

Recognized as an American National Standard (ANSI)

IEEE Std 896.4a-1995
Supplement to IEEE Std 896.4-1993

IEEE Standard for Conformance Test Requirements for Futurebus+®— Errata, Corrections, and Clarifications

Sponsor
**Bus Architecture Standards Committee
of the
IEEE Computer Society**

Approved 12 December 1995
IEEE Standards Board

Approved 16 July 1996
American National Standards Institute

Abstract: Errata, corrections, and clarifications to IEEE Std 896.4-1993, as well as a new diskette version of the conformance test suite to replace the original test suite distributed with the 1993 publication of the standard, are provided. IEEE Std 896.4-1993 defines a suite of tests that govern the testing of the requirements of Futurebus+® as defined originally in IEEE Std 896.1-1991 and IEEE Std 896.2-1991. This supplement also updates IEEE Std 896.4-1993 so that the requirements of ISO/IEC 10857: 1994 [ANSI/IEEE Std 896.1, 1994 Edition] and IEEE Std 896.2a-1994 are tested.

Keywords: application environment profile (AEP), BICS pro forma, BIXIT pro forma, block copy, implementation under test (IUT), interactive BICS

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA
Copyright © 1996 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 1996. Printed in the United States of America
ISBN 1-55937-756-9

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE that have expressed an interest in participating in the development of the standard.

Use of an IEEE Standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of all concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying all patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; (508) 750-8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

(This introduction is not part of IEEE Std 896.4a-1995, IEEE Standard for Conformance Test Requirements for Futurebus+®—Errata, Corrections, and Clarifications.)

The following is a list of participants in the P896.4a working group:

Stephen J. Cecil, *Chair*
Michael L. Roby, *Vice Chair*
Kim D. Burris, *Technical Editor*
David Scott Thomas, *Secretary*

Mark D. Chestnutwood
Sam Duncan
Karl S. Franklin

David Hollenbeck
Joseph E. Jachim
Jeffrey Lear

Karl E. McClure, Jr.
Eric Walters

The following persons were on the balloting committee:

Ghassan A. Abbas
Malcolm J. Airst
Keith D. Anthony
Stephen E. Belvin
Christos Bezirtzoglou
Lim Boon Lum
John C. Brightwell
Charles Brill
Kim D. Burris
Stephen J. Cecil
Mark D. Chestnutwood
Robert S. Crowder
Sourav K. Dutta

Roger D. Edwards
Wayne P. Fischer
Karl Franklin
Julio Gonzalez Sanz
Gerald E. Laws
Gary Manchester
Thanos Mentzelopoulos
Klaus-Dieter Mueller
J. D. Nicoud
Fred J. Orlando
Granville Ott
Mira Pauker
Delia Rankin

Michael L. Roby
John Rynearson
Frederick E. Sauer
Richard C. Spratt
Dongzhuang Su
David Scott Thomas
Michael G. Thompson
Robert C. Tripi
Eike G. Waltz
David L. Wright
Oren Yuen

When the IEEE Standards Board approved this standard on 12 December 1995, it had the following membership:

E. G. “Al” Kiener, *Chair*
Donald C. Loughry, *Vice Chair*
Andrew G. Salem, *Secretary*

Gilles A. Baril
Clyde R. Camp
Joseph A. Cannatelli
Stephen L. Diamond
Harold E. Epstein
Donald C. Fleckenstein
Jay Forster*
Donald N. Heirman
Richard J. Holleman

Jim Isaak
Ben C. Johnson
Sonny Kasturi
Lorraine C. Kevra
Ivor N. Knight
Joseph L. Koepfinger*
D. N. “Jim” Logothetis
L. Bruce McClung
Marco W. Migliaro

Mary Lou Padgett
John W. Pope
Arthur K. Reilly
Gary S. Robinson
Ingo Rüsçh
Chee Kiow Tan
Leonard L. Tripp
Howard L. Wolfman

*Member Emeritus

Also included are the following nonvoting IEEE Standards Board liaisons:

Satish K. Aggarwal

Steve Sharkey
Robert E. Hebner

Chester C. Taylor

Kristin M. Dittmann
IEEE Standards Project Editor

IEEE Standard for Conformance Test Requirements for Futurebus+®—Errata, Corrections, and Clarifications

1. Scope

This document contains errata, corrections, and clarifications to IEEE Std 896.4-1993 and a new diskette version of the conformance test suite to replace the original test suite distributed with the 1993 version of the standard. This supplement also updates IEEE Std 896.4-1993 so that the requirements of ISO/IEC 10857 :1994 are tested.

Several errata in this document pertain to the Bus Information Conformance Statement (BICS) pro forma in annex B. In cases where errata have deleted BICS entries or deleted whole BICS tables, entries or tables following the item deleted will not be renumbered by this supplement.

Changes are shown as additions to the text. In some cases, where changes involve corrections to existing text, the new text is underlined and deleted text is ~~struck through~~. Text shown in ***bold italics*** provides editing instructions necessary for incorporating these changes into the base document. After clause 2., the clause, subclause, table, and figure numbering corresponds to that of IEEE Std 896.4-1993.

2. Errata

5.1.5.1.1 Normal status

On pages 32–33, replace TEST_ID numbers 1.5.1.1.2.1–1.5.1.1.8.1 (the last five paragraphs) with the following:

TEST_ID = "1.5.1.1.2.1";

TEST_NAME = "Mask & Swap Lock Command, IUT Slave, Connected Lock \ Command Method, Normal.";

TEST_ID = "1.5.1.1.6.1";

TEST_NAME = "Compare & Swap Lock Command, IUT Slave, Connected Lock \ Command Method, Normal, Operands Match.";

TEST_ID = "1.5.1.1.6.2";

TEST_NAME = "Compare & Swap Lock Command, IUT Slave, Connected Lock \ Command Method, Normal, Operands Don't Match.";

TEST_ID = "1.5.1.1.7.1";

TEST_NAME = "Fetch & Add Big Endian Lock Command, IUT Slave, Connected \ Lock Command Method, Normal.";

Test ID = "1.5.1.1.8.1";

TEST_NAME = "Fetch & Add Little Endian Lock Command, IUT Slave, \ Connected Lock Command Method, Normal.";

5.1.10.1.1 Normal status

On pages 35–36, replace TEST_ID numbers 1.10.1.1.2.1–1.10.1.1.5.1 (the last five paragraphs) with the following:

TEST_ID = "1.10.1.1.2.1";

TEST_NAME = "Write Partial Locked Transaction, Mask & Swap Lock \ Command, IUT Slave, Connected Lock Command Method, Normal.";

TEST_ID = "1.10.1.1.3.1";

TEST_NAME = "Write Partial Locked Transaction, Compare & Swap Lock \ Command, IUT Slave, Connected Lock Command Method, Normal - Match.";

TEST_ID = "1.10.1.1.3.2";

TEST_NAME = "Write Partial Locked Transaction, Compare & Swap Lock \ Command, IUT Slave, Connected Lock Command Method, Normal - Operands \ Don't Match.";

TEST_ID = "1.10.1.1.4.1";

TEST_NAME = "Write Partial Locked Transaction, Fetch and Add Big \ Endian Lock Command, IUT Slave, Connected Lock Command Method, Normal.";

TEST_ID = "1.10.1.1.5.1";

TEST_NAME = "Write Partial Locked Transaction, Fetch and Add Little \ Endian Lock Command, IUT Slave, Connected Lock Command Method, Normal.";

5.1.11.2 IUT master

On page 37, add the following subclauses after 5.1.11.2.1. Subclauses 5.1.11.2.2–5.1.11.2.12 have no tests, but have been added for editorial consistency.

5.1.11.2.2 Transaction error (TE)

There are no tests for this category.

5.1.11.2.3 Beat error (BE)

There are no tests for this category.

5.1.11.2.4 Wait (WT)

There are no tests for this category.

5.1.11.2.5 Busy (BS)

There are no tests for this category.

5.1.11.2.6 End of data (ED)

There are no tests for this category.

5.1.11.2.7 Broadcast (BC)

There are no tests for this category.

5.1.11.2.8 Transaction flag (TF)

There are no tests for this category.

5.1.11.2.9 Intervention (IV)

There are no tests for this category.

5.1.11.2.10 Split response (SR)

There are no tests for this category.

5.1.11.2.11 Unselected

There are no tests for this category.

5.1.11.2.12 Self-detected condition

There are no tests for this category.

5.1.11.2.13 Miscellaneous

In the following test category, the fifth field of the function name and TEST_ID has the following meaning:

- 1) —Full data width lock command
- 2) —Partial data width lock command

The sixth field of the function name and TEST_ID has the following meaning:

- 1) —Compare and swap lock
- 2) —Mask and swap lock
- 3) —Fetch and add big lock
- 4) —Fetch and add little lock

TEST_ID = "1.11.2.13.1.1";

TEST_NAME = "Mask & Swap Lock Command, IUT Master, Split Lock \ Command Method, Normal.";

TEST_ID = "1.11.2.13.1.2";

TEST_NAME = "Compare & Swap Lock Command, IUT Master, Split Lock \ Command Method, Normal.";

TEST_ID = "1.11.2.13.1.3";
TEST_NAME = "Fetch & Add Big Endian Lock Command, IUT Master, Split \ Lock Command Method, Normal.";

TEST_ID = "1.11.2.13.1.4";
TEST_NAME = "Fetch & Add Little Endian Lock Command, IUT Master, \ Split Lock Command Method, Normal.";

TEST_ID = "1.11.2.13.2.1";
TEST_NAME = "Write Partial Locked, Mask & Swap Lock Command, IUT \ Master, Split Lock Command Method, Normal.";

TEST_ID = "1.11.2.13.2.2";
TEST_NAME = "Write Partial Locked, Compare & Swap Lock Command, \ IUT Master, Split Lock Command Method, Normal - Match.";

TEST_ID = "1.11.2.13.2.3";
TEST_NAME = "Write Partial Locked, Fetch & Add Big Endian Lock \ Command, IUT Master, Split Lock Command Method, Normal.";

TEST_ID = "1.11.2.13.2.4";
TEST_NAME = "Write Partial Locked, Fetch & Add Little Endian Lock \ Command, IUT Master, Split Lock Command Method, Normal.";

5.1.15.1.2 Transaction error (TE)

On page 42, replace the entire subclause with the following:

Test_ID = "1.15.1.2.1"
TEST_NAME = "Write Invalid, IUT Memory Slave, Illegal Transaction Flag";

Test_ID = "1.15.1.2.2"
TEST_NAME = "Write Invalid, IUT Memory Slave, Illegal Transaction Flag w/ SR*";

Test_ID = "1.15.1.2.3"
TEST_NAME = "Write Invalid, IUT Memory Slave, Illegal Intervention";

Test_ID = "1.15.1.2.4"
TEST_NAME = "Write Invalid, IUT Memory Slave, Illegal Intervention w/ TF*";

Test_ID = "1.15.1.2.5"
TEST_NAME = "Write Invalid, IUT Memory Slave, Illegal Intervention w/ SR*";

5.2.2.8 Phase 4 timeout

On page 65, change TEST_ID numbers 2.2.8.1, 2.2.8.4, 2.2.8.5, and 2.2.8.6 as follows:

Test_ID = "2.2.8.1"; ~~**Test_ID** = "2.2.8.1";~~
TEST_NAME = "IUT reports a phase 4 timeout as a bystander when the \ transaction generator does not assert ar* after winning a one pass \ competition; transaction generator is the arbitration owner";

Test_ID = "2.2.8.4";
TEST_NAME = "~~After reset,~~ IUT reports a phase 4 timeout when the \

transaction generator does not assert ar* in pass two after \ winning a two pass competition”;

Test_ID = “2.2.8.5”;

TEST_NAME = ~~“After reset,~~ IUT reports a phase 4 timeout when the \ transaction generator does not assert ar* in pass two after \ winning a two pass competition; IUT was a losing competitor \ in pass 1. ”;

Test_ID = “2.2.8.6”;

TEST_NAME - ~~“After reset,~~ IUT reports a phase 4 timeout when the \ transaction generator does not assert ar* in pass two after winning \ a two pass competition; IUT was a losing competitor in pass 2”;\

5.3.1 Central arbitration requestor

On page 67, add tests 3.1.11 and 3.1.12 as follows:

Test_ID = “3.1.11”;

Test_Name = “Basic Requesting Module Test with System Reset, No PE*”;

Test_ID = “3.1.12”;

Test_NAME = “Basic Requesting Module Test with Bus Initialize”;

5.4 Message passing

On page 68, delete the second paragraph of text.

5.4.1 Frame level tests

On page 68, add the following text as a new paragraph at the beginning of the subclause:

The tests for this clause can be found in files “5_4_1_1.c,” “5_4_1_2.c,” and “5_4_1_3.c” on one of the accompanying diskettes.

5.4.2 Message level tests

On page 69, add the following text as a new paragraph at the beginning of the subclause:

The tests for this clause can be found in files “5_4_2_1.c,” “5_4_2_2.c,” and “5_4_2_3.c” on one of the accompanying diskettes.

5.5 CSR, bus/system management

On page 71, modify the text as shown:

The tests for this clause can be found in ~~file “5_5.c”~~ files “5_5_1.c,” “5_5_2_1.c,” “5_5_2_2.c,” “5_5_3.c,” “5_5_4.c,” and “5_5_5.c” on one of the accompanying diskettes. Tests that are included in that file are as follows:

5.5.1 Bus system management

On page 71, add the following text as a new paragraph at the beginning of the subclause:

The tests for this clause can be found in file “5_5_1.c” on one of the accompanying diskettes.

5.5.1.1 ~~4.5.1.1~~ Power up and system reset

Delete “4.5.1.1” in the subclause title as shown above.

Add “ as a Central Arbitration Requester.” to test 5.1.1.3 as follows:

TEST_ID = “5.1.1.3”;

TEST_NAME = “IUT Detects System Reset as a Central Arbitration Requester.”;

Add test 5.1.1.4 as follows:

TEST_ID = “5.1.1.4”;

TEST_NAME = “IUT Detects System Reset as a Distributed Arbitration Requester.”;

5.5.2 Core CSRs

On page 71, add the following text as a new paragraph to this subclause:

The tests for this clause can be found in files “5_5_2_1.c” and “5_5_2_2.c” on one of the accompanying diskettes.

5.5.3 Futurebus+ specific CSRs

On page 75, add the following text as a new paragraph to this subclause:

The tests for this clause can be found in files “5_5_3_1.c” and “5_5_3.c” on one of the accompanying diskettes.

5.5.4 ROM tests

On page 76, add the following text as a new paragraph to this subclause:

The tests for this clause can be found in file “5_5_4.c” on one of the accompanying diskettes.

5.5.5 Tests for unimplemented registers

On page 76, add the following text as a new paragraph to this subclause:

The tests for this clause can be found in file “5_5_5.c” on one of the accompanying diskettes.

Also, change

TEST_ID = 5_5;

to

TEST_ID = 5.5;

6.2.1 Timing monitors

On page 91, add new subclauses 6.2.1.4, 6.2.1.5, 6.2.1.6, and 6.2.1.7 as follows:

6.2.1.4 Release of AP* and assertion of AR* out of sequence timing monitor

This timing monitor is used in tests 2.1.2.5 and 2.1.4.5 when the release of AP* occurs after the assertion of AR*. Tests call this monitor by assigning the label AP_AR_REVERSED to one of the monitor_pass variables in the response structure.

```
void ap_ar_reversed_timing () {
    Setup (AB[*], ~AR*, GE, 0)      /* ISO/IEC 10857 , 5.2.8.2, 5.2.9.2 */
    Stable(AQ*, AB[*], ~AP*)        /* ISO/IEC 10857 , 5.2.8.4, 5.2.9.4 */
    Hold(AB[*], ~AP*, GE, 0)        /* ISO/IEC 10857 , 5.2.8.5, 5.2.9.5 */
    Setup(~AB[*], ~AQ*, GE, 0)      /* ISO/IEC 10857 , 5.2.5, COMPETE */
    Setup(AC[*], ~AP*, GE, 0)        /* ISO/IEC 10857 , 5.2.8.4, 5.2.9.4 */
    Stable(~AP*, AC[*], ~AQ*)        /* ISO/IEC 10857 , 5.2.5, PHASE_5 */
    Hold(AC[*], ~AQ*, GE, 0)        /* ISO/IEC 10857 , 5.2.8.1, 5.2.9.1 */
}
```

6.2.1.5 Release of AQ* and assertion of AP* out of sequence, pass 1, timing monitor

This timing monitor is used in tests 2.1.2.5 and 2.1.4.5 when the assertion of AP* occurs before the release of AQ*, at the end of the pass in which this monitor is used. Tests call this monitor by assigning the label AQ_AP_REVERSED_I to one of the monitor_pass variables in the response structure. The timing for the second pass is covered in the monitors of section 6.2.1.6 for test 2.1.4.5 and section 6.1.2.7 for test 2.1.2.5.

NOTE — Throughout these monitors, <1> and <2> are used to distinguish between the first <1> and second <2> times that a signal has been asserted or released during that monitor.

```
void aq_ap_reversed_time_1 () {
    Setup(AB[*], ~AP*, GE, 0)        /* ISO/IEC 10857 , 5.2.8.2, 5.2.9.2 */
    Stable(AQ*, AB[*], ~AP*)        /* ISO/IEC 10857 , 5.2.8.4, 5.2.9.4 */
    Hold(AB[*], ~AP*, GE, 0)        /* ISO/IEC 10857 , 5.2.8.5, 5.2.9.5 */
    Setup(~AB[*], AP*<2>, GE, 0)      /* ISO/IEC 10857 , 5.2.5, COMPETE */
    Setup(AC[*], ~AP*, GE, 0)        /* ISO/IEC 10857 , 5.2.8.4, 5.2.9.4 */

    /* This monitor shall also save the state of AC[*] as AC[*]_previous for use in
    the AQ_AP_REVERSED_2 and AQ_AP_REV2_W_TRANSFER timing monitors. AC[*] shall
    remain stable until AQ* is released in that monitor. */
}
```

6.2.1.6 Release of AQ* and assertion of AP* out of sequence, pass 2, timing monitor

This timing monitor is used in test 2.1.4.5 when the assertion of AP* occurs before the release of AQ*, at the beginning of the pass in which this monitor is used. Tests call this monitor by assigning the label AQ_AP_REVERSED_2 to one of the monitor_pass variables in the response structure. The timing for the first pass is covered in the monitors of section 6.2.1.5.

NOTE — Throughout these monitors, <1> and <2> are used to distinguish between the first <1> and second <2> times that a signal has been asserted or released during that monitor.

```
void aq_ap_reversed_2_time () {
```

```

Hold(AC[*]_previous, ~AQ*<1>, GE, 0)      /* ISO/IEC 10857 , 5.2.5, PHASE_0 */
Setup(AB[*], ~AR*, GE, 0)                  /* ISO/IEC 10857 , 5.2.8.2, 5.2.9.2 */
Stable(AQ*, AB[*], ~AP*)                   /* ISO/IEC 10857 , 5.2.8.4, 5.2.9.4 */
Hold(AB[*], ~AP*, GE, 0)                   /* ISO/IEC 10857 , 5.2.8.5, 5.2.9.5 */
Setup(~AB[*], ~AQ*<2>, GE, 0)              /* ISO/IEC 10857 , 5.2.5, COMPETE */
Setup(AC[*], ~AP*, GE, 0)                  /* ISO/IEC 10857 , 5.2.8.4, 5.2.9.4 */
Stable(AR*, AC[*], ~AQ*<2>)               /* ISO/IEC 10857 , 5.2.8.6, 5.2.9.6 */
Hold(AC[*], ~AQ*<2>, GE, 0)               /* ISO/IEC 10857 , 5.2.8.1, 5.2.9.1 */
Setup(~AC[*], AP*next, GE, 0)             /* ISO/IEC 10857 , 5.2.8.1, 5.2.9.1 */

}

```

6.2.1.7 Release of AQ* and assertion of AP* out of sequence with transfer of tenure to the master elect, pass 2, timing monitor

This timing monitor is used in test 2.1.2.5 when the assertion of AP* occurs before the release of AQ*, at the beginning of the pass in which this monitor is used. Tests call this monitor by assigning the label AQ_AP_REV2_W_TRANSFERENCE to one of the monitor_pass variables in the response structure. The timing for the first pass is covered in the monitors of section 6.2.1.5.

NOTE — Throughout these monitors, <1> and <2> are used to distinguish between the first <1> and second <2> times that a signal has been asserted or released during that monitor.

```
void aq_ap_rev2_time_w_transfer () {
```

```

Hold(AC[*]_previous, ~AQ*<1>, GE, 0)      /* ISO/IEC 10857 , 5.2.5, PHASE_0 */
Setup(AB[*], ~AR*, GE, 0)                  /* ISO/IEC 10857 , 5.2.8.2, 5.2.9.2 */
Stable(AQ*, AB[*], ~AP*)                   /* ISO/IEC 10857 , 5.2.8.4, 5.2.9.4 */
Hold(AB[*], ~AP*, GE, 0)                   /* ISO/IEC 10857 , 5.2.8.5, 5.2.9.5 */
Setup(~AB[*], ~AQ*<2>, GE, 0)              /* ISO/IEC 10857 , 5.2.5, COMPETE */
Setup(AC[*], ~AP*, GE, 0)                  /* ISO/IEC 10857 , 5.2.8.4, 5.2.9.4 */
Stable(AR*, AC[*], ~AQ*<2>)               /* ISO/IEC 10857 , 5.2.8.6, 5.2.9.6 */
Setup(~AD[*], AR*, GE, 0)                 /* ISO/IEC 10857 , 5.2.8.5 */
Setup(~CA[*], AR*, GE, 0)                 /* ISO/IEC 10857 , 5.2.8.5 */
Setup(~CM[*], AR*, GE, 0)                 /* ISO/IEC 10857 , 5.2.8.5 */
Setup(~ST[*], AR*, GE, 0)                 /* ISO/IEC 10857 , 5.2.8.5 */

} "

```

6.2.2 Protocol monitors

On page 93, add subclauses 6.2.2.8–6.2.2.11 as follows:

6.2.2.8 Release of AP* and assertion of AR* out of sequence protocol monitor

This protocol monitor is used in tests 2.1.2.5 and 2.1.4.5 when the release of AP* occurs after the assertion of AR*. Tests call this monitor by assigning the label AP_AR_REVERSED to one of the monitor_pass variables in the response structure.

```
void ap_ar_reversed_protocol (int competition_settling_time) {
```

```

Delay(PHASE_0, AP*<1>, GE, 0) /* ISO/IEC 10857 , 5.2.8.1, 5.2.9.1 */
Delay(AP*, ~AR*, GE, 0)      /* ISO/IEC 10857 , 5.2.8.2, 5.2.9.2 */
Delay(~AR*, AQ*, GE, competition_settling_time)

```