols of the Internet protocol suite. TCP provides reliable, ordered, and error-checked delivery of a stream of octets (bytes) between applications running on hosts communicating via an IP network.

Transport Layer Security (TLS)

A cryptographic protocol that provides end-to-end security of data sent over a computer network.

User Datagram Protocol (UDP)

One of the core members of the Internet protocol suite that lets computer applications send messages (datagrams) to other hosts on and Internet Protocol network. Prior communications are not required to set up communication channels or data paths.

User Equipment (UE)

Any device used by an end-user to communicate. The UE consists of the Mobile Equipment (ME) and the Universal Integrated Circuit Card (UICC).

Legal notices

By using this documentation you agree to our terms and conditions of use. Nordic Semiconductor may change these terms and conditions at any time without notice.

Liability disclaimer

Nordic Semiconductor ASA reserves the right to make changes without further notice to the product to improve reliability, function, or design. Nordic Semiconductor ASA does not assume any liability arising out of the application or use of any product or circuits described herein.

Nordic Semiconductor ASA does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. If there are any discrepancies, ambiguities or conflicts in Nordic Semiconductor’s documentation, the Product Specification prevails.

Nordic Semiconductor ASA reserves the right to make corrections, enhancements, and other changes to this document without notice.

Life support applications

Nordic Semiconductor products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury.

Nordic Semiconductor ASA customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Nordic Semiconductor ASA for any damages resulting from such improper use or sale.

RoHS and REACH statement

Complete hazardous substance reports, material composition reports and latest version of Nordic's REACH statement can be found on our website www.nordicsemi.com.

Trademarks

All trademarks, service marks, trade names, product names, and logos appearing in this documentation are the property of their respective owners.

Copyright notice

QUALITY SYSTEM CERTIFICATION

DNV

ISO 9001