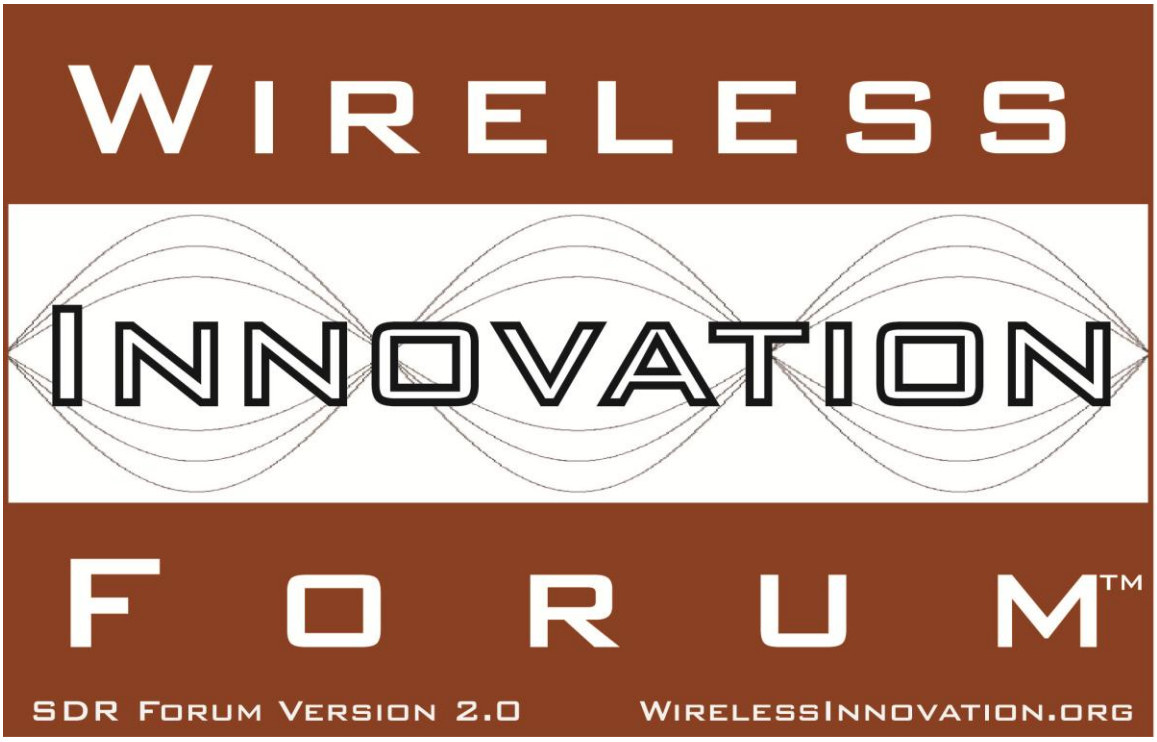

Change proposal to Draft SCA Next AEP (Appendix B) for harmonization with ESSOR Architecture

WINNF-11-R-0005

Version V1.0.0

14 September 2011

TERMS, CONDITIONS & NOTICES

This document has been prepared by the SCA Next Working Group to assist The Software Defined Radio Forum Inc. (or its successors or assigns, hereafter "the Forum"). It may be amended or withdrawn at a later time and it is not binding on any member of the Forum or of the SCA Next Working Group.

Contributors to this document that have submitted copyrighted materials (the Submission) to the Forum for use in this document retain copyright ownership of their original work, while at the same time granting the Forum a non-exclusive, irrevocable, worldwide, perpetual, royalty-free license under the Submitter's copyrights in the Submission to reproduce, distribute, publish, display, perform, and create derivative works of the Submission based on that original work for the purpose of developing this document under the Forum's own copyright.

Permission is granted to the Forum's participants to copy any portion of this document for legitimate purposes of the Forum. Copying for monetary gain or for other non-Forum related purposes is prohibited.

THIS DOCUMENT IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY USE OF THIS SPECIFICATION SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER THE FORUM, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER, DIRECTLY OR INDIRECTLY, ARISING FROM THE USE OF THIS DOCUMENT.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the specification set forth in this document, and to provide supporting documentation.

Attention is also drawn to the possibility that the Forum shall not be responsible for identifying any or all such IPR.

Wireless Innovation Forum ™ and SDR Forum ™ are trademarks of the Software Defined Radio Forum Inc.

Table of Contents

Preface

The JTRS JPEO has released in December 2010 the Draft SCA Next specification.

Simultaneously, the ESSOR Industries have finalized the elaboration of the ESSOR SCA-based Architecture.

The SCA Next Work Group of the WInnF prepared this change proposal in one attempt to maximize the harmonization between ESSOR and JTRS achievements, using as an input the document WINNF-11-I-0007 submitted by ESSOR Industries to WINNF SCA Next WG.

This document is being submitted by the WInnF to the JTRS SCA Next Working Panel for consideration.

It is the SCA Next WG expectation that it will be followed by JTRS SCA Next Working Panel in order to achieve harmonization on the topic of interest.

Change proposal to Draft SCA Next AEP (Appendix B) for harmonization with ESSOR Architecture

1 References

- **[Ref 1] –** Draft SCA Next, JTRS Standards
  http://www.public.navy.mil/jpeojtrs/sca/Pages/scanext.aspx
- **[Ref 2]** – WINNF-11-I-0007, ESSOR Industries
  http://groups.winnforum.org/d/do/4690
- **[Ref 3]** – Appendix B of Draft SCA Next, JTRS Standards
  http://www.public.navy.mil/jpeojtrs/sca/Documents/scanextdraft/SCA_NextDraft_20101130_App_B_ScaApplicationEnvironmentProfiles.pdf
- **[Ref 4]** – JTRS SCA v2.2.2 Standards
  http://www.public.navy.mil/jpeojtrs/sca/Documents/SCAv2_2_2/SCA_version_2_2_2.pdf

2 Introduction

2.1 Purpose of the document

As indicated in Preface, this document reports the WInnF SCA Next WG conclusions in preparing a Change Proposal in order to harmonize Appendix B of the SCA Next with correlated parts of the ESSOR SDR Architecture, pursuing the aim to define additional profile(s) for DSP or comparable restricted processing elements where the full SCA 2.2.2. AEP is not suitable.

This document has been prepared in perspective to serve as input to JTRS Standards work to finalize the SCA Next, from the draft published in December 2010 [Ref 1].

It takes the following inputs as references for the harmonization effort:

- **[Ref 2]**: extract from ESSOR SDR Architecture relative to LwAEP submitted by ESSOR Industries in May 2011 as an input to WInnF Activities,
- **[Ref 3]**: Appendix B of Draft JTRS SCA Next, released in December 2010.

2.2 Executive summary

The document recommends the introduction of an Ultra Light Weight Application Environment Profile (ULwAEP) and a number of modifications on the LwAEP of the Draft SCA Next. Both are aimed to support the needs of DSP processing elements (with CORBA or Non-CORBA connection mechanisms). The introduced ULwAEP is a strict subset of the LwAEP.

The document includes a rationale and a set of specific amendments to draft SCA Next Appendix B [Ref 3].

2.3 Document organization

The document is further structured in two parts:

- §3 - Change Proposal Rationale, reporting the main principles and arguments in support of the recommended changes,
- §4 - Change Proposal Amendments, where corresponding amendments applicable to the Draft Appendix B are proposed for finalization of SCA Next.

3 Change Proposal Rationale

3.1 Driving principles of the ESSOR DSP AEP

The ESSOR profile for DSP environments, provided in [Ref 2], has been defined in accordance with the following principles:

Principle 1. **Operating System (OS) focused.** The ESSOR DSP AEP applies to an Execution Environment, providing the definition of the core technical software services an OS should supply.

Principle 2. **No standard C processing libraries in the Profile.** Standard C libraries are not in the scope of what the platform has to provide.

Principle 3. **Supporting creation of OS resources.** Functions enabling creation of OS resources are selected.

Principle 4. **Supporting essential run-time usage of OS resources.** Any necessary run-time function enabling the use of OS resources are specified in the profile.

Principle 5. **Supporting no post-creation modification of OS resources.** No function enabling post-creation modifications of OS resources are present in the ESSOR DSP AEP.

Principle 6. **WF dynamic tear-down capability.** The ESSOR DSP AEP has been defined in order to support destruction of OS resources during the tear-down of a waveform.

3.2 Analysis of the ESSOR DSP AEP and Draft SCA Next LwAEP

The Appendix B of the Draft SCA Next, cf. [Ref 3], specifies two profiles: the "AEP" and the "Lightweight AEP", abbreviated "LwAEP".

The correlated extract of the ESSOR Architecture, cf. [Ref 2], introduces a "DSP AEP".

In general, the original DSP AEP of ESSOR Architecture appears as a subset of the LwAEP of Draft SCA Next. Though, a number of features are present in the DSP AEP of ESSOR, while not being present in the LwAEP of Draft SCA Next.

More specifically, the following section is reporting the differences between:

- ESSOR DSP AEP as in [Ref 2],
- Draft SCA Next LwAEP as in [Ref3].

3.2.1 Comparison of the ESSOR DAP AEP and Draft SCA Next LwAEP

The comparison of the two profiles is performed by the two following chapters in the following way: first, all the features that are contained in the Draft SCA Next LwAEP, but are not in the ESSOR DSP AEP, are listed; second, all the features of that are contained in the ESSOR DSP AEP while not in the Draft SCA Next LwAEP are reported.

Any other features are common between the two profiles and are not specifically reported.

3.2.1.1 Specific Draft SCA Next LwAEP features

This chapter identifies what is present in the SCA Next Draft LwAEP while not in the ESSOR DSP AEP. These identified features are:

- A set of C language support elements specified as MAN in the Draft SCA Next LwAEP are not specified in the ESSOR DSP AEP, due to the fact the ESSOR DSP AEP focuses on OS aspects.
- The OS support functions provided by the ESSOR DSP AEP are mainly included in the SCA Next Draft LwAEP. However, the ESSOR DSP AEP contains only the necessary functionalities to support WF deployment and execution.

3.2.1.2 Specific ESSOR DSP AEP features

This chapter identifies the ESSOR DSP AEP functions that are not present in the SCA Next Draft LwAEP.

The following table provides a view of these specific aspects: the first two columns are referencing the ESSOR Profile features while the third column reports if, for each line, the capability is included ("Present") in Draft SCA Next LwAEP or not ("Not Present").

The column "POSIX Profile" is capturing the clauses from the ESSOR DSP AEP that are avoiding to impose some aspects of the POSIX full behavior that could imply overheads in an DSP RTOS implementation.

From ESSOR DSP AEP		From Draft SCA Next LwAEP
Functions	**POSIX Profile**	**Presence**
PTHREAD		
pthread_attr_init		Present
pthread_attr_setschedparam		Present
pthread_attr_setstacksize		Present
pthread_create		Present
pthread_cancel	At least allow the following behavior: the cancellation processing is run in the calling thread	Not Present
MQUEUE		
mq_open	At least the following mq_attr structure values are supported: • mq_flags == O_RDWR • lmq_msgsize = 4 bytes	Present
mq_close		Present
mq_send	Msg_prio is ignored , all message are processed with the same priority within a given queue	Present
mq_receive	At least msg_prio = NULL is supported	Present
mq_timedreceive	At least msg_prio = NULL is supported	Present
SEMAPHORE		
sem_init		Present
sem_timedwait		Present
sem_post		Present
Sem_wait		Present
sem_destroy		Present
MUTEX		
pthread_mutex_init	At least attr = NULL (mutex type is PTHREAD_MUTEX_DEFAULT) is supported	Present
pthread_mutex_lock		Present
pthread_mutex_trylock		Present
pthread_mutex_unlock		Present
pthread_mutex_destroy		Present
pthread_mutex_attr_init		Not Present
TIMER		
timer_create	At least Clockid = CLOCK_REALTIME is supported At least the following sigevent structure values are supported : int Sigev_notify = SIGEV_THREAD; (pthread_attr_t*) sigev_notify_attributes = NULL;	Present
timer_delete		Present
timer_settime	At least Old_setting =NULL is supported At least the following Timespec structure values are supported : At least seconds = 0 is supported Timer Accuracy is 1 us.	Present
MEMORY MANAGEMENT		
malloc()		Present
free()		Present

WInnF Table 1. Draft SCA Next LwAEP compliancy with ESSOR DSP AEP

3.2.2 Resolution of differences

In order to resolve the differences, two solutions have been analyzed:

- **Merge.** This solution proposes a merge process for all differences, including and/or removing some functionalities from both AEP's in order to have a new one including all the agreed functionalities,

- **Additional Profile.** This solution proposes the inclusion, inside the SCA Next specification, of a third profile called Ultra Lightweight AEP (ULwAEP), containing only the functionalities needed for WF support.

3.2.2.1 Merge solution

In conformance with the comparison presented beforehand, a number of specific Draft SCA Next LwAEP and ULwAEP features were analyzed to move to/from MAN to NRQ in the final release of SCA Next in an effort to merge the LwAEP and the proposed ULwAEP.

However, the LwAEP and the ULwAEP feature analysis did not yield and exact match for a merged profile. Specific feature areas that did not match were:

- For some RTOS related features,
- For features related to C language support.

3.2.2.2 Additional Profile solution

The solution consists in introducing the ULwAEP inside the final SCA Next specification, while making evolve the existing LwAEP.

The recommendations for introduction of the ULwAEP are:

- The ULwAEP is to reflect all the features of the ESSOR DSP AEP in [Ref 2],
- All the C language support functions are to be introduced as NRQ,
  - This implies to change the malloc() / free() functions from MAN to NRQ (difference with ESSOR DSP AEP),
- The function pthread_cancel() is to be changed from MAN to NRQ (difference with ESSOR DSP AEP).

The recommendations for evolution of the LwAEP are:

- Function pthread_mutex_attr_init() to be changed from NRQ to MAN,
- A number of OS support functions to be changed from MAN to NRQ.

3.2.3 Summary of the recommendations

It is recommended to introduce the ESSOR Profile as the Ultra Lw AEP (ULwAEP) of the SCA Next, as expressed in the previous chapter.

The recommended outcome would be composed of three different profiles, which can be chosen depending on the needs of any specific development:

- **SCA AEP**: the AEP defined in JTRS SCA v2.2.2 specification [Ref 4],
- **LwAEP**: a lightweight profile derived from the one introduced in Draft SCA Next,
- **ULwAEP**: an ultra-lightweight profile derived from ESSOR SDR Architecture.

The Chapter 4 translates these general modifications into detailed amendments to the JTRS SCA Next Appendix B specification, aiming the facilitation of its inclusion in final SCA Next.

4 Change Proposal Amendments

The impacted documentation is [Ref 3]:

"APPENDIX B: SCA APPLICATION ENVIRONMENT PROFILES"

Referenced version:

"Next <Draft>, 20 November 2010"

There are 11 change proposal amendments, specifically described in the following subchapters.

4.1 Referencing ULwAEP in B.1 SCOPE

4.1.1 Proposed text

<START>
This appendix defines the Application Environment Profile (AEP) , Lightweight (LwAEP) and Ultralightweight (ULwAEP) for the SCA, based on Standardized Application Environment Profile - POSIX® Realtime Application Support, IEEE Std 1003.13-2003.

This appendix also extends the SCA AEP with networking capabilities for SCA compliant applications that require this functionality.

The SCA AEP, LwAEP and ULwAEP are the SCA required profiles referenced in sections 3.1.1, 3.2.1, and 3.3.1 of the main document. The SCA dictates that an Operating Environment provides the options and functions designated as mandatory within the supported profile and constrains an application to only use those services.

<END>

4.2 Reference ULwAEP in B.2 CONFORMANCE

Remark: typos “CONFORMANACE” and “ApplicationResourceCompo**net**” in original [Ref 3].

4.2.1 Proposed text

<START>

B.2.1 Application Environment Profiles Conformance on the Part of an SCA Product

The elements of this specification are not required to be used solely for a particular platform or application. This specification identifies the collection of Operating System (OS) services that are available for use by an ApplicationResourceComponent. However for an Operating Environment this specification identifies the minimum set of OS services that must be supported. Conformance for an SCA Product is at the level of usage as follows:

- An SCA Product needs to be conformant with the mandatory elements of a Profile defined within this Appendix.

The networking capabilities (i.e. networking and event management function behavior) are beyond the scope of the AEP, LwAEP and ULwAEP profiles. AEP (only) conformance for an SCA Product may be extended to incorporate the networking capabilities as follows:

- An AEP conformant SCA Product needs to be conformant with the mandatory elements of the networking capabilities defined within this Appendix.

B.2.2 Sample Conformance Statements

An SCA product can be identified as being conformant to a specific version of the SCA and the specific technology that the product realizes.

- “Product A is an SCA Next conformant waveform application in accordance with the LwAEP and the CORBA/XML platform.”
- “Product B is an SCA Next conformant Operating Environment in accordance with an AEP POSIX layer with networking and a CORBA (full profile) transfer mechanism.”
- “Product C is an SCA Next conformant Operating Environment in accordance with a LwAEP POSIX layer and a CORBA (full profile) transfer mechanism.”
- “Product D is an SCA Next conformant Operating Environment in accordance with a ULwAEP POSIX layer and a CORBA (Lightweight/UltraLightweight profile) transfer mechanism.”

<END>

4.3 Reference ULwAEP in "B.4 STANDARDS"

4.3.1 Principle

Table 1 of [Ref 3] to be completed with a column dedicated to ULwAEP.

4.3.2 Proposed text

<START>
The standards identified in Table 1 are required in whole or in part by the SCA AEP,LwAEP and ULwAEP.

Table 1. Required Standards

Standard	AEP	LwAEP	ULwAEP
C Standard (ISO/IEC 9899:1999)	PRT	PRT	NRQ
POSIX (ISO/IEC 9945:2003)	PRT	PRT	PRT

The specific functions and options will be identified in following chapters.

<END>

4.4 Update introduction of B.5

4.4.1 Principle

Modification: insert a paragraph bringing a necessary clarification on the ULwAEP.

Location: after first paragraph, "(…) These constraints must be observed by an application that conforms to the profile when using each of the required functions."

4.4.2 Proposed text

<START>

The ULwAEP is intended to specify expectations towards RTOS capabilities to be provided by compliant platforms.

<END>

4.5 Update POSIX options table

4.5.1 Principle

Table 2 of [Ref 3] to be completed with a column dedicated to ULw AEP.

4.5.2 Proposed text

<START>

The options, limits, and any other constraints on POSIX.1 shall be provided as described in Table 2.

Table 2. POSIX.1 Option Requirements

Option	AEP	LwAEP	ULwAEP
{_POSIX_ASYNCHRONOUS_IO}	MAN	NRQ	NRQ
{_POSIX_CHOWN_RESTRICTED}	NRQ	NRQ	NRQ
{_POSIX_CLOCK_SELECTION}	NRQ	NRQ	NRQ
{_POSIX_FSYNC}	PRT	NRQ	NRQ
{_POSIX_MAPPED_FILES}	NRQ	NRQ	NRQ
{_POSIX_MEMLOCK_RANGE}	MAN	NRQ	NRQ
{_POSIX_MEMLOCK}	MAN	NRQ	NRQ
{_POSIX_MEMORY_PROTECTION}	NRQ	NRQ	NRQ
{_POSIX_MESSAGE_PASSING}	MAN	PRT	PRT
{_POSIX_MONOTONIC_CLOCK}	NRQ	NRQ	NRQ
{_POSIX_NO_TRUNC}	PRI	NRQ	NRQ
{_POSIX_PRIORITIZED_IO}	NRQ	NRQ	NRQ
{_POSIX_PRIORITY_SCHEDULING}	NRQ	NRQ	NRQ
{_POSIX_REALTIME_SIGNALS}	MAN	NRQ	NRQ
{_POSIX_SAVED_IDS}	NRQ	NRQ	NRQ
{_POSIX_SEMAPHORES}	MAN	PRT	PRT
{_POSIX_SHARED_MEMORY_OBJECTS}	NRQ	NRQ	NRQ
{_POSIX_SYNCHRONIZED_IO}	PRT	NRQ	NRQ
{_POSIX_THREAD_ATTR_STACKADDR}	MAN	MAN	NRQ
{_POSIX_THREAD_ATTR_STACKSIZE}	MAN	MAN	MAN
{_POSIX_THREAD_CPUTIME}	NRQ	NRQ	NRQ
{_POSIX_THREAD_PRIO_INHERIT}	MAN	NRQ	NRQ
{_POSIX_THREAD_PRIO_PROTECT}	MAN	NRQ	NRQ
{_POSIX_THREAD_PRIORITY_SCHEDULING}	MAN	MAN	NRQ
{_POSIX_THREAD_PROCESS_SHARED}	NRQ	NRQ	NRQ
{_POSIX_THREAD_SAFE_FUNCTIONS}	PRT	NRQ	NRQ
{_POSIX_THREAD_SPORADIC_SERVER}	NRQ	NRQ	NRQ
{_POSIX_TIMEOUTS}	NRQ	NRQ	NRQ
{_POSIX_TIMERS}	MAN	PRT	PRT
{_POSIX_TRACE_EVENT_FILTER}	NRQ	NRQ	NRQ
{_POSIX_TRACE_LOG}	NRQ	NRQ	NRQ
{_POSIX_TRACE}	NRQ	NRQ	NRQ
{_POSIX_VDISABLE}	NRQ	NRQ	NRQ

<END>

4.6 Specify Message Queues capabilities

4.6.1 Principle

Modification: add a section with table specifying Message Queues capabilities, since message queues are not referenced in **[Ref 3]**.

Corresponds to the PRT statement in {_POSIX_MESSAGE_PASSING} from previous Table 2 of **[Ref 3]**.

Location: to be defined by JTRS Standards.

4.6.2 Proposed text

<START>

B.5.1.<TBD> POSIX Message Queues Function Behavior

The functions listed in Table <tbd> shall behave as described in the applicable clauses of the referenced POSIX specifications.

Table <tbd>. POSIX_MQUEUE functions

Function	AEP	LwAEP	ULwAEP
mq_close()	MAN	MAN	MAN
mq_getattr()	MAN	NRQ	NRQ
mq_notify()	MAN	NRQ	NRQ
mq_open()	MAN	MAN	MAN[1]
mq_receive()	MAN	MAN	MAN[2]
mq_send()	MAN	MAN	MAN[3]
mq_setattr()	MAN	NRQ	NRQ
mq_timedreceive()	MAN	MAN	MAN[4]
mq_timedsend()	MAN	NRQ	NRQ
mq_unlink()	MAN	NRQ	NRQ

Note: use of POSIX Message Queue functionality is restricted to intra-process communications. For processing elements (e.g. DSP) that do not support process /partitioning the use of POSIX Message Queue functionality is restricted to task to task (also referred to threads) communications.

<END>

---

[1] At least the following mq_attr structure values are supported:

- mq_flags == O_RDWR
- lmq_msgsize = 4 bytes

[2] At least msg_prio = NULL is supported

[3] Msg_prio is ignored , all message are processed with the same priority within a given queue

[4] At least msg_prio = NULL is supported

4.7 Specify Semaphores capabilities

4.7.1 Principle

Table 20 of [Ref 3] to be completed with a column dedicated to ULwAEP.

4.7.2 Proposed text

<START>

B.5.1.18 POSIX Semaphore Function Behavior

The functions listed in Table 20 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table 1.

Table 20. POSIX_SEMAPHORES Functions

Function	AEP	LwAEP	ULwAEP
sem_close()	MAN	NRQ	NRQ
sem_destroy()	MAN	MAN	MAN
sem_getvalue()	MAN	MAN	NRQ
sem_init()	MAN	MAN	MAN
sem_open()	MAN	NRQ	NRQ
sem_post()	MAN	MAN	MAN
sem_timedwait()	MAN	MAN	MAN
sem_trywait()	MAN	NRQ	NRQ
sem_unlink()	MAN	NRQ	NRQ
sem_wait()	MAN	MAN	MAN

<END>

4.8 Specify Timers capabilities

4.8.1 Principle

Table 21 of [Ref 3] to be completed with a column dedicated to ULwAEP.

4.8.2 Proposed text

<START>

B.5.1.19 POSIX Timer Function Behavior

The functions listed in Table 21 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table 1.

Table 21. POSIX_TIMERS Functions

Function	AEP	LwAEP	ULWAEP
clock_getres()	MAN	MAN	NRQ
clock_gettime()	MAN	MAN	NRQ
clock_settime()	MAN	MAN	NRQ
nanosleep()	MAN	NRQ	NRQ
timer_create()	MAN	MAN	MAN[5]
timer_delete()	MAN	MAN	MAN
timer_getoverrun()	MAN	NRQ	NRQ
timer_gettime()	MAN	MAN	NRQ
timer_settime()	MAN	MAN	MAN[6]

ULwAEP only mandates functions required for scheduling management. Any other time related capabilities are assumed to be provided by other mechanisms.

<END>

[5] At least Clockid = CLOCK_REALTIME is supported

At least the following sigevent structure values are supported :

- int Sigev_notify = SIGEV_THREAD;
- (pthread_attr_t*) sigev_notify_attributes = NULL;

[6] At least Old_setting =NULL is supported

At least the following Timespec structure values are supported :

At least seconds = 0 is supported

Timer Accuracy is 1 us.

4.9 Threading and Mutex capabilities

4.9.1 Principle

Table 22 of [Ref 3] to be completed with a column dedicated to ULwAEP.

4.9.2 Proposed text

<START>

B.5.1.20 POSIX Threading Function Behavior

The functions listed in Table 22 shall behave as described in the applicable clauses of the referenced POSIX specifications contained in Table 1.

Table 22. POSIX_THREADS_BASE Functions

Function	AEP	LwAEP	ULwAEP
pthread_atfork()	NRQ	NRQ	NRQ
pthread_attr_xxx()	MAN	PRT[7]	PRT[8]
pthread_cancel()	MAN	NRQ	NRQ
pthread_cleanup_xxx()	MAN	NRQ	NRQ
pthread_cond_xxx()	MAN	NRQ	NRQ
pthread_condattr_xxx()	MAN	NRQ	NRQ
pthread_create()	MAN	MAN	MAN
pthread_detach()	MAN	NRQ	NRQ
pthread_equal()	MAN	NRQ	NRQ
pthread_exit()	MAN	NRQ	NRQ
pthread_getschedparam()	MAN	NRQ	NRQ
pthread_getspecific()	MAN	NRQ	NRQ
pthread_join()	MAN	MAN	NRQ
pthread_key_xxx()	MAN	NRQ	NRQ
pthread_kill()	MAN	NRQ	NRQ
pthread_mutex_xxx()	MAN	PRT[9]	PRT[10]
pthread_mutexattr_xxx()	MAN	PRT[11]	PRT[12]
pthread_once()	MAN	NRQ	NRQ
pthread_self()	MAN	NRQ	NRQ
pthread_setcancelstate()	MAN	NRQ	NRQ
pthread_setcanceltype()	MAN	NRQ	NRQ
pthread_setschedparam()	MAN	NRQ	NRQ

[7] Except for pthread_attr_getstack (), pthread_attr_setstack(),pthread_attr_getinheritsched(), pthread_attr_getschedpolicy(), pthread_attr_getscope(), pthread_attr_setinheritsched(), pthread_attr_setschedpolicy(), and pthread_attr_setscope().
[8] Only included: pthread_attr_init(),pthread_attr_setschedparam(),pthread_attr_setstacksize()
[9] pthread_mutex_getprioceiling() and pthread_mutex_setprioceiling() excluded since _POSIX_THREAD_PRIO_PROTECT is not required.
[10] Only included: pthread_mutex_init(), pthread_mutex_lock(), pthread_mutex_trylock(), pthread_mutex_unlock(), pthread_mutex_destroy()
[11] Only included: pthread_mutex_attr_init()
[12] Only included: pthread_mutex_attr_init()

Function	AEP	LwAEP	ULwAEP
pthread_setspecific()	MAN	NRQ	NRQ
pthread_sigmask()	MAN	NRQ	NRQ
pthread_testcancel()	MAN	NRQ	NRQ

<END>

4.10 Remaining tables in B.5

4.10.1 Principle

In compliance with §3.2.2.2, create the ULw in all tables of B.5 not mentioned in previous amendments and turn them to NRQ.

4.10.2 Proposed Text

As specified above.

4.11 Capture ULwAEP requirements in B.6

4.11.1 Principle

Add the ULwAEP column in all the B.6 tables and provide appropriate MAN/ NRQ indications in conformance with the B.5 content relative to ULwAEP.

4.11.2 Proposed text

None.

Editorial and consistency checking activity subordinate to JTRS Standards decision on previous amendments.

END OF DOCUMENT