EMB OBTAINING SLEEP CLOCK ACCURACY

Product Family: **Example: EM9305**

Part Number: EM9305

Keywords: BLE, Firmware, Peripheral, Sleep Clock Accuracy, ppm, connection indication

Contents

1. OVERVIEW 1
2. CONNECTION INDICATION 1
3. LLCP CLOCK ACCURACY EXCHANGE 1
4. CODE SNIPPETS 2
4.1 Reading SCA from connection event 2
4.2 Checking SCA update support 3
4.3 Reading SCA during SCA exchange procedure 4

1. OVERVIEW

This document provides an explanation and examples of the two most common methods to obtain peer sleep clock accuracy (SCA) from a BLE application. There are four existing methods the application can get sleep clock accuracy from a peer device:

1. Connection Indication
2. Link Layer Control Protocol (LLCP) Clock Accuracy Exchange
3. Synchronizing to periodic advertising
4. Periodic advertising sync transfer (PAST)

This document focuses on the first two ways because it is assumed the example application is operating in the peripheral role. The other two ways follow a similar pattern of initiating a procedure and obtaining the sleep clock accuracy value from an asynchronous event.

2. CONNECTION INDICATION

This is the easiest way to get the sleep clock accuracy from the peer device. When a connection is formed, the initiator clock accuracy is included in the connection complete event. The Code snippet 4-1 below shows how to modify the `nvm_emb_fit` SDK example application to get the sleep clock accuracy from the connection indication.

3. LLCP CLOCK ACCURACY EXCHANGE

To get the SCA from the LLCP clock accuracy exchange, both the connecting or connected devices must support the BLE "Sleep Clock Accuracy Updates" feature. This can be determined during the LLCP feature exchange procedure and checked at the application level. The Code snippet 4-2 shows how the application can check that the peer supports the sleep clock accuracy updates feature. The application is assumed to be the `nvm_emb_fit` example.

If the application is using the application framework (AF), the read remote features procedure is automatically started when a connection is opened. If the application is not using the AF (i.e. relying only on the device manager DM), reading the remote features can be achieved by calling DmReadRemoteFeatures after a connection is opened.

Similarly, if the application is using the AF, and sleep clock accuracy updates are supported by both devices, the clock accuracy exchange procedure is automatically initiated by the AF when the connection is opened. If it is not automatically started (app is DM only), then it can be started by calling DmConnRequestPeerSca only after the application receives the `DM_CONN_OPEN_IND` event.

Note: If one or more of the devices do not support sleep clock accuracy updates, the `DM_REQ_PEER_SCA_IND` event shall not be received after calling DmConnRequestPeerSca.

If both devices support sleep clock accuracy updates, and clock accuracy exchange procedure completes, the application will receive the `DM_REQ_PEER_SCA_IND` event. Code snippet 4-3 shows how to modify the `nvm_emb_fit` example to obtain the sleep clock accuracy value from that event.

4. CODE SNIPPETS

4.1 READING SCA FROM CONNECTION EVENT

```
static void APP_FitProcMsg(fitMsg_t *pMsg)
{
  …
  switch (pMsg->hdr.event)
  {
    …
    case DM_CONN_OPEN_IND:
      HrpsProcMsg(&pMsg->hdr);
      BasProcMsg(&pMsg->hdr);
      RSCPS_ProcMsg(&pMsg->hdr);
      // Read a value representative of the sleep clock accuracy ppm
      uint8_t peerSCA = ((hciLeConnCmplEvt_t *)pMsg)->clockAccuracy;
      // to associate SCA with connection
      dmConnId_t connId = (dmConnId_t)pMsg->hdr.param;
    …
  }
  …
}
```

Code snippet 4-1: Obtaining SCA from a Connection Indication event

In this code snippet, the `peerSCA` variable contains a value ranging from 0 to 7 defined by the Bluetooth LE core specification. This value indicates a ppm range related to the clock accuracy. These ranges are given in Table 4-1.

SCA	Central SCA or peripheral SCA
0	251 ppm to 500 ppm
1	151 ppm to 250 ppm
2	101 ppm to 150 ppm
3	76 ppm to 100 ppm
4	51 ppm to 75 ppm
5	31 ppm to 50 ppm
6	21 ppm to 30 ppm
7	0 ppm to 20 ppm

Table 4-1: SCA value to PPM range

4.3 CHECKING SCA UPDATE SUPPORT

```
#include "ll_api.h" // Provides access to the LL_FEAT_SCA_UPDATE constant.


static void APP_FitProcMsg(fitMsg_t *pMsg)
{
  switch (pMsg->hdr.event)
  {
    …
    case DM_REMOTE_FEATURES_IND:
    {
      uint64_t peerFeatures;
      BYTES_TO_UINT64(peerFeatures, ((hciLeReadRemoteFeatCmplEvt_t *)pMsg)-
>features)
      if (peerFeatures & LL_FEAT_SCA_UPDATE)
      {
        // Sleep clock accuracy updates supported.
      }
      else
      {
        // Sleep clock accuracy updates not supported.
      }
      break;
    }
    …
  }
  …
```

Code snippet 4-2: Sleep Clock Accuracy Updates Feature Check

4.5 READING SCA DURING SCA EXCHANGE PROCEDURE

```
…
static void APP_FitProcMsg(fitMsg_t *pMsg)
{
   switch (pMsg->hdr.event)
   {
      case DM_REQ_PEER_SCA_IND:
         uint8_t peerSCA = ((HciLeReqPeerScaCmplEvt_t *)pMsg)->peerSca;
         // to associate SCA with connection
         dmConnId_t connId = (dmConnId_t)pMsg->hdr.param;
      break;
      …
   }
   …
}
```

Code snippet 4-3: Obtaining SCA from LLCP Clock Accuracy Exchange