Connection Parameter Update

The macro `IS_ACTIVE(CONN_PARAM_UPDATE) / ACTIVATE_CONN_PARAM_UPDATE` enables the firmware to handle connection parameter updates as provided by the Nordic SoftDevice in the form of a SoftDevice call and two events.

There are two major variations of the connection parameter update. When initiated on a device in the *central* role, the parameters are updated and both the peripheral and the central are informed via an `BLE_GAP_EVT_CONN_PARAM_UPDATE / GapConnParamUpdateEvent` about the new parameters. When initiated on the a device in the *peripheral* role, the parameters are not immediately updated, but the partner in the *central* role receives a `BLE_GAP_EVT_CONN_PARAM_UPDATE_REQUEST / GapConnParamUpdateRequestEvent` which can be either

- *accepted* on the central by updating the connection parameters to the requested ones (they are available from the event),
- *rejected* on the central by calling `FruityHal::BleGapRejectConnectionParamsUpdate`,
- *rejected* via timeout on the central by doing nothing.

After the request was handled both the central and peripheral are informed about the new (or old) parameters via the `BLE_GAP_EVT_CONN_PARAM_UPDATE / GapConnParamUpdateEvent`.

Long-Term Connection Interval

The first user of the connection parameter update is the `prod_ruuvi_weather_nrf52` featureset, which utilitzes the long-term connection interval to drop power consumption a bit after the connections have been established. This is also enabled through the `IS_ACTIVE(CONN_PARAM_UPDATE)` and thusly it is not automatically activated for other featuresets.

There are two new configuration member variables in the `Conf` class

- `Conf::GetInstance().meshMinLongTermConnectionInterval`
- `Conf::GetInstance().meshMaxLongTermConnectionInterval`

which are of similar purpose as the initial connection interval values (`Conf::GetInstance().meshMinConnectionInterval`, ...). They are initialized to the same value as the initial connection interval, which will not update the connection parameters automatically.

If they are set to some larger value, after a mesh connection has an age over `Conf::meshConnectionLongTermAgeDs` deciseconds, the firmware will try to update the connection interval to the specified long-term values. It only does this once, such that requests are not repeated forever.

Updates initiated on a peripheral are additionally delayed by `Conf::meshConnectionLongTermAgePeripheralPenaltyDs` deciseconds to avoid that two updates are started simultaneously by both partners (this does not produce errors, but it results in hard-to-follow logs). This delay also results in the central's long-term parameters being used, if both parties have different ones.

SoftDevice Documentation

- SoftDevice function for updating the connection parameters.
  (https://infocenter.nordicsemi.com/topic/com.nordic.infocenter.s132.api.v5.0.0/group___b_l_e___g_a_p___f_u_n_c_t_i_o_n_s.html#gaf89b434841998ab384e0612dca9e12f8)
- Message sequence chart for update on central
  (https://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.s132.api.v5.0.0%2Fgroup___b_l_e___g_a_p___c_e_n_t_r_a_l___c_p_u___m_s_c.html)
  .
- Message sequence chart for update on peripheral
  (https://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.s132.api.v5.0.0%2Fgroup___b_l_e___g_a_p___c_p_u___m_s_c.html)
  .