Ultra low power PPP connection using nRF9151

NORDIC
SEMICONDUCTOR

WHITEPAPER

Ultra low power PPP connection using nRF9151

ABSTRACT: This white paper describes a method to achieve 2 µA power consumption while maintaining IP networking access in a setup typical for cellular dial-up modems. By properly handling the power state of the UART interface used for PPP communication, power consumption can be reduced by over 99%—from approximately 400-450 µA down to approximately 2 µA during idle periods.

This dramatic power reduction eliminates the need to tear down the PPP connection during extended idle periods, enabling battery-powered IoT devices to maintain network connectivity for many hours or even days while consuming minimal power. The power management is handled transparently at the system level, allowing application developers to focus on their core functionality without cluttering the application logic with modem power state management logic.

To illustrate the impact, consider a realistic IoT application scenario: a device powered by a 2000 mAh battery that sends 500 bytes of data once per hour. Nordic Semiconductor's Online Power Profiler for LTE estimates that the average radio consumption is 30.61 µA over one hour period where PSM mode sleep current is about 2.7 µA.

With traditional UART power management where the UART remains open during PSM, the idle power consumption is around 450 µA. When radio consumption is counted in, the estimated battery lifetime is around 170 days.

However, with the power-optimized approach where the UART is closed during PSM idle periods, the idle power consumption drops down to 2 µA level. When the estimated hourly average is used,the battery lifetime is increased to 7 years.

Typical Cellular PPP Modem setup

A typical cellular PPP modem configuration consists of a host processor connected to a cellular modem via a RT interface. The host processor establishes a Point-to-Point Protocol (PPP) connection over the serial link to enable IP networking capabilities, effectively treating the cellular modem as a network interface similar to traditional dial-up modems.

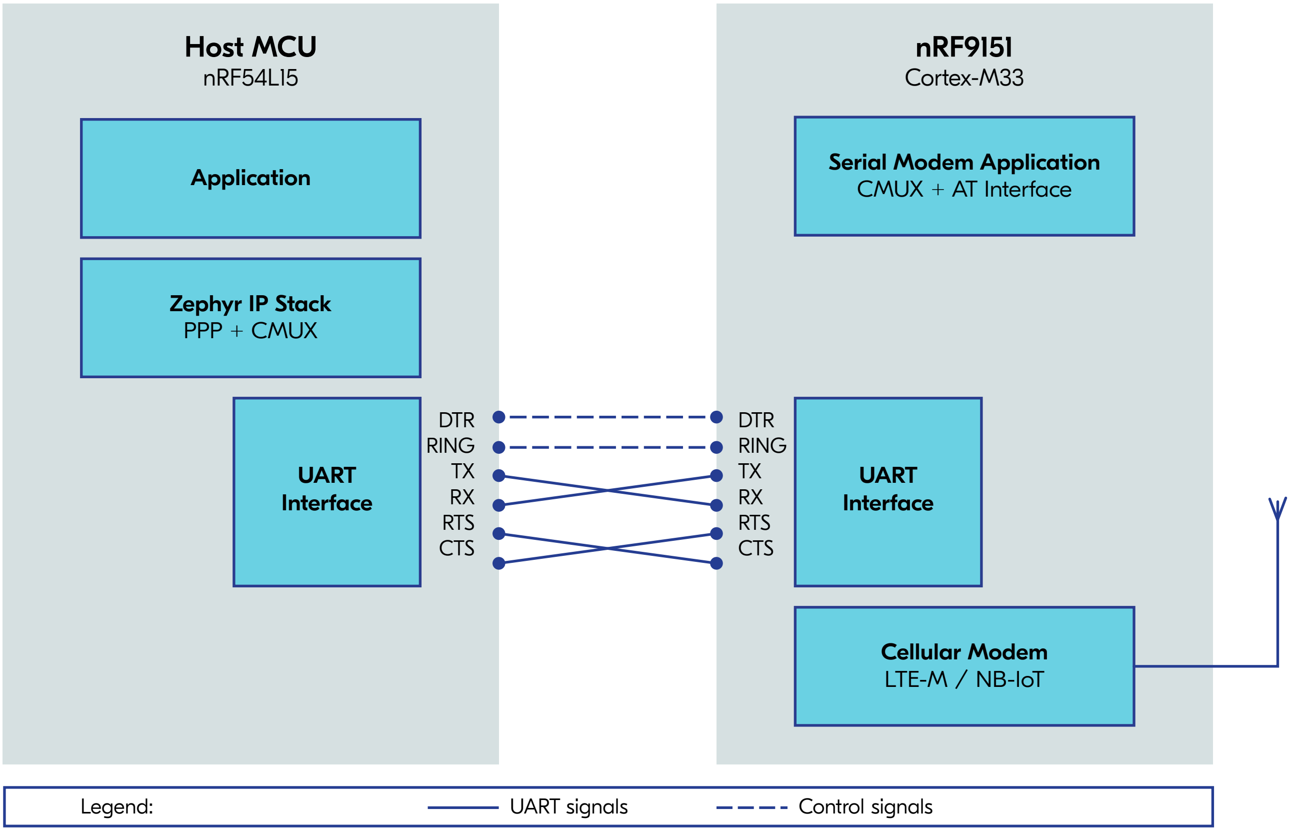

Figure 1. Cellular PPP modem setup

The UART connection serves as the physical transport layer, carrying both AT commands for modem control and PPP data frames for network communication. However, a fundamental challenge arises when using PPP: once the PPP connection is established, the serial link is dedicated to PPP data frames, making it difficult to send AT commands for modem management andmonitoring.

PPP Frame Structure

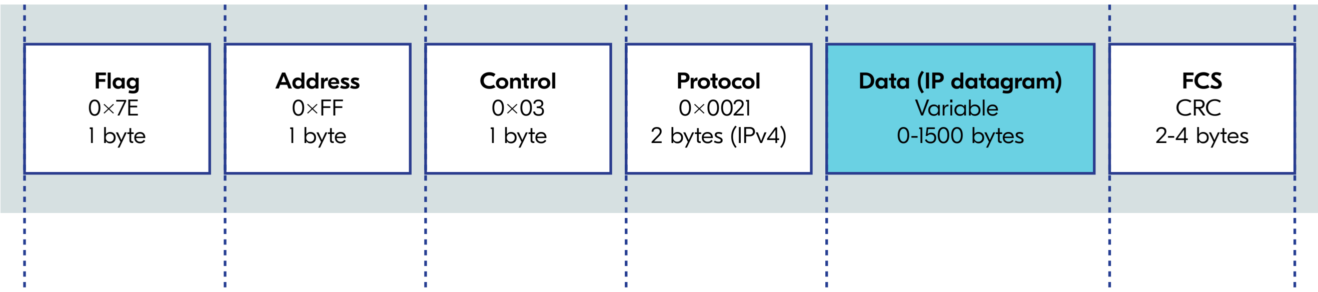

Figure 2. PPP frame

This limitation is addressed by the CMUX (Multiplexing Control Protocol) as specified in 3GPP TS 27.010. CMUX enables multiple logical channels, called Data Link Connection Identifiers (DLCIs), to be multiplexed over a single physical UART connection. This allows simultaneous operation of different communication channels:

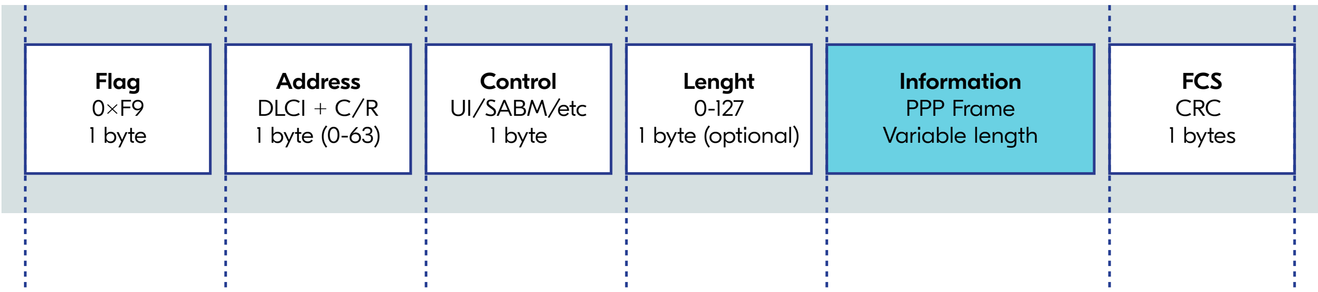

Figure 3: CMUX frame

Both PPP and CMUX frame structures follow the High-Level Data Link Control (HDLC) specification, which provides a standardized frame format for point-to-point and multipoint data links as illustrated in figures 2 and 3. The frame structure consists of flag delimiters, address/control fields, variable-length data payload, and frame check sequence (FCS) for error detection. The
primary difference is that CMUX uses flag value 0×F9 instead of PPP's 0×7E, and includes a DLCI field in the address byte to identify the logical channel, enabling multiple virtual connections over a single physical link.

- **DLCI 0:** Reserved for control channel
- **DLCI 1:** Typically used for AT commands
- **DLCI 2:** Typically used for PPP data connection
- **DLCI 3-63:** Available for additional channels (e.g., GPS, secondary data connections)

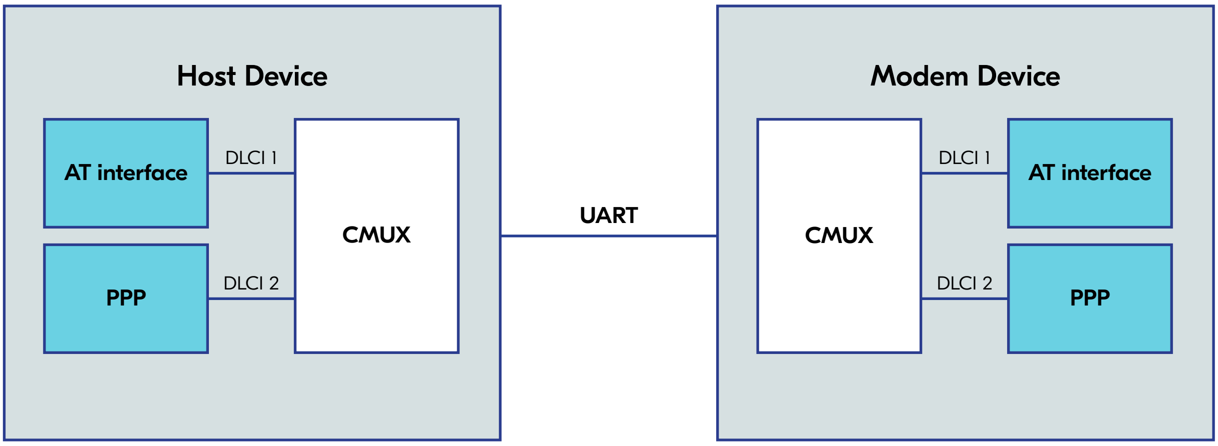

Figure 4: Multiplexing a serial pipe

The ability to maintain an active AT command interface (DLCI 1) while the PPP connection operates on a separate DLCI channel is crucial for embedded and roaming devices. These devices must continuously monitor network conditions and respond to 3GPP-defined notifications such as:

- Network registration status changes
- Signal quality indicators
- Network rejection or de-registration events
- Emergency service availability notifications
- Location area updates

For example, when a device roams between network cells or experiences temporary network loss, the modem generates unsolicited result codes (URCs) such as +CEREG , +CSQ , or +CGEV that are delivered over the AT command channel. Without CMUX, these critical notifications would be lost once PPP is established, potentially leaving the device unaware of network state changes until the PPP connection fails.

CMUX Multiplexing Example

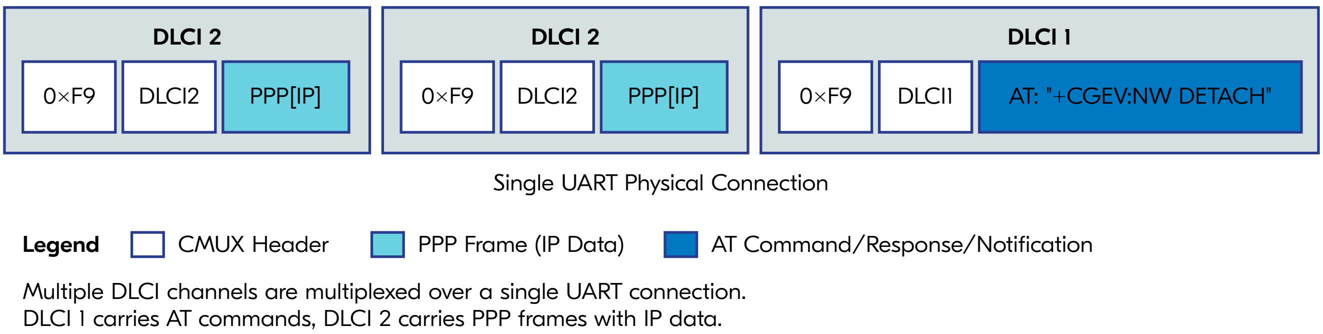

Figure 5: CMUX Multiplexing Example

LTE Power saving modes PSM and eDRX

LTE networks provide two complementary power saving mechanisms designed to extend battery life for IoT devices: Power Saving Mode (PSM) and extended Discontinuous Reception (eDRX). These mechanisms allow devices to minimize power consumption during idle periods while maintaining network connectivity.

Power Saving Mode (PSM) enables devices to enter a deep sleep state while remaining registered with the network. During PSM, the device is unreachable by the network but maintains its registration, eliminating the need for re-registration upon wake-up. The device can remain in PSM for extended periods, waking only when it needs to transmit data or when the active timer expires.

The nRF9151 modem achieves exceptionally low power consumption of approximately 2 µA while in PSM mode, making it ideal for battery-powered IoT applications that require long operational lifetimes. The nRF9151 also has a significant advantage over competitors as all RAM is retained during PSM, which allows exceptionally fast wake-up and network attach.

Extended Discontinuous Reception (eDRX) complements PSM by allowing devices to reduce the frequency of paging channel monitoring. Instead of continuously monitoring for incoming data, the device wakes up periodically at configurable intervals ranging from 5.12 seconds to 174 minutes (link: https://docs.nordicsemi.com/bundle/ref_at_commands_nrf91x1/page/REF/at_commands/nw_service/cedrxs_set.html) to check for paging messages. This significantly reduces power consumption compared to continuous reception while maintaining the ability to rreceive incoming data within the eDRX paging time window.

When combined, PSM and eDRX provide a powerful mechanism for ultra-low power operation. The device can enter PSM after a configurable active timer, and when it needs to be reachable, it can use eDRX to minimize paging overhead.

However, in traditional dial-up modem configurations using PPP over UART, the UART interface remains the dominant power consumer. The UART peripheral requires high-speed clocks to operate at typical baud rates (e.g., 115200 baud), and these clocks remain active as long as the UART receiver is enabled, even when no data is being transmitted or received. This continuous clock operation prevents the system from entering deeper sleep states, resulting in power consumption in the hundreds of microamps range even when the cellular modem itself is in PSM mode.

CMUX Power Saving

The 3GPP TS 27.010 specifies a power saving mechanism for CMUX. The power saving mechanism is covered in the following sections on the specification:

- 5.2.5 Inter-frame Fill
- 5.4.6.3.2 Power Saving Control (PSC) message
- 5.4.7 Power Control and Wake-up Mechanisms

The power saving mechanism allows devices to switch UART into a low power state when there is no data to be sent or received on any DLCI channel. When the CMUX module detects idle conditions, it will enter the power saving state after a configurable timeout.

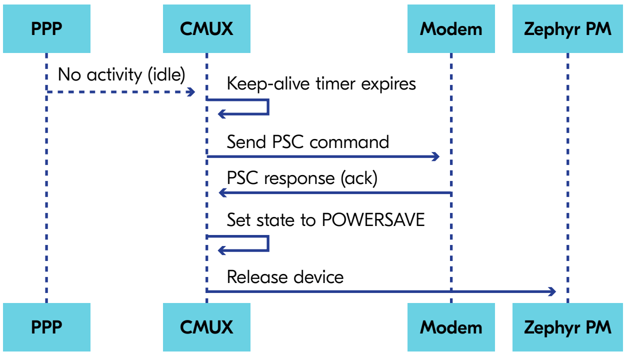

Figure 6: CMUX idle sequence

In the idle state, the CMUX module will send a Power Saving Control (PSC) message to the modem, requesting it to enter a low power state. The CMUX standard allows devices to enter low power state but does not require them to do so—the actual power management implementation depends on the system's runtime power management capabilities.

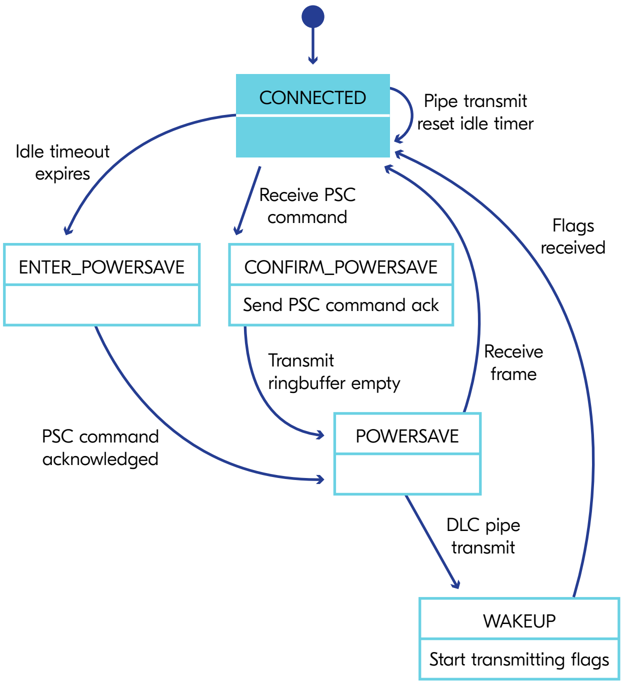

Figure 7: CMUX state diagram

When data needs to be sent or received on any DLCI channel, the CMUX module will exit the idle state and wake the remote end up by sending repeated flag characters (0×F9) until it receives a flag character as a response. These state transitions are illustrated in figure 7. Either end of the CMUX channel can initiate the wake up procedure.

CMUX Wake-up Sequence

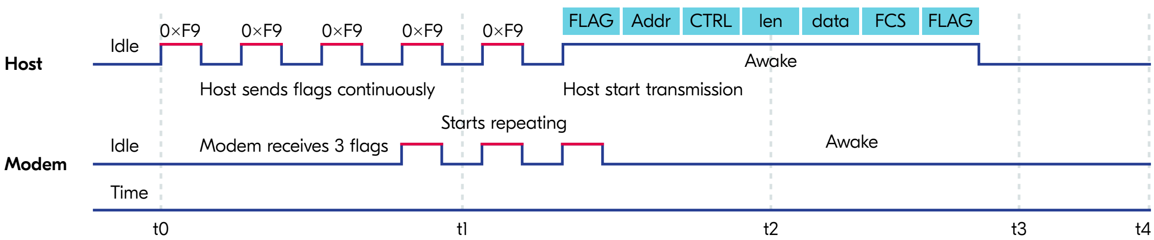

Figure 8: CMUX Wake-up sequence

The wake-up sequence consists of the peer repeatedly sending flag characters (0×F9) on the UART. The remote peer, upon detecting these flags, responds with its own flag character to indicate it is awake and ready to receive data. This handshake ensures both ends of the communication link are synchronized before data transmission resumes. As the flag characters carry no information, there is no issue of losing data when waking up. When the peer that initiates the wake-up receives a flag character, it can continue by sending a valid frame.

The power saving mode of CMUX is completely transparent to upper layer protocols, including the PPP stack. The PPP implementation operates on a DLCI channel as its serial pipe interface and requires no modifications or special handling to benefit from CMUX power saving. When the PPP connection is idle—which naturally occurs when the device enters eDRX or PSM power saving states at the cellular network level—there is no data traffic on the PPP pipe. This idle condition triggers the CMUX power saving mechanism, allowing both devices to enter low power states without breaking the PPP connection or requiring application-level intervention.

Handling the UART power state

While CMUX provides the protocol-level mechanism for entering low power states, the actual power management of the UART hardware is handled separately through runtime power management. When CMUX enters the idle state and sends a Power Saving Control message, the system can take advantage of this by powering down the UART device entirely.

When the UART is powered down, it releases its high-speed clocks, which are then automatically stopped by the system's power management. This achieves the lowest possible power consumption, as the UART is one of the primary power consumers in dial-up modem configurations.

The UART can be powered down when CMUX allows the low power state—specifically, when there is no data to be sent or received on any DLCI channel and CMUX has entered its idle state. At this point, runtime power management can power down the UART device, reducing power consumption from hundreds of microamps to just a few microamps.

However, when the UART is completely powered down, special mechanisms are required to wake it up for incoming data. This requires the use of GPIO signals: DTR (Data Terminal Ready) and RING.

Modem waking the host: When incoming data arrives at the modem while the UART is powered down, the modem uses the RING signal to wake up the host processor. The RING signal is handled by the modem driver on the host, which opens the UART device and asserts the DTR line, signaling the remote end that the UART is powered on. This allows the CMUX module to wake up the modem and process incoming data.

UART Wake-up Sequence

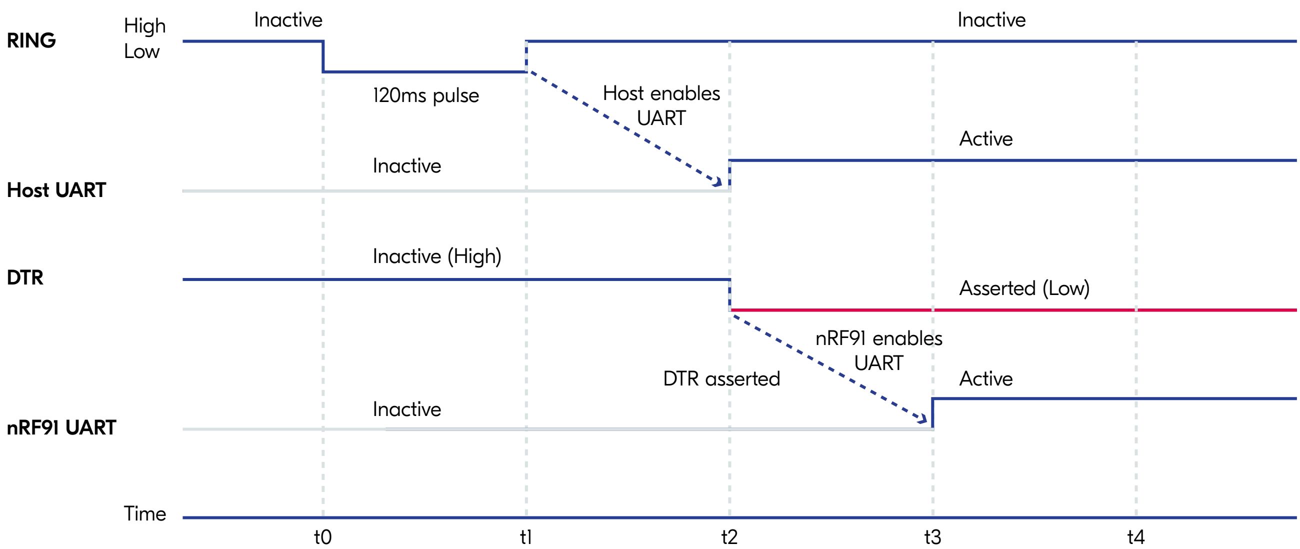

Figure 9: Modem waking up the host

The timing diagram above illustrates the wake-up sequence when the UART is powered down. The RING signal triggers the host to power up its UART and assert DTR, which in turn signals the modem to power up its UART interface. This coordinated wake-up ensures both ends of the communication link are ready before data transmission begins.

Host waking the modem: When the host processor needs to send data, it first powers up the UART device, then asserts the DTR signal to indicate that the UART is active. The modem detects the DTR assertion and powers up its UART interface, allowing communication to resume.

Host Waking Modem via DTR

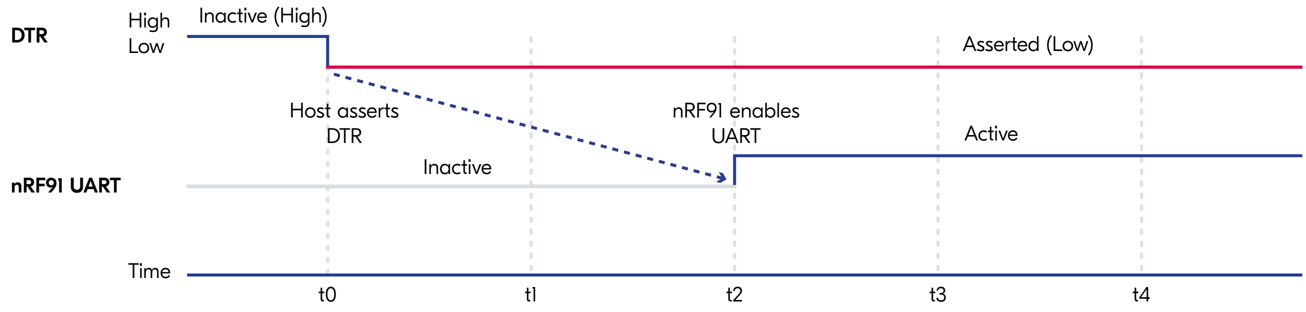

Figure 10: Host waking up the modem

When UART have been woken up, the CMUX wake-up sequence continues as explained in the previous section.

Evaluation of the CMUX power saving

This section presents an evaluation of the CMUX power saving mechanism's impact on power consumption using the nRF9151 modem. The evaluation setup consists of an nRF9151 running the Serial Modem application, with an nRF54L15 serving as the host MCU.

The Devicetree for nRF54L15 is constructed as follows and Zephyr's runtime power management is used to allow UART to be powered down.

```
&uart1 {
  current-speed = <115200>;
  hw-flow-control;
  zephyr,pm-device-runtime-auto;
  label = "modem_uart";
  status = "okay";
  pinctrl-0 = <&uart1_default_alt>;
  pinctrl-1 = <&uart1_sleep_pull>;
  pinctrl-names = "default", "sleep";

  modem: modem {
    compatible = "nordic,nrf91-slm";
    status = "okay";
    mdm-ring-gpios = <&gpio1 23 (GPIO_PULL_UP |
    GPIO_ACTIVE_LOW)>;
    mdm-dtr-gpios = <&gpio1 22 GPIO_ACTIVE_LOW>;
    zephyr,pm-device-runtime-auto;
    cmux-enable-runtime-power-save;
    cmux-close-pipe-on-power-save;
    cmux-idle-timeout-ms = <500>;
  };
};
```

Zephyr's runtime power management is enabled:

```
CONFIG_MODEM=y
CONFIG_MODEM_CELLULAR=y
CONFIG_PM_DEVICE=y
CONFIG_PM_DEVICE_RUNTIME=y
```

The modem driver configuration was modified so that the nRF91 AT initialization script requests PSM+eDRX mode during modem startup:

```
AT+CPSMS=1,,,"00101100","00000111"
AT+CEDRXS=2,4,"0000"
AT%XPTW=4,"0000"
```

The example above requests active time of 14s, eDRX paging window 1.28s and cycle 5.12s.

The nRF9151 Devicetree is configured as follows

```
&uart2 {
  compatible = "nordic,nrf-uarte";
  current-speed = <115200>;
  hw-flow-control;
  status = "okay";

  dtr_uart2: dtr-uart {
    compatible = "nordic,dtr-uart";
    dtr-gpios = <&gpio0 31 (GPIO_PULL_UP | GPIO_
    ACTIVE_LOW)>;
    ri-gpios = <&gpio0 30 GPIO_ACTIVE_LOW>;
    status = "okay";
  };
};

&gpio0 {
  /* Use Pin sense mechanism for DTR (31) pin to save
  power */
  sense-edge-mask = <BIT(31)>;
};
```

It is worth mentioning that to achieve lowest power consumption, the GPIO pins that are used for waking up the device, DTR in this case, should be configured to use Pin sense mechanism, instead of edge triggered GPIOTE event.

Effects on power usage of nRF9151

The test case used here allows devices to idle until PSM mode is reached. Then one UDP packet is sent, expecting one response—a DNS query in this case. The nRF9151 wakes up from PSM mode automatically when there is data to be sent.

When the runtime power management is not used, the measured power usage on nRF9151 while being in PSM mode is around 450 µA. This power usage is mainly caused by UART reception being active as it keeps clocks running.

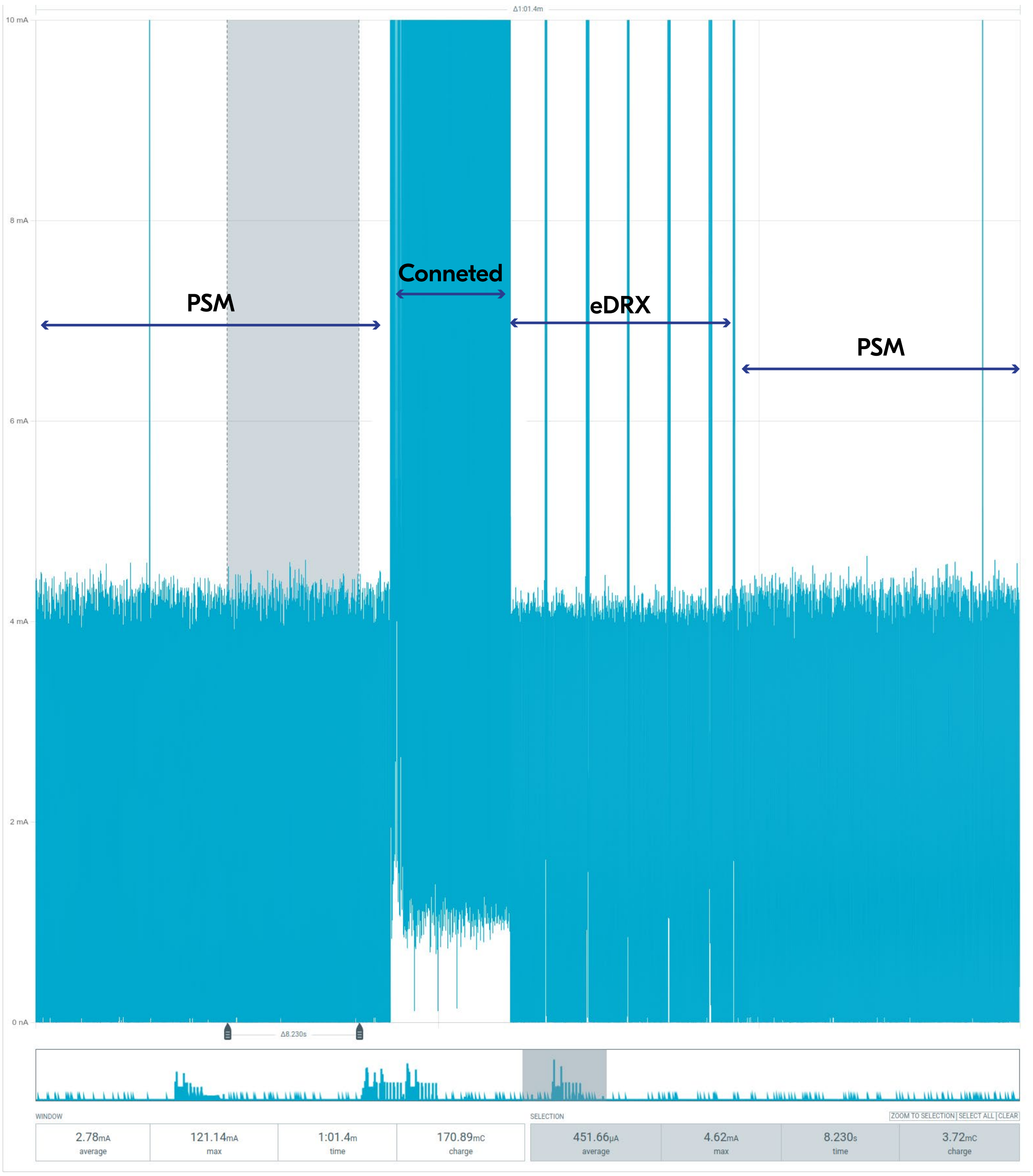

Figure 11: Power measurements on PSM+eDRX without UART power management

When the CMUX power saving mode is enabled and runtime power management is allowed to shut down the UART, the power usage drops to approximately 2 µA.

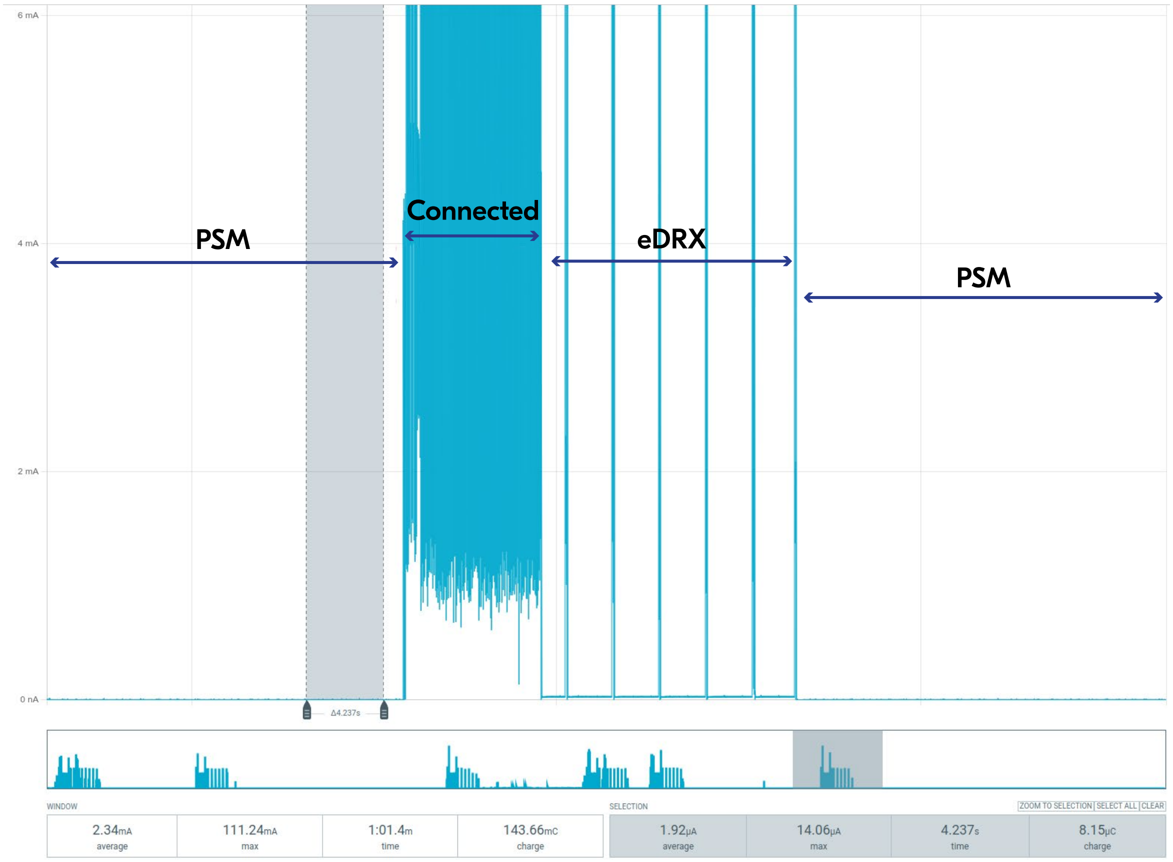

Figure 12: Power measurements of PSM+eDRX with active UART power management

Conclusion

The evaluation demonstrates that using the nRF9151 modem with CMUX power saving enabled, power consumption during PSM mode is around 2 µA while maintaining an active PPP+CMUX connection. This achievement is made possible by the CMUX power saving mechanism, which allows the UART interface to be completely shut down during idle periods.

The measured power consumption reduction is substantial: PSM power usage decreases from approximately 400-450 µA down to approximately 2 µA—a reduction of over 99%. This dramatic improvement in power efficiency makes it unnecessary to tear down the IP connectivity during extended idle periods, enabling battery-powered IoT devices to maintain network connectivity for many hours or even days while consuming minimal power.