



# TA Document 1999006

## AV/C Bulletin Board Type Specification – Resource Schedule Board 1.0a

September 24, 1999

**Sponsored by:**

1394 Trade Association

**Approved for Release by:**

This document has been approved for release by the 1394 Trade Association Board of Directors.

**Abstract:**

The Resource Schedule Board is a type of board in AV/C Bulletin Board Subunit. It provides information for finding resource scheduling conflicts in advance. A device, which has its own scheduled action, can write its own use schedule the device's resource. Other controllers can then detect other's use schedule by reading this board. This document defines the data structures and command sets, which are specific for the Resource Schedule Board. The general data structures and command sets are defined in the Bulletin Board Subunit General Specification and AV/C General Specification.

**Keywords:**

Audio, Video, 1394, Digital, Interface, Bulletin Board, Scheduled Action.

---

Copyright © 1996-2000 by the 1394 Trade Association.  
Regency Plaza Suite 350, 2350 Mission College Blvd., Santa Clara, CA 95054, USA  
<http://www.1394TA.org>  
All rights reserved.

Permission is granted to members of the 1394 Trade Association to reproduce this document for their own use or the use of other 1394 Trade Association members only, provided this notice is included. All other rights reserved. Duplication for sale, or for commercial or for-profit use is strictly prohibited without the prior written consent of the 1394 Trade Association.

**1394 Trade Association Specifications** are developed within Working Groups of the 1394 Trade Association, a non-profit industry association devoted to the promotion of and growth of the market for IEEE 1394-compliant products. Participants in working groups serve voluntarily and without compensation from the Trade Association. Most participants represent member organizations of the 1394 Trade Association. The specifications developed within the working groups represent a consensus of the expertise represented by the participants.

Use of a 1394 Trade Association Specification is wholly voluntary. The existence of a 1394 Trade Association Specification is not meant to imply that there are not other ways to produce, test, measure, purchase, market or provide other goods and services related to the scope of the 1394 Trade Association Specification. Furthermore, the viewpoint expressed at the time a specification 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 specification. Users are cautioned to check to determine that they have the latest revision of any 1394 Trade Association Specification.

Comments for revision of 1394 Trade Association Specifications are welcome from any interested party, regardless of membership affiliation with the 1394 Trade Association. 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 about the meaning of specifications in relationship to specific applications. When the need for interpretations is brought to the attention of the 1394 Trade Association, the Association will initiate action to prepare appropriate responses.

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

Editor, 1394 Trade Association  
Regency Plaza Suite 350  
2350 Mission College Blvd.  
Santa Clara, Calif. 95054, USA

1394 Trade Association Specifications are adopted by the 1394 Trade Association without regard to patents which may exist on articles, materials or processes, or to other proprietary intellectual property which may exist within a specification. Adoption of a specification by the 1394 Trade Association does not assume any liability to any patent owner or any obligation whatsoever to those parties who rely on the specification documents. Readers of this document are advised to make an independent determination regarding the existence of intellectual property rights which may be infringed by conformance to this specification.

**Table of Contents**

- 1. Preface ..... 7
  - 1.1 Purpose and Scope ..... 7
    - 1.1.1 Purpose ..... 7
    - 1.1.2 Scope ..... 7
- 2. References ..... 8
  - 2.1 Contact Information ..... 8
    - 2.1.1 1394 Trade Association (1394 TA) ..... 8
    - 2.1.2 1394 Trade Association Specification ..... 8
- 3. Change History ..... 9
- 4. Definitions and Abbreviations ..... 10
  - 4.1 Conformance glossary ..... 10
  - 4.2 Technical glossary ..... 10
- 5. The Resource Schedule Board Model ..... 11
  - 5.1 Function ..... 11
  - 5.2 Structure Model and definition ..... 11
    - 5.2.1 posting device model ..... 11
    - 5.2.2 Target Device Model ..... 12
  - 5.3 Control Model ..... 12
    - 5.3.1 Basic Control ..... 13
    - 5.3.2 Scheduling Data ..... 13
    - 5.3.3 Conflicting Schedules ..... 15
    - 5.3.4 Maintaining a Resource Scheduling Entry ..... 15
    - 5.3.5 Deleting a resource schedule entry ..... 16
    - 5.3.6 Clock considerations ..... 16
- 6. Bulletin Board Subunit Identifier Descriptor ..... 18
  - 6.1 The Resource Schedule Board type specific values ..... 18
- 7. Information List Descriptor ..... 19
  - 7.1 The Resource Schedule Board type specific values ..... 19
- 8. Information Entry Descriptor ..... 20
  - 8.1 The Resource Schedule Board type specific values ..... 20
    - 8.1.1 child\_list ..... 20
    - 8.1.2 object\_ID ..... 20
    - 8.1.3 Resource schedule entry (entry\_specific\_information fields) ..... 20
    - 8.1.4 entry\_specific\_information field descriptions ..... 21
    - 8.1.5 start\_time fields ..... 21
    - 8.1.6 duration fields ..... 21
    - 8.1.7 Repeat information ..... 22
    - 8.1.8 Subunit\_resource\_info\_block (8900<sub>16</sub>) ..... 25
- 9. Board type specific command support level ..... 27
- 10. Resource Schedule Board profiles ..... 28
- Annex A: Resource schedule entry high level view (informative) ..... 31



A.1 Resource schedule entry ..... 31

Annex B: Scheduling Conflicts..... 32

    B.1 Posting device schedules time conflicting with another posting device’s entry ..... 32

    B.2 Devices Exceeding Duration ..... 32

    B.3 Conflicts with events using no duration ..... 33

**List of Figures**

Figure 5.1 – Posting device architecture *example* ..... 11

Figure 5.2 – Target device architecture..... 12

Figure 5.3 – Resource Schedule Board Control Model..... 13

Figure 5.4 – Resource Schedule Types ..... 14

Figure 5.5 – Conflicts based on clock time variances..... 17

Figure 8.1 – entry\_specific\_information fields for Resource Schedule Entries..... 20

Figure 8.2 – start\_time fields for Resource Schedule Entries..... 21

Figure 8.3 – duration fields for Resource Schedule Entries ..... 22

Figure 8.4 – repeat\_information fields for Weekly..... 23

Figure 8.5 – Setting number\_of\_events for a weekly repeat type..... 24

Figure 8.6 – repeat\_information fields for Interval..... 24

Figure 8.7 – Subunit\_Resource\_info\_block ..... 25

Figure A.1 – Resource schedule entry high level view ..... 31

Figure B.1 – Simple Scheduling Conflict ..... 32

Figure B.2 – Duration exceeding original duration conflict..... 33

Figure B.3 – Duration unspecified conflict ..... 34

## List of Tables

Table 6.1 – Resource Schedule Board type specific value assignments ..... 18  
Table 7.1 – Resource Schedule Board type specific value assignments ..... 19  
Table 8.1 – repeat\_type value assignment ..... 23  
Table 9.1 – Board type specific command support level ..... 27  
Table 10.1 – Bulletin Board Subunit Profile Value Assignment ..... 28

## 1. Preface

### 1.1 Purpose and Scope

This document defines the Resource Schedule Board, which is one type of board contained in the AV/C Bulletin Board Subunit. The purpose and scope of this board are summarized below.

#### 1.1.1 Purpose

The purpose of the Resource Schedule Board is to provide a location on a target device where other devices can post a schedule of the use of a target's resources. Devices on a 1394 network can then avoid resource-scheduling conflicts on that target device.

#### 1.1.2 Scope

This document defines the model for the Resource Schedule Board, Resource Schedule Board specific fields, and command set. The model and data structure in this specification follow the AV/C Digital Interface Command General Specification version 3.0 [R1], the Enhancements to AV/ General Specification 3.0 version 1.0 [R2], and the AV/C Bulletin Board Subunit General Specification [R3].

## 2. References

### 2.1 Contact Information

Much of the information in this document is preliminary and subject to change. Members of the AVWG are encouraged to review and provide inputs for this proposal. For document status updates, please contact:

Mari Horiguchi Workitem Project Leader  
i.LINK Development Department SUPC Sony Corporation  
Shinagawa Tec. Shinagawa INTERCITY C Tower W/9F  
2-15-3, Kounan, Minato-ku, Tokyo  
108-6201 Japan  
E-Mail: [marik@arch.sony.co.jp](mailto:marik@arch.sony.co.jp)  
Phone: +81-3-5769-5336  
Fax: +81-3-5769-5834

For technical comments, please contact:

Technical contributor  
Jon Brelin  
Sony, America  
3300 Zanker Rd  
San Jose, CA 95124  
E-Mail: [jon.brelin@am.sony.com](mailto:jon.brelin@am.sony.com)  
Phone: 408-955-6731  
Fax: 408-955-6236

Other technical officer  
Hisato Shima  
Sony, America  
3300 Zanker Rd.  
San Jose, CA 95124  
E-Mail: [hisato.shima@am.sony.com](mailto:hisato.shima@am.sony.com)  
Phone: 408-955-5524  
Fax: 408-955-6236

You can also submit comments using the 1394 TA reflector. Its web site address is:

[www.1394ta.org](http://www.1394ta.org)

Go to the members-only section and select the AVWG reflector. You will need to have signed up for the reflector.

The documents referenced herein may be obtained from the following organizations:

#### 2.1.1 1394 Trade Association (1394 TA)

The 1394 Trade Association can be contacted via the references provided on the cover page of this and all AV/C specification documents.

#### 2.1.2 1394 Trade Association Specification

- [R1] AV/C Digital Interface Command Set General Specification version 3.0.
- [R2] Enhancements to the AV/C General Specification 3.0, version 1.0.
- [R3] AV/C Bulletin Board Subunit General Specification version 1.0.



### 3. Change History

The following table shows all changes made to this document.

Date	Version	Changes	Editor(s)
July 1, 1999	1.0	Original Version	Mari Horiguchi, Jon Brelin
Mar 1, 2000	1.0a	Regenerated table of contents, updated figures 10.1, 10.2, and 10.3 to B.1, B.2, and B.3 respectively, and removed space in "A.1" for Figure A.1.	Jon Brelin

## 4. Definitions and Abbreviations

### 4.1 Conformance glossary

Several keywords are used to differentiate between different levels of requirements and optionality, as follows:

**4.1.1 expected:** A keyword used to describe the behavior of the hardware or software in the design models assumed by this specification. Other hardware and software design models may also be implemented.

**4.1.2 may:** A keyword that indicates flexibility of choice with no implied preference.

**4.1.3 shall:** A keyword indicating a mandatory requirement. Designers are required to implement all such mandatory requirements to ensure interoperability with other products conforming to this specification.

**4.1.4 should:** A keyword indicating flexibility of choice with a strongly preferred alternative. Equivalent to the phrase “is recommended.”

### 4.2 Technical glossary

**4.2.1 scheduled action:** A series of operations on a posting device that control a target device on a 1394 network according to a preset program and time schedule.

**4.2.2 program for a scheduled action:** A set of control instructions on a posting device that is executed on a target device by a scheduled action.

**4.2.3 event:** An operation that starts at a preset start time and finishes at a preset end time. A scheduled action contains one or more events.

**4.2.4 Resource Schedule Board:** (a) A type of AV/C Bulletin Board. (b) An Information List Descriptor that contains one or more Information Entry Descriptors for (a). The Information Entry Descriptor contains scheduling information, or a resource schedule entry, which is described in this document.

**4.2.5 Resource schedule entry:** The scheduling information particular to the Resource Schedule Board that is placed within the *entry\_specific\_information* fields in an Information Entry Descriptor.

## 5. The Resource Schedule Board Model

### 5.1 Function

The Resource Schedule Board is a type of board in the AV/C Bulletin Board Subunit that uses the two-tier structure as shown in the AV/C Bulletin Board Subunit General Specification version 1.0 [R3]. It provides devices with information for avoiding resource-scheduling conflicts. By writing to the board, a posting device can post a time to use one or more resources in the target where the Bulletin Board Subunit is located. Subsequently, other controllers can read the board to prevent double-scheduling resources.

The Resource Schedule Board does not lock a resource. It just provides scheduling information. A device should check the Resource Schedule to avoid scheduling conflicts before scheduling a resource. It is the responsibility of the posting device to prevent schedule conflicts because the Resource Schedule Board does not detect them.

The resources specified in a resource schedule can only be the resources in the device that contains the Resource Schedule Board.

### 5.2 Structure Model and definition

The target device contains the Resource Schedule Board and has resources used by posting devices. The posting device and target device models are described in the next clauses.

#### 5.2.1 posting device model

The posting device has the information necessary to post a resource schedule entry. The internal data structure in the posting device is application dependent. It is assumed that the posting device knows the current time, and that all controllers on the network exist in the same time zone. The following figure, however, provides a general guideline of the structure and the information the posting device will need to supply to the Resource Schedule Board. Note that only scheduling data is written to the Resource Schedule Board. Program data is used only when operating the resource.

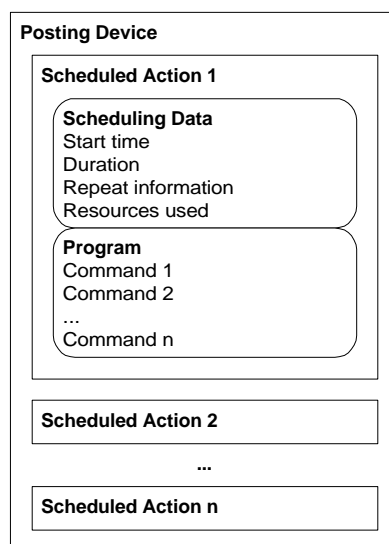


Figure 5.1 – Posting device architecture example

## 5.2.2 Target Device Model

The target device contains the Resource Schedule Board that any posting device can write to. The “Resource Schedule Board” uses the Information List Descriptor as given in reference [R3]. It contains a collection of Resource Schedule Entries that use Information Entry Descriptors also given in reference [R3]. Though the Bulletin Board Subunit supports child lists, the Resource Schedule Board has none. This Board is listed with other boards in the Bulletin Board Subunit’s Subunit Identifier Descriptor.

The resource schedule entry contains the following information.

- Time information: Start time/ Duration/ Repeat information
- Resource information: subunit, etc...

The target device model is illustrated in the following figure.

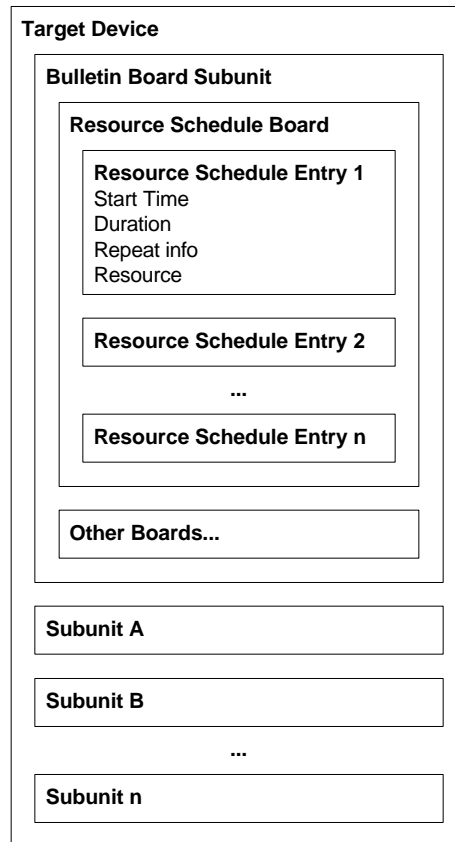


Figure 5.2 – Target device architecture

A resource schedule entry may contain one or more events. An event is an operation that starts at a preset start time and finishes at a preset end time. The number of events is determined by the repeat information in the entry.

## 5.3 Control Model

The control model explains how the Resource Schedule Board is used by a controlling application. In the Resource Schedule Board control model, the posting device has the information source, which includes scheduling data and a program for the scheduled action. The following sections explain basic control using this information.

### 5.3.1 Basic Control

When a user creates a scheduled action in a device, the device posts (or writes) the information from the scheduled action to one or more Resource Schedule Entries in the Resource Schedule Board. The posting device should update the resource schedule entry if the schedule changes. At the scheduled start time, the posting device executes its program, which controls the specified subunit. This is illustrated in the following figure.

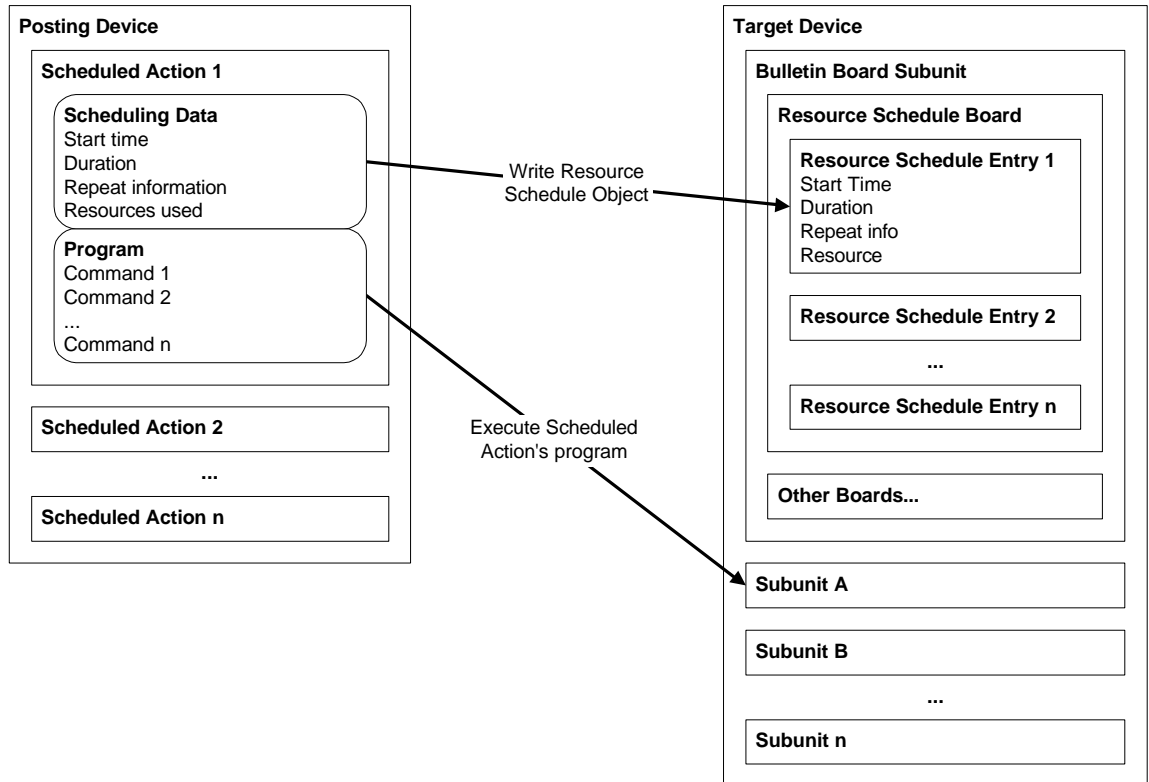


Figure 5.3 – Resource Schedule Board Control Model

### 5.3.2 Scheduling Data

Scheduling data consists of a start time, duration, repeat information, and the resource(s) that the program uses. By combining these fields, there are four types of resource schedules possible, which differ based on their repeat and duration information.

- Type A: Schedules where an event occurs once. This is accomplished by specifying a start time, a duration, and no repeat information.
- Type B: Schedules where events occur at specified intervals (hours, minutes, or seconds). This is accomplished by specifying a start time, a duration, and interval repeat information.
- Type C: Schedules where events occur on specified days of the week. This is accomplished by specifying a start time, a duration, and weekly repeat information.
- Type D: Schedules where an event occurs with no specified duration. This is accomplished by specifying a start time, no duration, and no repeat information.

Note that each resource schedule entry defines one or several repeated events. The Resource Schedule does not store each event separately in memory. It is up to the reading application to determine the start and end

times of each event in each resource schedule entry. To get a complete “picture” of the schedule, the posting device needs to read and interpret every entry in the Resource Schedule Board.

The four schedule types are illustrated in the following figure.

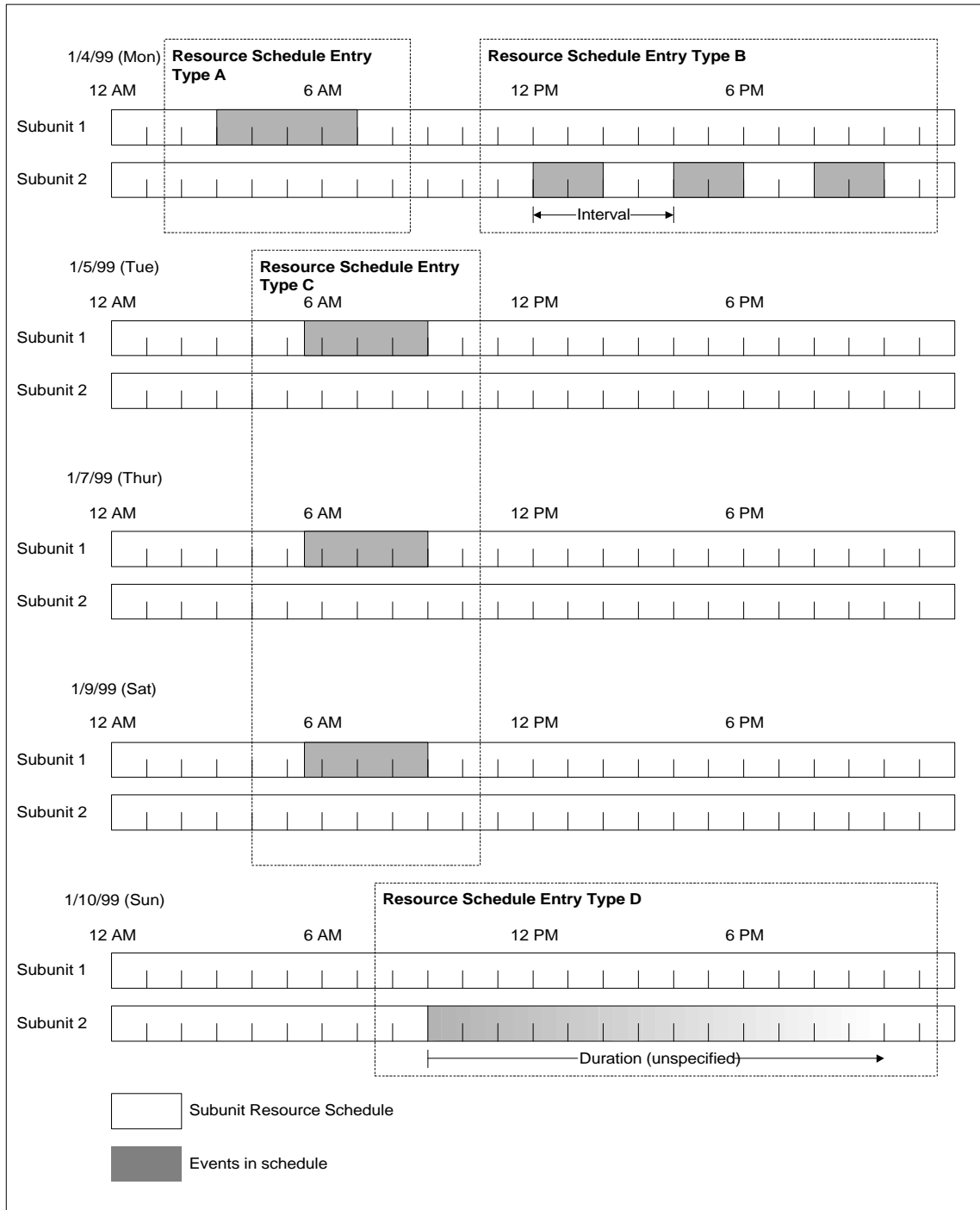


Figure 5.4 – Resource Schedule Types

### 5.3.3 Conflicting Schedules

It is the responsibility of the posting device's application to check for schedule conflicts by reading all the individual resource schedule entries in the Resource Schedule Board prior to posting its own schedule. The handling of conflicting schedules is application dependent, and out of the scope of this specification. The posting device's implementation should interact with the user to resolve or warn the user of a conflict. A posting device may post conflicting schedules to a resource schedule board if user intends it.

To prevent the resource from accepting commands from two different posting devices at the same time, a posting device can use the RESERVE command to reserve the resource.

### 5.3.4 Maintaining a Resource Scheduling Entry

Clauses 5.3.4.1 - 5.3.4.2 enable other controllers to know that the posting device is actively running its schedule.

#### 5.3.4.1 Maintaining repeated schedule entries

##### 5.3.4.1.1 Weekly repeat type

When an event is completed in a repeated schedule based on a weekly repeat type, the *start\_time* fields shall be updated to the next event's date, and the *number\_of\_events* field shall be updated as follows:

$$number\_of\_events_{new} = number\_of\_events_{old} - 1 \quad (5.1)$$

##### 5.3.4.1.2 Interval repeat type

When an event is completed in a repeated schedule based on an interval repeat type, the *start\_time* fields shall be updated for the next repeated event in the following manner:

$$start\_time_{new} = start\_time_{old} + interval \quad (5.2)$$

The *number\_of\_events* field shall also be updated according to equation 5.1.

##### 5.3.4.1.3 Both repeat types

For both repeat types, at the end of the last event (when *number\_of\_events* = 1), the controller shall delete the resource schedule entry.

If these values are not updated, it is assumed that the posting device was not present at the end of the last event.

#### 5.3.4.2 Maintaining schedule entries with no duration

For an event without a duration, the posting device shall update the *start\_time* fields at hour intervals with the current time. If these fields are not updated within one hour, it is assumed that the posting device is no longer present. The entry can then be deleted by another controller freeing the schedule.

### 5.3.4.3 Maintaining future schedule entries

If the user decides to change the Scheduling Data prior to running the Program, the posting device shall update the schedule in the Resource Schedule Board.

### 5.3.5 Deleting a resource schedule entry

When the program finishes, or is cancelled by the user, the posting device shall delete the resource schedule entry.

A controller shall not delete another's entry if that entry is valid. However, any controller can delete another's resource schedule entry under the following conditions:

- 1) The resource schedule entry is expired.
  - a) An entry's end time + five minutes is before the current time. The end time is determined by the following formula:

$$\text{End Time} = \text{start\_time} + \text{duration}$$

- 2) The resource schedule entry contains one of the following invalid values.
  - a) The start time is one year ahead. This provision helps to conserve memory usage.
  - b) A start time field contains a non-existent date and time.
  - c) *start\_time*, *duration*, or *interval* contains invalid BCD values.
  - d) If the repeat type is *Interval*, and the duration value is larger than the interval value.
  - e) The duration value is unspecified, and a repeat value is specified.
  - f) A start time field's date field does not correspond to a weekly flag's date.
  - g) An entry where the subunit is unspecified, or incorrectly specified.
- 3) The schedule entry is not maintained properly.
  - a) If the duration field is specified, the current time is five minutes past the sum of the start time and the duration.
  - b) If the duration field is not specified, the current time is five minutes past the sum of the start time and one hour.

### 5.3.6 Clock considerations

The following timing conditions may occur between posting devices.

- 1) Two posting device's clocks may be off more than five minutes from each other. If this occurs, a posting device can inadvertently delete another posting device's entry.
- 2) When events are scheduled back-to-back by two posting devices with clock inaccuracies, it is possible to have a schedule conflict. The following figure illustrates this phenomenon:



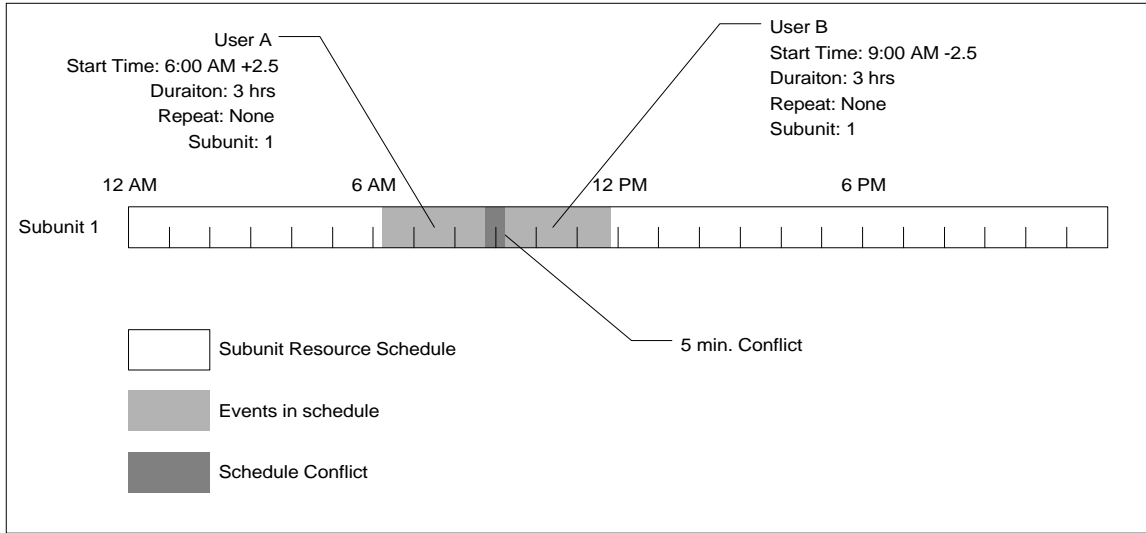


Figure 5.5 – Conflicts based on clock time variances

## 6. Bulletin Board Subunit Identifier Descriptor

The Bulletin Board Subunit Identifier Descriptor is defined in the Bulletin Board Subunit General specification [R3]. This section defines the Resource Schedule Board specific values and fields in the Bulletin Board Subunit Identifier Descriptor.

### 6.1 The Resource Schedule Board type specific values

The Resource Schedule Board specific values for the fields in the AV/C Bulletin Board Subunit Identifier Descriptor are as follows.

**Table 6.1 – Resource Schedule Board type specific value assignments**

Field Name	Value	Description
root_object_list_ID	1001 <sub>16</sub>	list_ID for Resource Schedule Board
supported_board_type	01 <sub>16</sub>	Resource Schedule Board type
supported_board_type_version	10 <sub>16</sub>	Version of the Resource Schedule Board
implementation_profile_ID	XX <sub>16</sub>	Implementation profile of Resource Schedule Board. See chapter 10.
supported_board_type_dependent_information_length	0000 <sub>16</sub>	Resource Schedule Board specific information length

## 7. Information List Descriptor

The Information List Descriptor is defined in the Bulletin Board Subunit General specification [R3]. This section defines the Resource Schedule Board specific values and fields.

### 7.1 The Resource Schedule Board type specific values

The Resource Schedule Board specific values for the fields in the AV/C Bulletin Board Information List Descriptor are as follows.

**Table 7.1 – Resource Schedule Board type specific value assignments**

Field Name	Value	Description
board_type	01 <sub>16</sub>	Resource Schedule Board
board_type_dependent_information_length	0000 <sub>16</sub>	Resource Schedule Board type dependent Info length

## 8. Information Entry Descriptor

The Information Entry Descriptor is defined in the Bulletin Board Subunit General specification [R3]. This section defines the Resource Schedule Board specific values and fields.

### 8.1 The Resource Schedule Board type specific values

The Resource Schedule Board specific values for the fields in the AV/C Bulletin Board Information Entry Descriptor are as follows. The posting device shall manage all entry specific information fields.

#### 8.1.1 child\_list

The Information Entry Descriptor has no child list.

#### 8.1.2 object\_ID

The posting device that owns the scheduled action’s program assigns the object\_ID. The *posting\_device\_ID* indicates Global Unique ID of the posting device and a unique *record\_ID*. See the Bulletin Board Subunit documentation [R3] for more details.

#### 8.1.3 Resource schedule entry (entry\_specific\_information fields)

The posting device shall manage the *entry\_specific\_information* fields, otherwise known as the resource schedule entry. The *entry\_specific\_information* fields for the Information Entry Descriptor are specified as follows.

Address_offset	Contents
00 <sub>16</sub>	non_info_block_length
01 <sub>16</sub>	
02 <sub>16</sub>	start_time
03 <sub>16</sub>	
04 <sub>16</sub>	
05 <sub>16</sub>	
06 <sub>16</sub>	
07 <sub>16</sub>	
08 <sub>16</sub>	
09 <sub>16</sub>	Duration
0A <sub>16</sub>	
0B <sub>16</sub>	
0C <sub>16</sub>	
0D <sub>16</sub>	repeat_information_length
0E <sub>16</sub>	repeat_information
:	
:	
:	Info blocks
:	

**Figure 8.1 – entry\_specific\_information fields for Resource Schedule Entries**

### 8.1.4 entry\_specific\_information field descriptions

**non\_info\_block\_length:** The *non\_info\_block\_length* field specifies the size, in bytes, of the non-info block fields down to and including the *repeat\_type\_dependent\_information*.

### 8.1.5 start\_time fields

The *start\_time* fields indicate the date and time when an event starts, using resources. For repeating events, the *start\_time* corresponds to the date and time when the next event starts. The value in the field is Binary Coded Decimal (BCD). It is indicated by local time. Each field is as follows.

Address_offset	Contents
00 <sub>16</sub>	year (msb)
01 <sub>16</sub>	year (lsb)
02 <sub>16</sub>	month
03 <sub>16</sub>	day
04 <sub>16</sub>	hour
05 <sub>16</sub>	minute
06 <sub>16</sub>	second

Figure 8.2 – start\_time fields for Resource Schedule Entries

#### 8.1.5.1 start\_time fields descriptions

**Year:** The *year* field is coded using 16 bits as 4 digits in 4 bit BCD (1999<sub>16</sub> – 9999<sub>16</sub>).

**Month:** The *month* field is coded using 8 bits as 2 digits in 4 bit BCD (01<sub>16</sub> – 12<sub>16</sub>).

**Day:** The *day* field is coded using 8 bits as 2 digits in 4 bit BCD (01<sub>16</sub> – 31<sub>16</sub>).

**Hour:** The *hour* field is coded using 8 bits as 2 digits in 4 bit BCD (00<sub>16</sub> – 23<sub>16</sub>).

**Minute:** The *minute* field is coded using 8 bits as 2 digits in 4 bit BCD (00<sub>16</sub> – 59<sub>16</sub>).

**Second:** The *second* field is coded using 8 bits as 2 digits in 4 bit BCD (00<sub>16</sub> – 59<sub>16</sub>).

#### 8.1.5.2 Specifying the start\_time fields

- 1) The fields shall comply to valid dates as given by a standard calendar.
- 2) The *start\_time* fields are updated according to the rules as given in 5.3.4 Maintaining a Resource Scheduling Entry.
- 3) If the posting device's time resolution is one minute, then it shall set the *seconds* field to 00<sub>16</sub>.

### 8.1.6 duration fields

*Duration* fields are used to specify the length of an event. Depending on user requirements, a posting device may or may not specify duration. The value in the field is Binary coded Decimal (BCD).

Address_offset	Contents	
00 <sub>16</sub>	Reserved (4 bits)	hours (msb)
01 <sub>16</sub>	hours (lsb)	
02 <sub>16</sub>	minutes	
03 <sub>16</sub>	seconds	

Figure 8.3 – duration fields for Resource Schedule Entries

### 8.1.6.1 duration fields descriptions

**Hour:** The *hour* fields are coded using 12 bits as 3 digits in 4 bit BCD (000<sub>16</sub> – 999<sub>16</sub>).

**Minutes:** The *minutes* field is coded using 8 bits as 2 digits in 4 bit BCD (00<sub>16</sub> – 59<sub>16</sub>).

**Seconds:** The *seconds* field is coded using 8 bits as 2 digits in 4 bit BCD (00<sub>16</sub> – 59<sub>16</sub>).

### 8.1.6.2 Specifying duration fields

The posting device is required to set and update the *duration* fields in the following manner.

- 1) Reserved fields should be set to 0<sub>16</sub>.
- 2) If the posting device's time resolution is one minute, it shall set the *seconds* field to the value 00<sub>16</sub>.
- 3) When the duration is unknown, the posting device shall set these fields to all 0. When the posting device knows the duration, it shall set this value to the *duration* fields.
- 4) The posting device shall maintain the duration fields according to clause 5.3.4.2 Maintaining schedule entries with no duration.
- 5) If the *duration* fields are specified and the scheduled action is not repetitive (*repeat\_information\_length* = 0), and the posting device continues to use the resource after the duration is up, the posting device should write the expected duration into the *duration* fields.
- 6) If the *duration* fields are specified and repeat is specified, and the posting device continues to use the resource after the duration is up, the posting device may write the expected duration into the *duration* fields. When the event is complete, the posting device may write the original duration or keep the new duration in the duration field.

### 8.1.7 Repeat information

The *repeat\_information* field indicates when and how the schedule will be repeated.

If the scheduled action is not repetitive, the *repeat\_information\_length* field shall be set to 00<sub>16</sub>.

The *repeat\_information\_length* field contains the number of bytes in this structure excluding the length field itself.

The repeat types are defined in the following table.

**Table 8.1 – repeat\_type value assignment**

values	definition
00 <sub>16</sub>	Weekly schedule
01 <sub>16</sub> – 0F <sub>16</sub>	reserved
10 <sub>16</sub>	Interval schedule
11 <sub>16</sub> – FF <sub>16</sub>	reserved

The *repeat\_information* fields are dependent on the repeat type chosen.

**8.1.7.1 repeat type: Weekly**

If the posting device knows the schedule will be repeated weekly, it can specify the day of the week and number of events to be repeated in the following fields.

	msb							lsb
address_offset	contents							
00 <sub>16</sub>	repeat_type							
01 <sub>16</sub>	number_of_events							
02 <sub>16</sub>	Sunday	Monday	Tuesday	Wed- nesday	Thurs- day	Friday	Saturday	Re- served

**Figure 8.4 – repeat\_information fields for Weekly**

**8.1.7.1.1 repeat type: Weekly descriptions**

**repeat\_type:** The value of the *repeat\_type* field is 00<sub>16</sub> (Weekly).

**number\_of\_events:** The *number\_of\_events* field shall hold the total number of events. For example, if Monday, Tuesday and Thursday were selected, and the *number\_of\_events* is set to 4, and now is the end of Sunday, then Monday, Tuesday, Thursday and the following Monday are scheduled. In the last week, as the *number\_of\_events* is decremented, the weekly flags need not be cleared.

**Weekly flags:** The weekly flags indicate the day of the week when the repetitive event starts.

**8.1.7.1.2 Specifying weekly repeat\_information fields**

The posting device is required to set and update the weekly repeat fields in the following manner.

- 1) The *reserved* flag should be set to 0.
- 2) Setting the *number\_of\_events* field to 00<sub>16</sub> can specify an infinite number of events.
- 3) If an event spans over more than one day, only the flag for the day when the event starts is set. For example, if there is a schedule that starts at 23:00, whose duration is 3 hours, and repeats every Monday and Wednesday, only the flags for Monday and Wednesday are set to 1. Tuesday and Thursday flags are 0.

- 4) The *number\_of\_events* field can be specified such that events start and stop on various days of the week. The following figure shows an example of setting start and stop dates at various dates in the week.

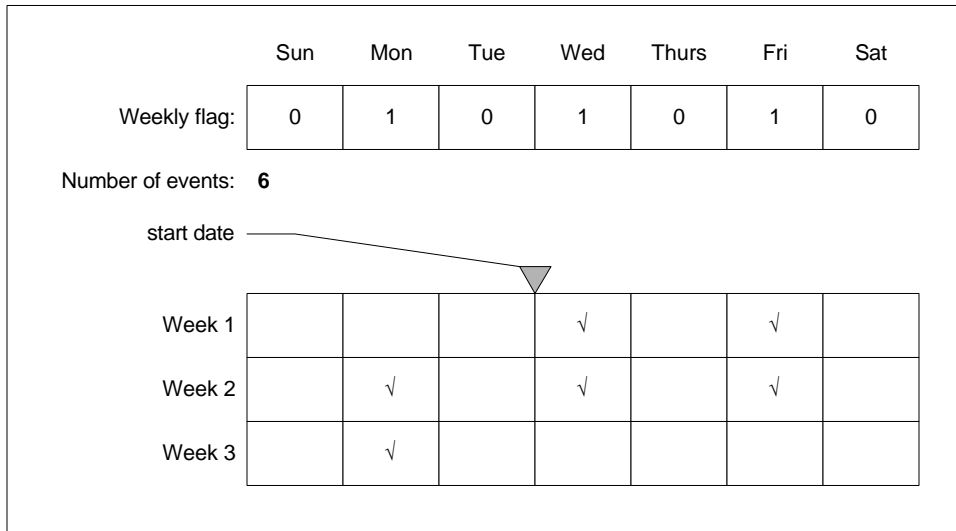


Figure 8.5 – Setting number\_of\_events for a weekly repeat type

- 5) The *start\_time* fields must correspond to a date of a set weekly flag. If the *start\_time* field does not correspond to a weekly flag’s date, the entry is considered invalid.
- 6) A posting device shall update the *number\_of\_events* field according to clause 5.3.4.1 Maintaining repeated schedule entries.

**8.1.7.2 repeat type: Interval**

If the posting device knows the scheduled action will be repeated at regular intervals, it shall specify the interval in the following fields.

address_offset	contents	
00 <sub>16</sub>	repeat_type	
01 <sub>16</sub>	number_of_events	
02 <sub>16</sub>	Reserved (4 bits)	Interval: hours (msb)
03 <sub>16</sub>	Interval: hours (lsb)	
04 <sub>16</sub>	Interval: minutes	
05 <sub>16</sub>	Interval: seconds	

Figure 8.6 – repeat\_information fields for Interval

**8.1.7.2.1 repeat type: Interval descriptions**

**repeat\_type:** The value of *repeat\_type* field is 10<sub>16</sub>.

**number\_of\_events:** The *number\_of\_events* field shall hold the total number of events. Events can flow over to new days.

**Interval:** The interval is the time from the current event’s *start\_time* to the next event’s *start\_time*.



**Interval:hours:** The *hour* field is coded using 12 bits as 3 digits in 4bit BCD (000<sub>16</sub> – 999<sub>16</sub>).

**Interval:minutes:** The *minutes* field is coded using 8 bits as 2 digits in 4 bit BCD (00<sub>16</sub> – 59<sub>16</sub>).

**Interval:seconds:** The *seconds* field is coded using 8 bits as 2 digits in 4 bit BCD (00<sub>16</sub> – 59<sub>16</sub>).

**8.1.7.2.2 Specifying interval repeat\_information fields**

The posting device is required to set and update the interval repeat fields in the following manner.

- 1) Reserved fields shall be set to 00<sub>16</sub>.
- 2) If the posting device’s time resolution is one minute, it shall set the *second* field to the value 00<sub>16</sub>.
- 3) The interval time must be greater than or equal to the duration. An interval time that is less than the duration is considered invalid.
- 4) If the value of duration is zero, the value of interval repeat fields also shall be zeros. This value means the interval repeat fields are not specified.
- 5) Refer to clause 5.3.4.1 “Maintaining repeated schedule entries” for information on how to maintain the *number\_of\_events* field.
- 6) Setting the *number\_of\_events* field to 00<sub>16</sub> can specify an infinite number of events.

**8.1.8 Subunit\_resource\_info\_block (8900<sub>16</sub>)**

*Subunit\_resource\_info\_block* indicates the subunit(s) the posting device will use.

address_offset	contents
00 <sub>16</sub>	compound_length
01 <sub>16</sub>	
02 <sub>16</sub>	info_block_type
03 <sub>16</sub>	
04 <sub>16</sub>	primary_fields_length
05 <sub>16</sub>	
06 <sub>16</sub>	number_of_subunits
07 <sub>16</sub>	subunit_type_and_ID[0]
:	:

**Figure 8.7 – Subunit\_Resource\_info\_block**

**8.1.8.1 Subunit\_resource\_info\_block field descriptions**

**compound\_length:** The *compound\_length* is the length in bytes of the info block, excluding the length field itself.

**info\_block\_type:** The *info\_block\_type* is set to 8900<sub>16</sub>.

**primary\_fields\_length:** The *primary\_fields\_length* is the length in bytes of the *number\_of\_subunits* field and *subunit\_type\_and\_ID* fields.

**number\_of\_subunits:** The *number\_of\_subunits* field contains the number of subunits that the posting device will use.

**subunit\_type\_and\_ID:** The *subunit\_type\_and\_ID* fields specify each of subunits that the posting device will use. The value of *Subunit type and ID* is referred to in the AV/C General specification. If the posting device intends to schedule the whole unit, it shall specify the unit in this field by setting it to FF<sub>16</sub>, and no other subunits shall be specified.

## 9. Board type specific command support level

This section defines the required commands for the Resource Schedule Board Type specification.

The support levels are described as follows:

- “M” means that the command is required.
- “O” means that the command is optional.
- “NS” means that the command is not supported.

The device that has the Resource Schedule Board shall implement all commands that are specified as Mandatory in Table 11.1 of the AV/C Bulletin Board Subunit General Specification version 1.0 [R3], and in the following table.

**Table 9.1 – Board type specific command support level**

opcode/operand		Support level
WRITE DESCRIPTOR (control)		
descriptor_identifier	an object list specified by ID	NS
WRITE DESCRIPTOR (status)		
descriptor_identifier	an object list specified by ID	NS
CREATE DESCRIPTOR (control)		
subfunction_1	create a new object and child list	NS
subfunction_1	create a new descriptor	
	descriptor_identifier	create a child list
		create an object
		NS
		M
READ INFO BLOCK (control)		
descriptor_identifier	subunit identifier	NS
	an object list specified by ID	NS
	an object entry position reference	M
	an object ID reference	M
info_block_identifier	by its type and instance count	M
	by position in the container structure	M
WRITE INFO BLOCK (control)		
subfunction	partial_replace	M
descriptor_identifier	an object list specified by ID	NS
	an object entry position reference	O
	an object ID reference	M
info_block_identifier	by its type and instance count	M
	by position in the container structure	O

## 10. Resource Schedule Board profiles

The implementation profile ID indicates the set of commands and data structures that a device supports. Future versions of this specification may add new data structures, and thus require new mandatory commands for those structures.

The following table illustrates the implementation profile ID values that have been defined for the Resource Schedule Board.

**Table 10.1 – Bulletin Board Subunit Profile Value Assignment**

<b>implementation_profile_ID</b>	<b>meaning</b>
10 <sub>16</sub>	Minimum implementation profile. Only mandatory commands are guaranteed to be supported. Repeat type and info block specified in the supported board version are supported.
All others	Reserved for future versions

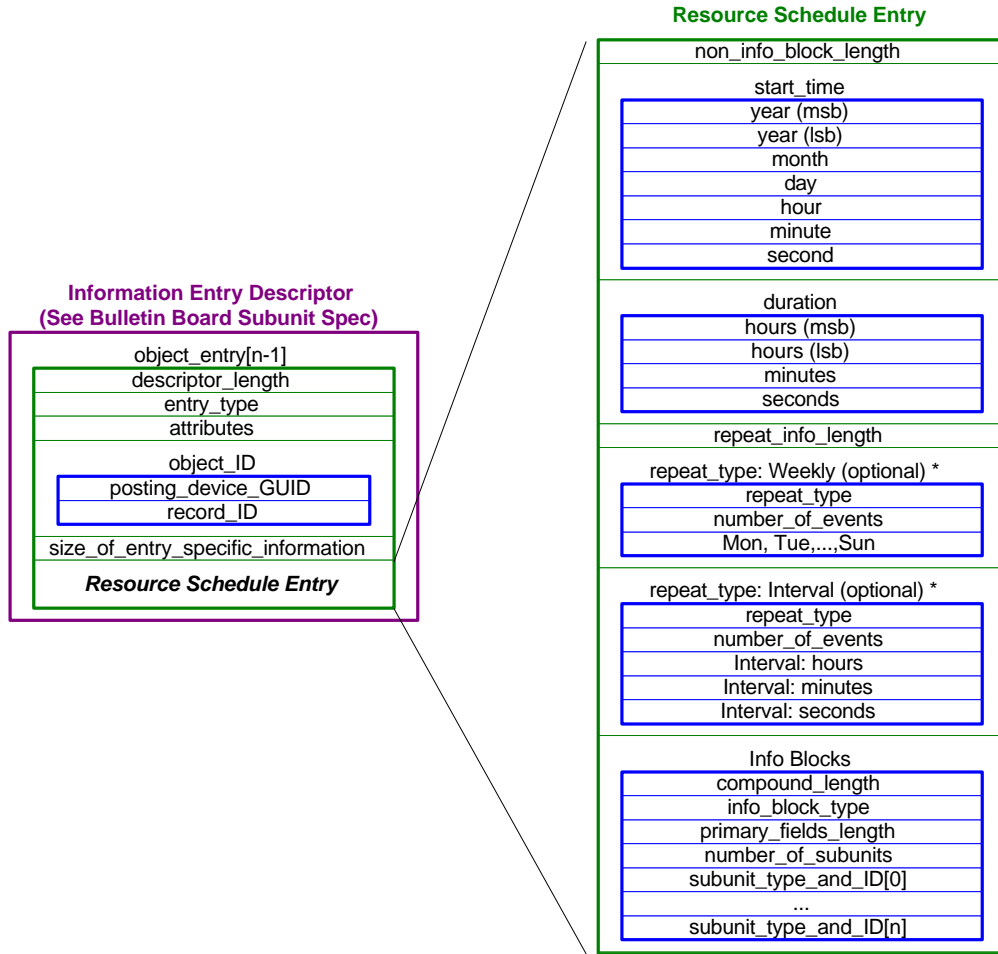
This page left blank intentionally



**Annex A: Resource schedule entry high level view (informative)**

**A.1 Resource schedule entry**

The following figure is the Resource Schedule Board’s resource schedule entry in a high-level view. This view includes the *Information Entry Descriptor* as described in the Bulletin Board Subunit General Specification [R3]. Note that the field sizes are not shown.



**Figure A.1 – Resource schedule entry high level view**

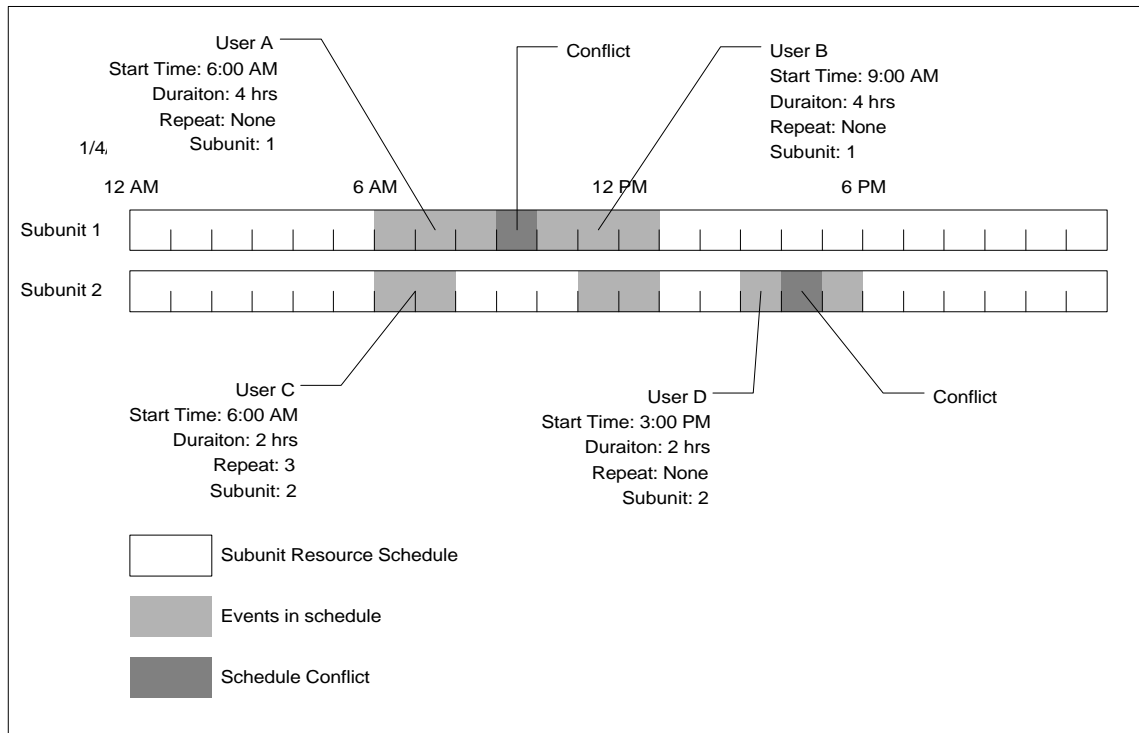
\* Either none or only one repeat type is possible

## Annex B: Scheduling Conflicts

The following information explains how scheduling conflicts could occur within the Resource Schedule Board.

### B.1 Posting device schedules time conflicting with another posting device's entry

In this case, a posting device schedules a time that conflicts with an existing entry. The following figure shows this occurrence:



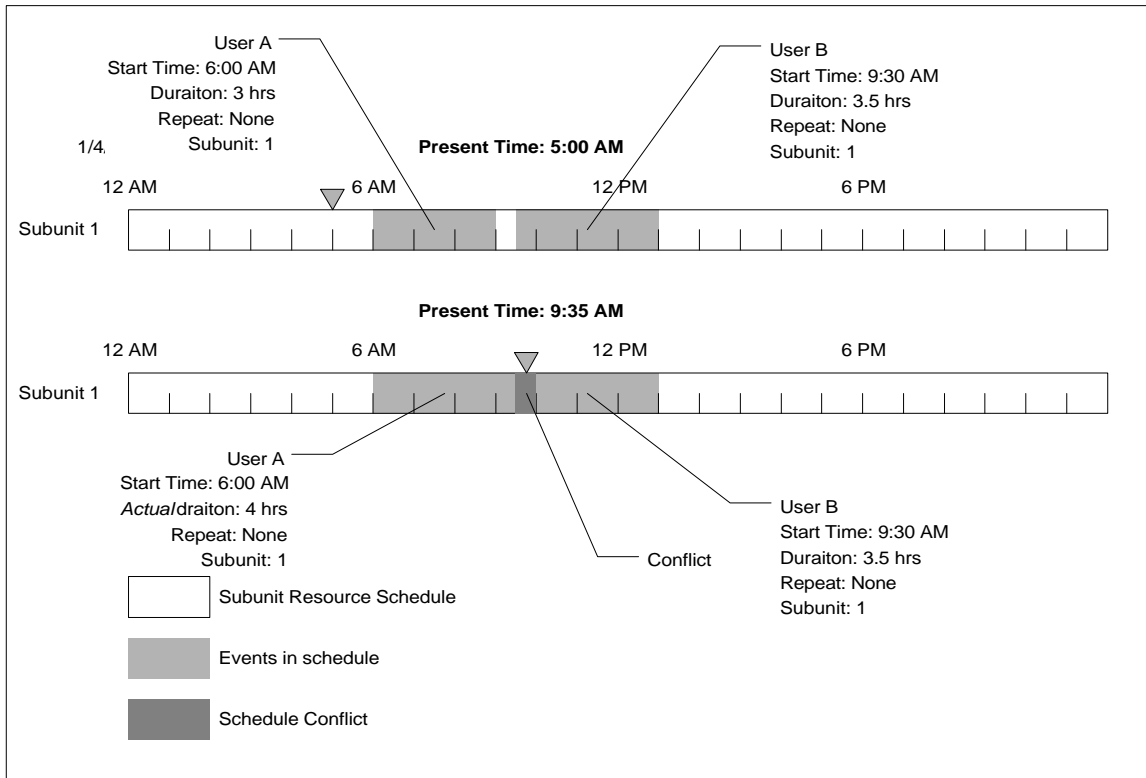
**Figure B.1 – Simple Scheduling Conflict**

Under these circumstances, the device whose start time is first will continue to use the schedule. So, for example, User A will use the schedule until 10:00 AM. User B will then be able to use the schedule at 10:00 AM. User C's first two events will run on time. The third event will run for one hour starting at 5:00 PM, since User D's event runs two hours starting at 3:00 PM.

### B.2 Devices Exceeding Duration

Other conflicts occur when devices exceed their posted use of the resource schedule. This is illustrated in the following figure.



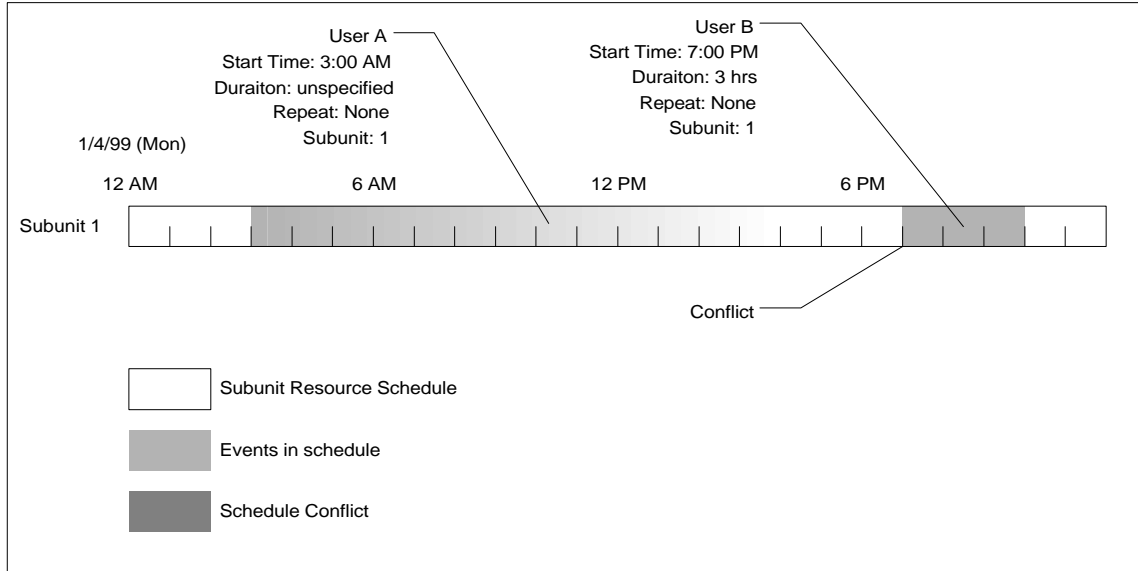


**Figure B.2 – Duration exceeding original duration conflict**

Under these circumstances, the user whose start time is first can continue if the RESERVE control command was used on the resource. When User A gives up control of the resource, it will be available for user B. If user A does not reserve the resource, then user B can use the RESERVE control command to acquire exclusive control of it.

### B.3 Conflicts with events using no duration

When events are scheduled with no duration, new events cannot be safely entered into the schedule without a possible resource schedule conflict. The following figure shows this condition:



**Figure B.3 – Duration unspecified conflict**

With this kind of conflict, the event with the earliest start time can continue to use the resource. The *start\_time* fields are updated once per hour indicating that the subunit is active. User B’s posting device may monitor User A’s event for completion. If the start time is not updated after one hour, User B’s posting device may delete User A’s schedule entry, freeing up the time slot.