APPLICATION NOTE

Atmel AVR2152: RTB Evaluation Application Software User's Guide

8-bit Atmel Microcontrollers

Features

- Evaluation of AT86RF233 radio transceiver phase difference measurement unit
- Design based on Atmel® AT86RF233 radio transceiver supporting IEEE® 802.15.4, ZigBee®, 6LoWPAN, RF4CE, SP100, *Wireless*HART® and ISM applications
- Software Architecture and API introduction

Introduction

The document describes the basic software architecture and Application Programming Interface (API) of the Ranging Toolbox (RTB) and the related evaluation application software package and how to include the RTB library into an application.

Figure 1. REB233SMAD based hardware platform.

Table of Contents

1. Disclaimer

Typical values contained in this application note are based on simulations and testing of individual examples.

Any information about third party materials or parts was included into this document for convenience. The vendor may have changed the information that has been published. Check the individual vendor information for latest changes.

Note: The RTB library included in this firmware package is provided for demonstration purposes only and adapted to this specific hardware configuration and application scenario. In case of a different hardware configuration and application scenario a customized firmware port is needed.

2. Overview

The RTB Evaluation Application (RTB Eval App), operated on a REB233SMAD-EK [6] hardware platform, is targeted for evaluating the AT86RF233 [4] phase difference measurement unit (PMU). The setup provides an ideal platform to:

- Demonstrate AT86RF233 phase difference measurement technology
- Evaluate the AT86RF233 radio transceiver performance
- Test the radio transceiver hardware support for the IEEE802.15.4 standard
- Test the enhanced radio transceiver feature set

3. Software Architecture

The RTB Eval App introduces the RTB library which provides the basic software functionality and API to the applications to utilize the ranging hardware and initiate ranging measurements.

3.1 General MAC architecture

The RTB uses IEEE 802.15.4 MAC [2] functionality as the basic architecture which is shown in Figure 3-1.

The MAC software implementation is modular, generally allowing different hardware to be used for the MAC applications. The microcontroller (MCU) and board are interfaced using the Platform Abstraction Layer (PAL). The radio transceiver (TRX) is interfaced using the Transceiver Abstraction Layer (TAL).

Note the RTB Eval App supports the REB233SMAD-EK hardware platform only.

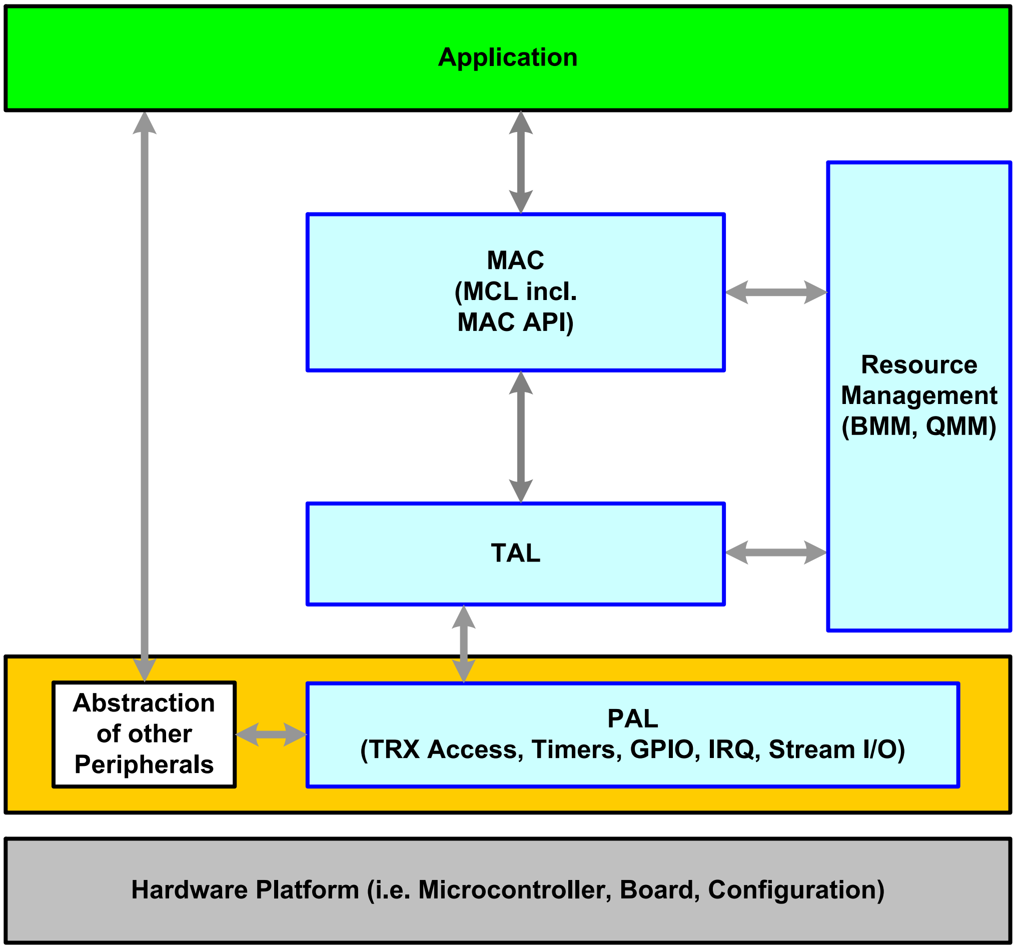
In the regular MAC architecture (without the RTB) the application interacts with the MAC Core layer (MCL) via the MAC-API, or directly accesses the PAL layer for platform oriented tasks. The MAC layer itself resides on top of the TAL and the PAL. Frames to be transmitted are handed over from the MAC to the TAL. Received frames from the hardware are handed over by the TAL to the MAC layer and are either processed directly within the MAC layer itself, or further uploaded to the application (for example, in case of a received data frame).

The application space is decoupled from the MAC stack space by means of two queues, residing inside the MAC-API:

- NHLE-MAC-Queue (NHLE: Next Higher Layer Entity)
- MAC-NHLE-Queue

Requests from the application (i.e. NHLE) are placed into the NHLE-MAC-Queue and processed by the MAC once the MAC is idle (i.e. the MAC is not busy processing other tasks). Confirmations (i.e. answers to requests) and indications (asynchronous events such as received frames) from the MAC are placed into the MAC-NHLE-Queue.

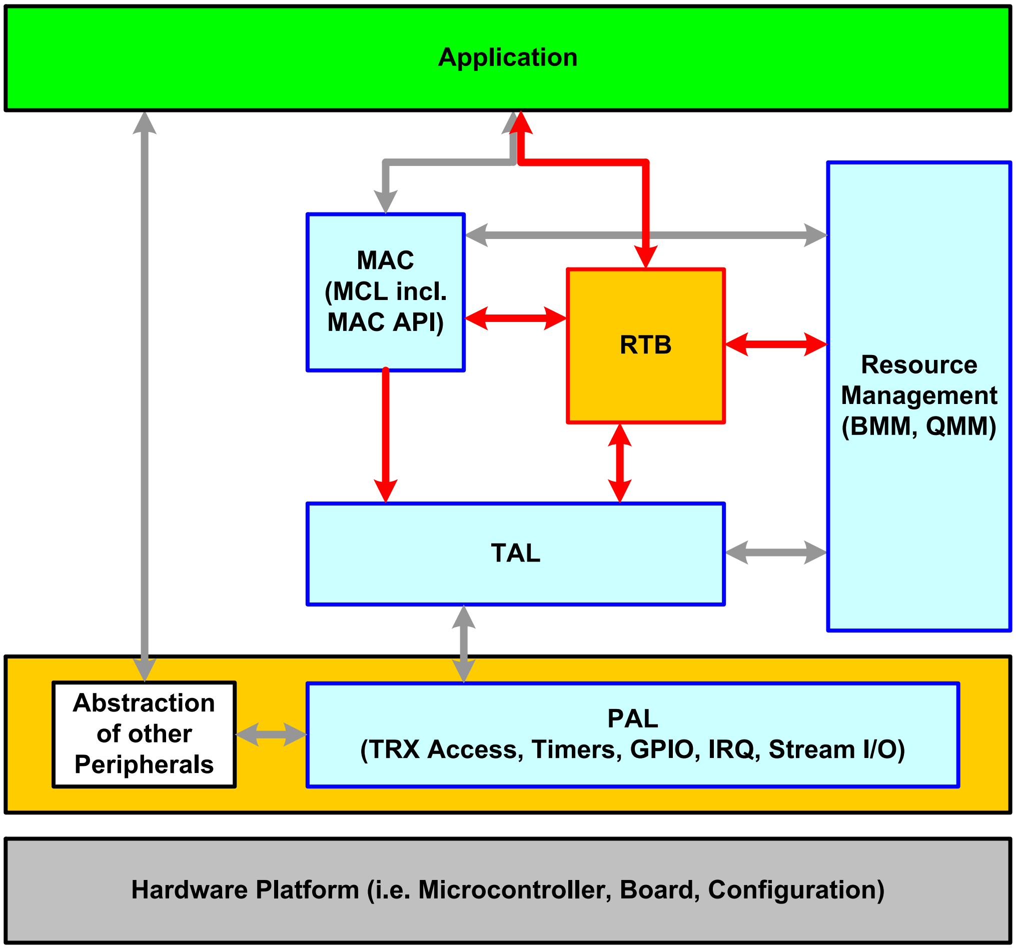
Figure 3-1. General simplified MAC software architecture.



3.2 Stack architecture with Ranging support

Figure 3-2 shows the software architecture used for the stack implementation including the RTB. The changes to the general MAC stack architecture are shown in red color.

Figure 3-2. Software architecture including RTB.



In the MAC architecture supporting ranging (including the RTB) the application still interacts with the MAC Core layer (MCL) via the MAC-API for MAC oriented tasks. Such tasks are data frame handling or MAC management related tasks, e.g. handling of associations, MAC/TAL PIB attribute handling, etc.

Additionally the stack provides the user an extended ranging API (i.e. the RTB-API, see Chapter 6) to allow the application for handling ranging measurement related tasks.

The application space is again decoupled from the MAC/RTB stack space by means of the same two queues as described in Section 3.1, i.e. the RTB does not require additional queues. This reduces the load for the application.

In order to support the quasi-parallel operation of MAC related standard tasks and ranging related tasks, the RTB is linked between the MAC and the TAL layer.

Data frames to be transmitted are still handed over from the MAC to the TAL, which initiates the proper frame transmission at the hardware. The same is valid for MAC management frames generated inside the MAC layer itself. Ranging measurement related frames to be transmitted are generated inside the RTB and also handed over to the TAL for further transmission.

Received frames from the hardware are still first processed by the TAL, but not directly handed over to the MAC layer. Since ranging procedures have very tight timing constraints, all received frames are first forwarded from the TAL to the RTB. The RTB then checks whether the received frame is a ranging frame (e.g. initiating a new ranging procedure from another ranging party) or belonging to an ongoing ranging procedure being initiated by the node itself. In this case the frame is handled within the RTB itself, and the corresponding response is generated within the RTB or the ranging related callback to the application is invoked.

If the received frame is not destined for the RTB, the frame is forwarded to the MAC Core layer, which normally acts without the RTB in place.

As stated, regular MAC tasks and ranging measurement tasks are handled quasi-parallel. This means, if the stack is idle, the application can either initiate a regular MAC operation or a ranging measurement, or the stack can properly act upon the reception of a non-ranging-related frame or a ranging frame. As soon as an operation is ongoing within the stack (either a regular MAC frame shall be transmitted or a ranging procedure is in progress) the stack enters a busy state. During this state no further requests from the application layer are handled or undesired received frames are not processed further.

As soon as the ongoing procedure has been finished, the stack is able to handle the next operation.

4. RTB Eval App Software Package Content

Once the RTB Eval App has been extracted into the proper place, e.g. `C:\Atmel`, the directory structure looks as follows:

```
├───Applications
│   ├───Helper_Files
│   └───RTB_Examples
│       ├───MACless_RTB_Demo_lib
│       │   ├───AT86RF233_ATXMEGA256A3_REB_8_1_CBB
│       │   │   └───GCC
│       │   ├───Inc
│       │   └───Src
│       ├───RTB_Eval_App_lib
│       │   ├───AT86RF233_ATXMEGA256A3_REB_8_1_CBB
│       │   │   └───GCC
│       │   ├───Inc
│       │   └───Src
│       ├───RTB_Eval_Tools
│       └───RTB_MAC_Star_Demo_lib
├───Build
│   └───Build_RTB
│       └───GCC
│           └───AVR
├───Doc
├───Include
├───MAC
├───PAL
├───Resources
├───RTB
│   ├───Inc
│   ├───RTB_Lib
│   │   └───AT86RF233_ATXMEGA256A3_REB_8_1_CBB
│   │       └───GCC
│   └───Src
└───TAL
```

These directories contain the following items (in alphabetical order):

- Applications:
  - The RTB Eval App provides a ranging evaluation application and a Python script to perform automated ranging measurements
  - Makefiles and AVR Studio® project files are available and can be used as quick start for new application development
  - Hex files to be downloaded onto available hardware are provided
- Build: This directory contains batch files that can be easily used to rebuild any desired application or all applications at once
- Doc: This directory contains the RTB reference manual in html format which can be started by double clicking file *RTB_readme.html* in the root directory of the RTB package
- Include: This directory contains header files that are of general interest both for applications and for all layers of the stack such as IEEE constants, data types, return values etc.
- MAC: This directory contains the MAC Core Layer (MCL) and the MAC-API
- PAL: This directory contains the Platform Abstraction Layer with subdirectories for the supported MCU family. It provides all required source and header files for the MCU and the supported board configurations

- Resources: This directory contains the buffer and queue management implementation used internally inside MCL and TAL. Also hooks for application usage are provided
- RTB: This directory contains the RTB library and related files
- TAL: This directory contains the Transceiver Abstraction Layer with subdirectories for the supported TRX providing specific implementations addressing the specific needs of the transceiver

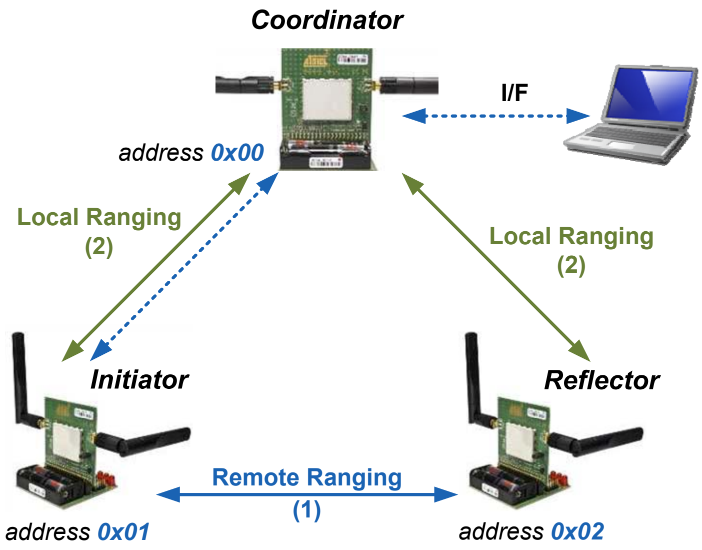
5. Introduction to Ranging Measurement

5.1 Remote and Local ranging

Two different ranging measurement principles are supported:

1. **Remote ranging**
   This method uses three wireless nodes referred to as *Coordinator*, *Initiator* and *Reflector*. The *Coordinator* remotely requests a ranging measurement between *Initiator* and *Reflector*. In this case the Initiator reports the results back to the *Coordinator*, which presents the ranging results to its own ranging application.
2. **Local Ranging**
   This method uses two wireless nodes only which are referred to as the *Initiator* and the *Reflector*. The *Initiator* controls the ranging procedure, performs the distance calculation and reports the results to its own ranging application.

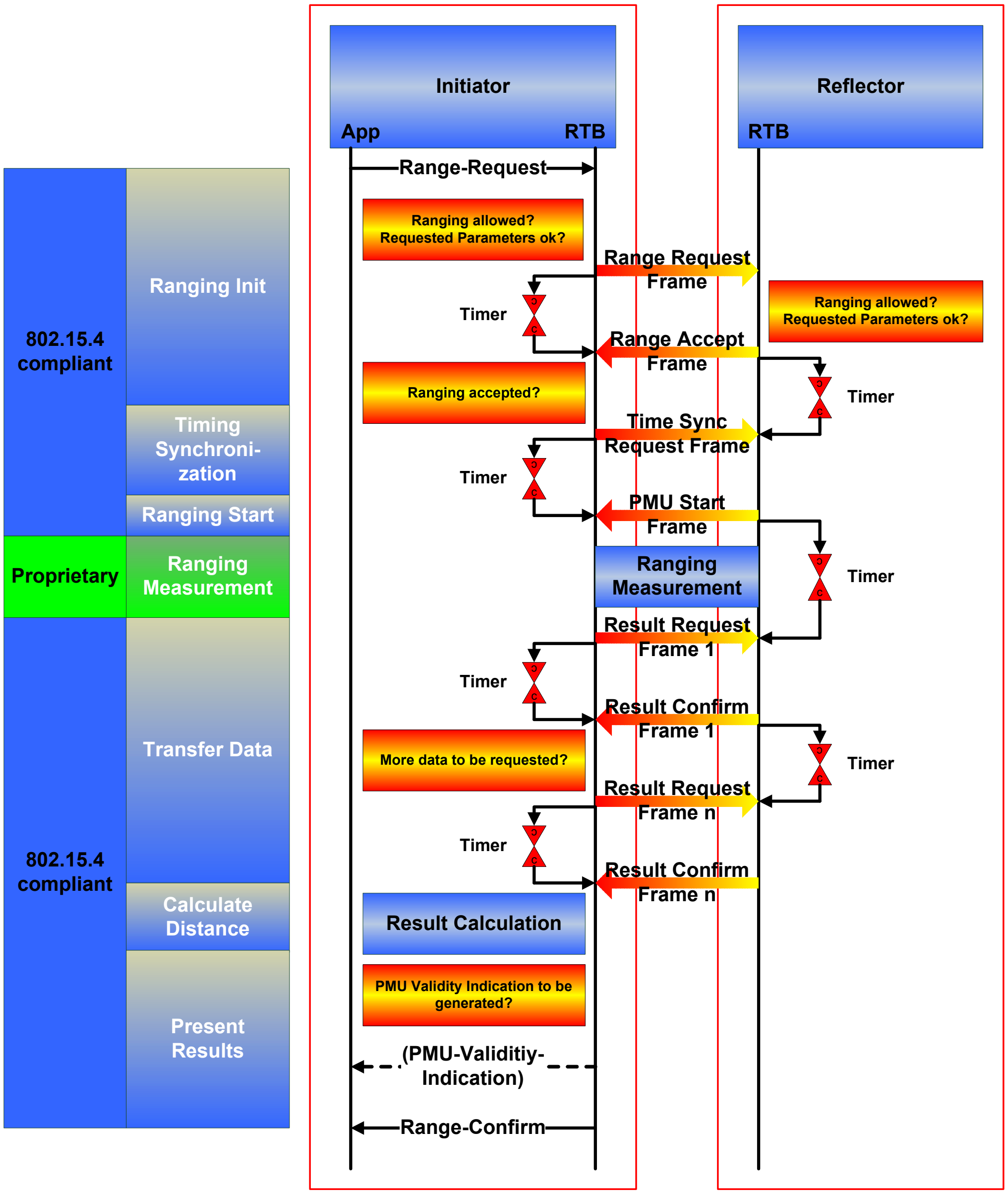
Figure 5-1. Remote and Local ranging.



5.2 Local ranging procedure

The entire local ranging procedure is shown in Figure 5-2.

Figure 5-2. Local ranging procedure.



5.2.1 Ranging Init phase

The regular local / peer-to-peer ranging procedure is initiated by the ranging application of the Initiator by means of a Range-Request message with the parameter Coordinator equal to zero indicating a local ranging measurement.

First the RTB checks whether a new ranging procedure is allowed at this time (for example, no other task is currently active within the MAC or RTB layer) and the requested parameters are valid. If ranging is not possible, a Range-Confirm message is generated by the RTB indicating the current error status to the application. Otherwise the ranging procedure is continued by transmitting the proper Range Request frame to the Reflector including its own ranging capabilities (see Section 5.6).

Once the Reflector has accepted the requested ranging procedure, it returns a Range Accept frame including the final ranging capabilities (see Section 5.6) and the status *RTB_SUCCESS* back to the Initiator. In case the Reflector does not accept the requested ranging procedure for any reason, it returns also Range Accept frame including the status *RTB_REJECT* and the corresponding reject reason.

5.2.2 Timing synchronization phase

During the Timing Synchronization Phase the Initiator sends a Time Sync Request frame to the Reflector and enters a special mode, where it expects the PMU (Phase Measurement Unit) Start frame to actually synchronize itself with the Reflector to start the ranging measurement. The Reflector also enters the special mode to synchronize with the Initiator.

5.2.3 Ranging start phase

Now the Reflector returns the PMU Start frame to the Initiator. At reception of this frame at the Initiator both nodes are fully synchronized with each other. All interrupts are disabled to prevent any disruption from other sources.

5.2.4 Ranging measurement phase

Both nodes perform the actual Ranging Measurement within the negotiated frequency band and collect the result data. This process cannot be interrupted by any other tasks. This phase creates proprietary non-802.15.4-compliant traffic. Once the core routine has finished, the global interrupts are enabled again, and the node is able to perform other high priority tasks again.

5.2.5 Data transfer phase

After the actual ranging measurement phase, the Initiator requests the collected ranging data from the Reflector by means of transmitting Result Request frames. The Reflector returns the requested ranging data using Result Confirm frames. The result exchange can incorporate one or several frame exchange sequences depending on the quantity of measured data.

5.2.6 Distance calculation phase

Once all required ranging data have arrived properly at the Initiator, distance and Distance Quality Factor (DQF) are calculated. In case of antenna diversity several distances and DQFs are calculated (one for each antenna) which require the calculation of weighted final distance results.

If the RTB PIB attribute *PMUVerboseLevel* (see Section 7.6) is greater than zero and at least one of the included ranging parties (Initiator and/or Reflector) utilizes antenna diversity, the Initiator generates a PMU-Validity-Indication message for its application to allow for the end user to check the frequency map for frequencies not suitable for the ranging algorithm, or apply further post-processing.

Next the Initiator generates a Range-Confirm message with the status code *RTB_SUCCESS* including the calculated distance and DQF. In case the PIB attribute *ProvideAntennaDivResults* (see Section 7.9) is true, and at least one of the involved parties utilizes antenna diversity, the Initiator includes the distance and DQF for each antenna pair (i.e. for each actual ranging measurement, see Section 5.4). The resulting Range-Confirm message allows the end user to perform further post-processing again using his own algorithms.

5.2.7 Timeout and error handling

In case a timeout occurs (e.g. due to a not properly received frame) the RTB finishes the ranging procedure and resets the RTB state machine. Additionally at the Initiator a Range-Confirm message indicating a timeout condition is generated (see Chapter 8).

In case any other error conditions occur (e.g. invalid parameters, ranging not allowed etc.) a Range-Confirm message including the corresponding error code is also generated (see Chapter 9).

5.2.8 General

During the entire ranging procedure both the Initiator and the Reflector cannot perform any other task. This means only one ranging measurement can be done at that time. Not any other task can be done by the MAC, i.e. the MAC cannot request the transmission of frames. Nevertheless frames destined to the node can still be received, but if they are not destined for the current ranging procedure, they are not further processed. Processing of these received frames can only be done once the current ranging procedure has been finished.

During the proprietary ranging core routine (the Ranging Measurement phase) the nodes a not able to receive any 802.15.4-compliant frame.

5.3 Remote ranging procedure

5.3.1 General

While the local ranging only involves the Initiator and the Reflector, the remote ranging procedure involves a third node – the Coordinator. The Coordinator actually controls a ranging procedure only, but may also act as Initiator and Reflector in local ranging procedures.

The Coordinator may request a ranging procedure between two other nodes, indicating which node shall be Initiator and Reflector. While a remote ranging is ongoing, the Coordinator is not blocked, but can rather perform any other tasks. This enables the Coordinator to act as ranging scheduler for the entire ranging network.

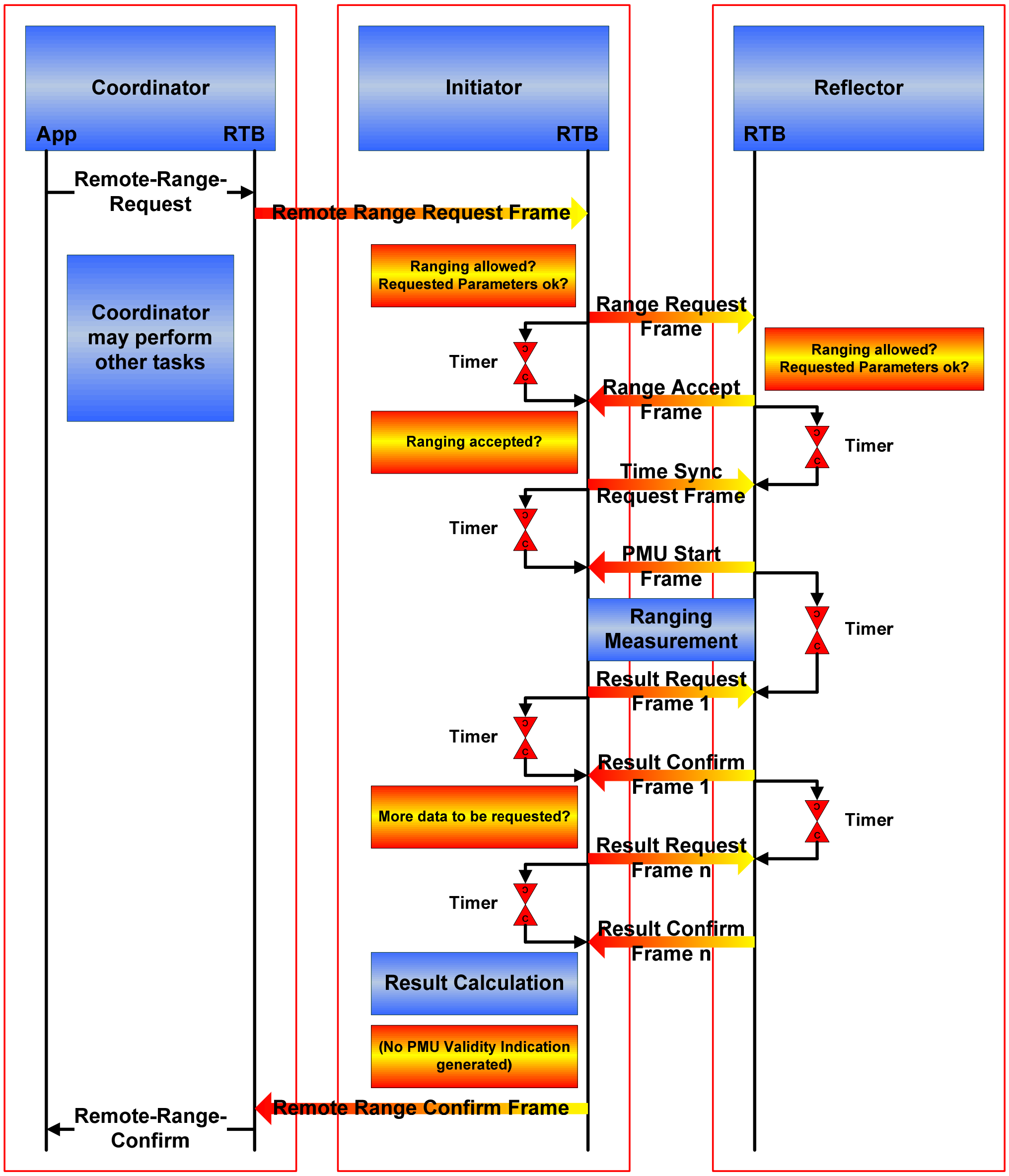
The entire remote ranging procedure is shown in Figure 5-3.

The remote ranging procedure is initiated by the ranging application of the Coordinator by means of a Remote-Range-Request message with the parameter Coordinator equal to two (indicating that the Coordinator uses its short address) or three (indicating that the Coordinator uses its long address).

The Coordinator RTB continues with the remote ranging procedure by transmitting the proper Remote Range Request frame to the Initiator. In case the Initiator does not accept this remote ranging request (due to invalid parameters or if ranging is currently not allowed), it returns a Remote-Range-Confirm message including the status code *RTB_REJECT* and the corresponding reject reason.

Otherwise (i.e. the remote ranging request is accepted by the Initiator) the Initiator continues as described in the local ranging procedure (see Section 5.2).

Figure 5-3. Remote ranging procedure.



5.3.2 Timeout handling

Note: In case the Initiator is actually busy with performing another ranging procedure, it will simply drop the received Remote-Range-Request frame from the Coordinator and does NOT notify the Coordinator of not performing this further ranging procedure. This is to prevent the implementation of various timers within the Coordinator for each remote ranging procedure. The timeout handling rather is requested to be done within the actual application requesting the remote ranging procedure.

5.4 Antenna diversity

The ranging algorithm supports antenna diversity if the utilized boards also support antenna diversity. This allows for more accurate ranging measurements by better coping with erroneous ranging results due to multipath propagation.

If antenna diversity is turned on, the ranging measurement is repeated on each frequency for each participating antenna. For example, in case only the Initiator supports antenna diversity, the ranging measurement is done twice, i.e. once for antenna 0 on the Initiator and once for antenna 1 on the Initiator, while the Reflector always uses its default antenna. If both nodes use antenna diversity, the ranging measurement is done four times. On the other hands this means that the overall ranging procedure will take longer, because the actual ranging measurement is longer.

The result exchange and the result calculation phases are longer, too, because more result data need to be transferred between the Reflector and the Initiator, and more data need to be calculated to get the final ranging results.

Depending on the number of actual ranging measurements (1 to 4) based on the used antennas, up to 4 different distances and DQF values are calculated (one for each antenna pair); result is one final (weighted) distance and DQF. The final distance and DQF is reported in the Range-Confirm message. In case the RTB PIB attribute *ProvideAntennaDivResults* (see Section 7.9) is true, and at least one node uses antenna diversity (i.e. more than one antenna pair results in more than one ranging measurement), the various distances and DQF values are included in the *RANGE-CONFIRM* message (see Ranging Measurement Pair parameters in Section 6.3).

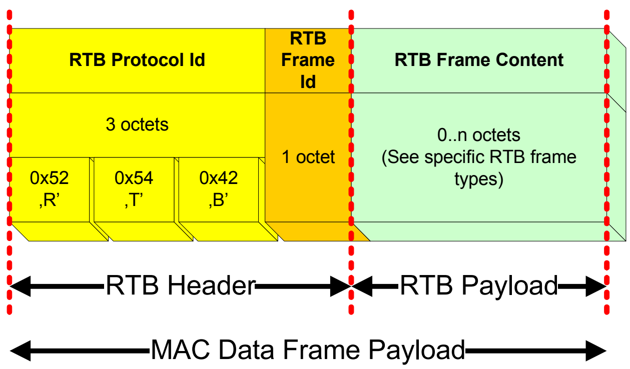
5.5 Ranging frame formats

As shown in Figure 5-2 and Figure 5-3, a variety of ranging specific frames is exchanged between the included nodes. The format of these frames is shown in the following sections.

5.5.1 General ranging frame format

Each RTB frame is an IEEE 802.15.4-2006 compliant Data frame including a properly defined MAC header. The payload of the MAC Data frame defines the actual RTB frame.

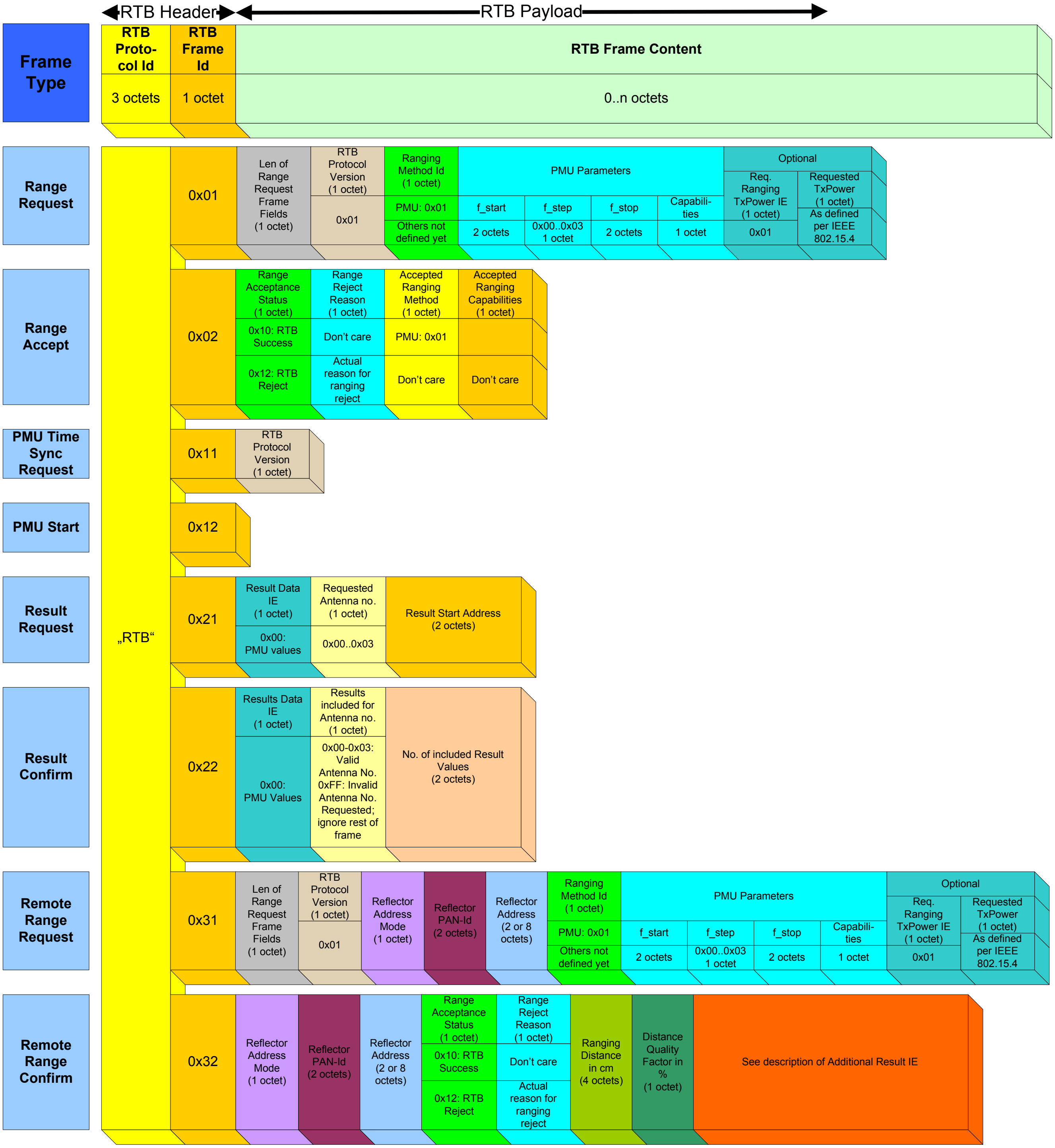
Figure 5-4. General RTB frame (i.e. MAC data payload).



5.5.2 Specific ranging frame formats

The content of the various RTB frames can be seen in Figure 5-5.

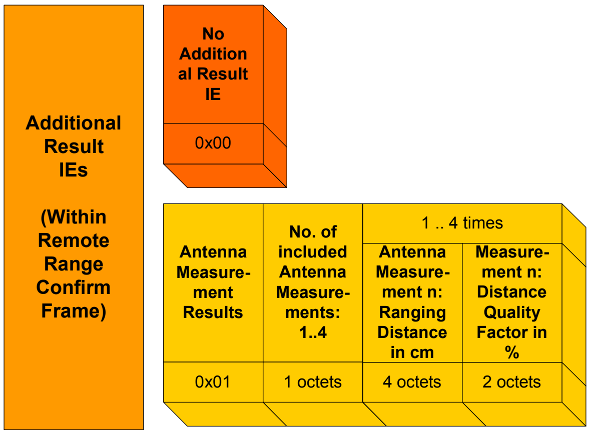
Figure 5-5. Specific RTB frame formats.



5.5.3 Additional Result Information Element within Remote Range Confirm frame

The Additional Result Information Element (IE) is appended at the end the Remote Range Confirm frame and it may contain the antenna diversity measurement results of the ranging measurement procedure.

Figure 5-6. Additional Result Information Element (IE) within the Remote Range Confirm.



5.6 Ranging capability handling

5.6.1 Local ranging capability negotiation

During the request for a local ranging procedure the Initiator sends its Ranging Capabilities to the Reflector based on its own RTB PIB attributes. This currently includes requesting the utilization of antenna diversity in case the initiator is able to handle antenna diversity.

These requested ranging capabilities are included in the initial Range Request frame. Upon reception of this frame the Reflector checks the received capabilities and compares it with its own capabilities. If a requested capability is incompatible with the Reflector's capabilities, this is corrected and the final ranging capabilities are assembled and returned to the Initiator by means of a Range Accept frame. The Initiator then stores the accepted ranging capabilities and applies them during the further ranging procedure.

5.6.2 Remote ranging capability

During the request for a remote ranging procedure the Coordinator sends its Remote Ranging Capabilities to the Initiator based on its own RTB PIB attributes. This includes (so far) only requesting the provisioning of antenna diversity results in case antenna diversity is used for any included node. If antenna diversity is actually used on the Initiator and/or Reflector, the Initiator finally returns the calculated antenna diversity measurement results (distance and DQF) to the Coordinator.

5.7 Handling of ranging transmit power

In order to adjust the ranging parameters to the given environment and create a more robust ranging system, the transmit power during the actual ranging measurement phase of all involved parties can be changed centralized or decentralized as required.

In order to handle the ranging transmit power decentralized at a given node, the RTB provides a PIB attribute RangingTransmitPower (see Section 7.11), which holds the ranging transit power to be utilized during the actual ranging measurement phase only. This allows for adjusting the transmit power during ranging for each node separately according the specific environment.

In conjunction to the decentralized approach, the utilized transmit power during the ranging measurement phase can be managed by the ranging initiating party. This is controlled by means of another RTB PIB attribute ProvideRangingTransmitPower (see Section 7.11). If this PIB attribute is set at the node requesting the ranging procedure, its current value of RangingTransmitPower is appended at the end of the Range Request frame transmitted by the Initiator to the Reflector, or the Remote Range Request frame initiated by the Coordinator. This received value of the ranging transmit power is then applied at the receiving party for its own ranging measurement, or forwarded to the Reflector (in Remote Ranging). This means, in Remote Ranging procedures the Initiator and Reflector apply the same transmit power as requested by the Coordinator; whereas in Local Ranging procedures the Reflector applies the same transmit power as requested by the Initiator.

The transmit power during the frame exchange phases of a ranging procedure (ranging setup and result exchange), is still controlled by the MAC/TAL based PIB attribute phyTransmitPower, which is independent from the actual value of RangingTransmitPower.

6. Ranging API

The RTB library provides the user an extended stack API which is the RTB-API. This API contains request functions for:

- Resetting the RTB
- Initiating a new regular or remote ranging measurement
- Setting ranging related PIB attributes

Furthermore the API contains callback functions (i.e. confirmations or indications) for:

- Confirming a requested RTB reset cycle
- Presenting the result of a previously requested ranging measurement
- Presenting PMU validity values calculated during the ongoing ranging measurement
- Presenting the result of a previously requested RTB PIB attribute change

The callback functions either need to be implemented (at least as stub functions) within the application or the provided stub function files (e.g. *usr_rtb_range_conf.c*) need to be linked to the project.

The complete list of provided RTB-API functions and callbacks with their parameters can be found in file *RTB/Inc/rtb_api.h*.

6.1 RTB reset functions

The RTB and all related PIB attributes can be reset from the application by calling the function *wpan_rtb_reset_req()*, which does not require any additional parameters.

```
bool wpan_rtb_reset_req(void);
```

The return value shall be checked to be true. Otherwise the request could not be queued into the corresponding queue for further processing due to an unavailability of buffers.

Once the reset cycle has been finished, the RTB invokes the callback function *usr_rtb_reset_conf()* providing the status of the reset cycle (e.g. *RTB_SUCCESS*) as member of the structure of type *usr_rtb_reset_conf_t*.

```
void usr_rtb_reset_conf(usr_rtb_reset_conf_t *urrc);
```

6.2 RTB PIB attribute set functions

The RTB contains a variety of ranging related PIB attributes (see Chapter 7) which need to be updated according to the specific needs of the application. These RTB PIB attributes can be changed by calling function *wpan_rtb_set_req()*.

```
bool wpan_rtb_set_req(wpan_rtb_set_req_t * wrsr);
```

The return value shall be checked to be true. Otherwise the request could not be queued into the corresponding queue for further processing due to an unavailability of buffers.

Once the PIB attribute setting has been finished, the RTB invokes the callback function *usr_rtb_set_conf()* providing the status of attempt to change the RTB PIB attribute (e.g. *RTB_SUCCESS, MAC_READ_ONLY*, *RTB_INVALID_PARAMETER*, *RTB_ UNSUPPORTED_ ATTRIBUTE*) as member of the structure of type *usr_rtb_set_conf_t*.

```
void usr_rtb_set_conf(usr_rtb_set_conf_t *ursc);
```

6.3 RTB ranging measurement functions

The RTB provides two basic means of ranging measurements:

1. Local / peer-to-peer ranging measurement: The Initiator directly invokes the Reflector to perform a ranging measurement.
2. Remote ranging measurement: The Coordinator remotely initiates a ranging procedure by requesting another node (i.e. the Initiator) to perform a ranging procedure with a third node (i.e. the Reflector).

All types of ranging are initiated by means of the same function *wpan_rtb_range_req()* using different parameters.

This function requires information about the ranging Initiator and Reflector, and furthermore about the Coordinator. In case the Coordinator address is omitted (Coordinator address mode is zero), a regular (i.e. peer-to-peer) ranging procedure shall be performed.

In case the Coordinator address is either two (Coordinator uses its short address) or three (Coordinator uses its long address), a Remote Ranging procedure shall be performed.

The actual ranging procedure is initiated by calling the function *wpan_rtb_range_req()* including the proper address information for the Initiator and Reflector, and the Coordinator address mode indicating the ranging type as member of the structure of type wpan_rtb_range_req_t.

```
bool wpan_rtb_range_req(wpan_rtb_range_req_t *wrrr);
```

The return value shall be checked to be true. Otherwise the request could not be queued into the corresponding queue for further processing due to an unavailability of buffers.

Once the ranging procedure has been finished, the RTB invokes the corresponding callback function *usr_rtb_range_conf()*. This callback function provides the ranging type and the corresponding ranging results within a structure of type *usr_rtb_range_conf_t*.

Part of the result structure is the status of the ranging measurement (e.g. *RTB_SUCCESS*, *RTB_UNSUPPORTED_RANGING*, *RTB_RANGING_IN_PROGRESS*, *RTB_ INVALID_PARAMETER*, etc.). If (and only if) the status of the ranging procedure is *RTB_SUCCESS*, the other included result parameters (distance, DQF, etc.) are valid.

Callback function for ranging:

```
void usr_rtb_range_conf(usr_rtb_range_conf_t *urrc);
```

Depending on the actual type of ranging (see member *ranging_type* of struct *usr_rtb_range_conf_t*) the results of the ranging procedure are contained within a union of type *range_conf_result_t* which has different parameters according to the actual ranging procedure.

For more information about the actual ranging structure see file *RTB/Inc/rtb_api.h*.

6.4 RTB PMU validity indication function

If the RTB PIB attribute *PMUVerboseLevel* is greater than zero and at least one of the included ranging parties (Initiator and/or Reflector) utilizes antenna diversity, the ranging procedure provides a PMU validity vector stating which frequency provided a valid ranging measurement. This may be of interest for applications performing some post-processing after the actual ranging measurement procedure.

This callback function *usr_rtb_pmu_validity_ind()* is generated asynchronously by the RTB itself before the ranging confirmation callback is generated.

```
void usr_rtb_pmu_validity_ind(usr_rtb_pmu_validity_ind_t *urpv);
```

7. Ranging PIB Attributes

7.1 RangingEnabled

bool RangingEnabled

This PIB attribute holds the current status whether ranging is generally enabled or not. It is set to true as soon as the RTB is included in the build by applying the build switch *ENABLE_RTB*, but can be disabled or enabled by the user.

7.2 RangingMethod

uint8_t RangingMethod

This PIB attribute holds the actually used ranging method. Currently only ranging based on PMU (Phase Measurement Unit) is implemented and allowed:

```
RANGING_METHOD_PMU        0x01
```

This PIB attribute is read-only at the moment.

7.3 PMUFreqStart

uint16_t PMUFreqStart

This PIB attribute holds the PMU measurement start index in the frequency table in case the ranging method PMU is used. This PIB attribute can be changed. Currently the PMU frequency start value is allowed within the range of the minimum and maximum frequency as shown below:

```
PMU_MIN_FREQ       2324 MHz
PMU_MAX_FREQ       2527 MHz
Default value           2403 MHz
```

The selected PMU frequency start value must be less than the selected PMU frequency stop value (see Section 7.5).

7.4 PMUFreqStep

uint8_t PMUFreqStep

This PIB attribute holds the PMU measurement step size in the frequency table in case the ranging method PMU is used. This PIB attribute can be changed.

Currently the following values are allowed:

```
PMU_STEP_FREQ_500kHz      0
PMU_STEP_FREQ_1MHz        1
PMU_STEP_FREQ_2MHz       2 (Default value)
PMU_STEP_FREQ_4MHz       3
```

7.5 PMUFreqStop

uint16_t PMUFreqStop

This PIB attribute holds the PMU measurement stop index in the frequency table in case the ranging method PMU is used. This PIB attribute can be changed. Currently the PMU frequency stop value is allowed within the range of the minimum and maximum frequency as shown below:

```
PMU_MIN_FREQ       2324 MHz
PMU_MAX_FREQ       2527 MHz
Default value      2443 MHz
```

The selected PMU frequency start value must be greater than the selected PMU frequency start value (see Section 7.3).

7.6 PMUVerboseLevel

uint8_t PMUVerboseLevel

This PIB attribute holds the verbose level for providing additional PMU measurement information. The following verbose levels are defined:

```
0      Default level, no further information provided
1      Provisioning of PMU validity information (see Section 6.4) is enabled (for
       all antenna measurement pairs)
```

7.7 DefaultAntenna

bool DefaultAntenna

This PIB attribute holds the ranging procedure default antenna of the node:

```
False  antenna 0 used as default
True   antenna 1 used as default
```

The default value for this PIB attribute is false (i.e. antenna 0). This attribute is only used actively in case the node does not use antenna diversity.

7.8 EnableAntennaDiv

bool EnableAntennaDiv

This PIB attribute is only available in case the node is generally capable using antenna diversity (*USE_ANTENNA_DIVERSITY = 1*) and holds the default value for antenna diversity:

```
False  antenna diversity not used as default
True   antenna diversity used as default
```

This PIB attribute is set to true as default (provided the board generally allows antenna diversity).

7.9 ProvideAntennaDivResults

bool ProvideAntennaDivResults

This PIB attribute holds the default value for providing all measured distances and DQFs based on the applied antenna diversity scheme within the Range-Confirm message if this node acts as the Initiator.

```
False  Only the final calculated distance and DQF are provided within Range-
       Confirm message (default value)
True   In addition to the final calculated distance and DQF also each separate
       measured distance and DQF are appended at the end of the Range-Confirm
       message (only valid if at least one node uses antenna diversity)
```

Note: Even if the particular node not provides antenna diversity, the peer node may use antenna diversity, so the utilization of this feature is not only dependent from the status of EnableAntennaDiv of this node, but also from the status of EnableAntennaDiv of the peer node.

7.10 RangingTransmitPower

uint8_t RangingTransmitPower

This PIB attribute holds the current transmit power during ranging of the node to be explicitly applied during the ranging measurement phase. The value of the PIB attribute represents the transmit power is dBm and is dependent on the transmit power range of the utilized TRX.

7.11 ProvideRangingTransmitPower

bool ProvideRangingTransmitPower

This PIB attribute holds the current value for sending the other ranging party the ranging transmit power to be applied during the actual ranging measurement phase.

```
False  (1) This node is the Initiator of a Local Ranging procedure and will sent
       the Reflector its own value of the PIB attribute RangingTransmitPower to be
       applied at the Reflector during ranging.
       (2) This node is the Coordinator during a Remote Ranging procedure and will
       sent the Reflector its own value of the PIB attribute RangingTransmitPower
       to be applied at both the Initiator and the Reflector during ranging.
True   This node will not provide the other parties (as described above) with its
       own ranging transmit power.
```

7.12 ApplyMinDistThreshold

bool ApplyMinDistThreshold

This PIB attribute holds the current value for applying a minimum threshold during weighted distance calculation.

```
False  The minimum threshold will be applied during weighted distance calculation
True   The minimum threshold will not be applied during weighted distance
       calculation; a regular minimum search among all measured distance values
       will be applied
```

8. Build Switches

A variety of build switches are available for configuring the RTB library.

8.1 ANTENNA_DIVERSITY

This build switch controls the general support of antenna diversity for a given node. If this switch is set to 1 (*ANTENNA_DIVERSITY = 1*), antenna diversity is supported.

8.2 DEBUG

This build switch enables the inclusion of additional debug code during the ranging procedure. This provides additional information during the validation of the RTB.

Additionally the Distance Offset (i.e. fixed offset based on board and antenna design to be applied for each ranging measurement) can be adjusted and tested within the RTB Evaluation Application during the ranging evaluation or board bring-up by enabling the DEBUG switch (DEBUG = 1).

8.3 ENABLE_RTB

This build switch enables the inclusion of the RTB in the end user application.

8.4 ENABLE_RTB_PRINT

This build switch enables additional printouts during the ranging operation. This build switch is applied within the RTB Evaluation Application, but is not recommended for production code.

8.5 ENABLE_RTB_REMOTE

This build switch enables the Remote Ranging functionality including Coordinator node type. This build switch is applied within the RTB Evaluation Application, but can be omitted if not required in order to reduce footprint.

8.6 RTB_TYPE

This build switch defines the utilized type of ranging. Currently AT86RF233 phase difference measurement technology is supported (RTB_TYPE = RTB_PMU_233R).

9. Status and Error Codes

The following ranging specific status codes are generated within the RTB (See file *Include/return_val.h*):

```
RTB_SUCCESS:                  Success of ranging procedure
RTB_RANGING_IN_PROGRESS:      Ranging procedure is already in progress
RTB_REJECT:                   Ranging procedure is rejected
RTB_UNSUPPORTED_ATTRIBUTE:    Unsupported attribute for RTB
RTB_INVALID_PARAMETER:        Unsupported parameter for RTB
RTB_OUT_OF_BUFFERS:           No further buffers available for RTB ranging
                              measurement
RTB_UNSUPPORTED_RANGING:      Ranging is currently not supported
RTB_UNSUPPORTED_METHOD:       Requested Ranging method is currently not supported at
                              reflector
RTB_TIMEOUT:                  Timeout since requested Ranging
                              response frame is not received
RTB_UNSUPPORTED_PROTOCOL:     Requested RTB Protocol is currently not supported at
                              node
```

Furthermore a variety of error codes are generated within the RTB further specifying the error cause in conjunction to a provided status. The pre-defined error codes are defined in file *RTB/Inc/rtb_internal.h* (see *typedef range_error_t*), but also status codes as mentioned above can be used as error codes.

10. Abbreviations

API	-	Application programming interface
BMM	-	Buffer management module
DQF	-	Distance quality factor
IE	-	Information Element
I/F	-	Interface
MAC	-	Medium access control
MCL	-	MAC core layer
MCU	-	Microcontroller Unit
NHLE	-	Next higher layer entity
PAL	-	Platform abstraction layer
PIB	-	PAN information base
PMU	-	Phase measurement unit
QMM	-	Queue management module
RTB	-	Ranging Toolbox
TAL	-	Transceiver abstraction layer
TRX	-	Radio Transceiver

This page is intentionally left blank.

Appendix A. EVALUATION BOARD/KIT IMPORTANT NOTICE

This evaluation board/kit is intended for use for **FURTHER ENGINEERING, DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY**. It is not a finished product and may not (yet) comply with some or any technical or legal requirements that are applicable to finished products, including, without limitation, directives regarding electromagnetic compatibility, recycling (WEEE), FCC, CE or UL (except as may be otherwise noted on the board/kit). Atmel supplied this board/kit "AS IS," without any warranties, with all faults, at the buyer's and further users' sole risk. The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies Atmel from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge and any other technical or legal concerns.

EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER USER NOR ATMEL SHALL BE LIABLE TO EACH OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

No license is granted under any patent right or other intellectual property right of Atmel covering or relating to any machine, process, or combination in which such Atmel products or services might be or are used.

Mailing Address: Atmel Corporation, 2325 Orchard Parkway, San Jose, CA 95131.

Appendix B. References

[1]. ATMEL Wireless Offerings - http://www.atmel.com/products/microcontrollers/wireless/default.aspx.

[2]. IEEE Standard 802.15.4TM-2006: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs). http://standards.ieee.org/getieee802/download/802.15.4-2006.pdf.

[3]. AVR®2150; RTB Evaluation Application – User's Guide; Application Note; Atmel Corporation.

[4]. AT86RF233; Low Power, 2.4GHz Transceiver for ZigBee, RF4CE, IEEE 802.15.4, 6LoWPAN, ISM and Ranging Applications; Datasheet; Atmel Corporation; http://www.atmel.com/Images/Atmel-8351-MCU_Wireless-AT86RF233_Datasheet.pdf.

[5]. AVR2151; RTB Evaluation Application – Quick Start Guide; Application Note; Atmel Corporation.

[6]. REB233SMAD-EK - www.atmel.com/tools/REB233SMAD-EK.aspx.

Appendix C. Revision History

Doc. Rev.	FW Rev.	Date	Comments
8443A	1.1.7	02/2013	Initial document release

Atmel Corporation
1600 Technology Drive
San Jose, CA 95110
USA
Tel: (+1)(408) 441-0311
Fax: (+1)(408) 487-2600
www.atmel.com

Atmel Asia Limited
Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
Tel: (+852) 2245-6100
Fax: (+852) 2722-1369

Atmel Munich GmbH
Business Campus
Parkring 4
D-85748 Garching b. Munich
GERMANY
Tel: (+49) 89-31970-0
Fax: (+49) 89-3194621

Atmel Japan G.K.
16F Shin-Osaki Kangyo Building
1-6-4 Osaki, Shinagawa-ku
Tokyo 141-0032
JAPAN
Tel: (+81)(3) 6417-0300
Fax: (+81)(3) 6417-0370