s14x_nrf5x migration document

Introduction to the s140_nrf52840 migration document

About the document

This document describes how to migrate to new versions of the s140 SoftDevices. The s140_nrf52840 release notes should be read in conjunction with this document.

For each version, we have the following sections:

- "Required changes" describes how an application would have used the previous version of the SoftDevice and how it must now use this version for the given change.
- "New functionality" describes how to use new features and functionality offered by this version of the SoftDevice. **Note:** Not all new functionality may be covered; the release notes will contain a full list of new features and functionality.

Each section describes how to migrate to a given version from the previous version. If you are migrating to the current version from the previous version, follow the instructions in that section. To migrate between versions that are more than one version apart, follow the migration steps for all intermediate versions in order.

Example: To migrate from version 5.0.0 to version 5.2.0, first follow the instructions to migrate to 5.1.0 from 5.0.0, then follow the instructions to migrate to 5.2.0 from 5.1.0.

s140_nrf52840_5.0.0-1.alpha

This section describes how to migrate to s140_nrf52840_5.0.0-1.alpha from s132_nrf52_3.0.0. This SoftDevice is designed to take advantage of the new features of the nrf52840 chip.

Required changes

SoftDevice flash and RAM usage

The size of the SoftDevice has changed and therefore a change to the application project file is required.

For Keil this means:

1. Go into the properties of the project and find the Target tab
2. Change IROM1 Start to `0x20000`.

If the project uses a scatter file or linker script instead, then these must be updated accordingly.

The RAM usage of SoftDevice has also changed. `sd_ble_enable()` should be used to find the APP_RAM_BASE for a particular configuration.

Renamed defines

Some defines have been renamed to make the API more consistent. Any code using these defines has to be updated with the new names:

- `GATT_MTU_SIZE_DEFAULT` renamed to `BLE_GATT_MTU_SIZE_DEFAULT`
- `BLE_EVTS_LEN_MAX` renamed to `BLE_EVT_LEN_MAX`
- `BLE_EVTS_PTR_ALIGNMENT` renamed to `BLE_EVT_PTR_ALIGNMENT`

New functionality

Multiple PHYs

The SoftDevice introduces support for using multiple PHYs to adapt the speed and reliability of data transmission to the channel capacity. For higher throughput, a 2 Mbps PHY is supported. For higher reliability, a 125kbps Coded PHY is supported.

API updates

- A new GAP option, `BLE_GAP_OPT_PREFERRED_PHYS_SET`, has been added to indicate to the controller about which PHYs the controller shall prefer so it can respond to any requests to update PHYs by peers.
- A new SV call, `sd_ble_gap_phy_request()`, has been added to request the controller to attempt to change to a new PHY.
- A new event, `BLE_GAP_EVT_PHY_UPDATE`, has been added to indicate that the PHY of a connection has changed or that a local initiated PHY update procedure has finished.

Usage

Example pseudo code for setting the preferred PHYs for new connections
Note: This will only have an effect if the peer device initiates the procedure to change the PHY. The stack will not initiate a PHY Update procedure autonomously.

```
ble_opt_t opts;
opts.gap_opt.preferred_phys.tx_phys = BLE_GAP_PHY_1MBPS | BLE_GAP_PHY_2MBPS;
opts.gap_opt.preferred_phys.rx_phys = BLE_GAP_PHY_1MBPS | BLE_GAP_PHY_2MBPS;
TEST_SD_UTIL_NRF_SUCCESS_OR_ASSERT(sd_ble_opt_set(BLE_GAP_OPT_PREFERRED_PHYS_SET,
&opts) );

[ Advertise and connect / Scan and connect ]
```

Request the controller to attempt to change to a new PHY for an established connection:

```
ble_gap_phys_t phys = {BLE_GAP_PHY_CODED, BLE_GAP_PHY_CODED};
sd_ble_gap_phy_request(conn_handle, &phys);
```

Handle PHY Update event:

```
/* Handle the event */
case BLE_GAP_EVT_PHY_UPDATE:
 if (ble_event.evt.gap_evt.params.phy_update.status == BLE_HCI_STATUS_CODE_SUCCESS)
 {
   // The PHY was changed (after either the application or the peer requested it)
   // ble_event.evt.gap_evt.params.phy_update.tx_phy and
 ble_event.evt.gap_evt.params.phy_update.rx_phy contain the new PHYs
 }
 else
 {
   // A PHY update was requested which could not be performed successfully
 }
```

Higher TX power on nRF52840

The SoftDevice now supports configuring higher TX power to be used with nRF52840.

The following additional values are supported by the `sd_ble_gap_tx_power_set()` SV-call +2dBm, +5dBm, +6dBm, +7dBm, +8dBm, +9dBm.

These power levels can be used in the same way the existing power levels are used in the s132_nrf52_3.0.0 SoftDevice.