RP 172-1993

SMPTE RECOMMENDED PRACTICE

Common Messages for Digital Control Interface

1 Scope

This practice details and defines the control message subset common messages. Common messages are used to perform certain functions common to all equipment types within a general-purpose communications channel of an interface system. This interface system shall transport data and digital control signals between equipment utilized in the production, post-production, and/or transmission of visual and aural information.

2 Notation

This practice describes the coding of keywords and information fields (I/F) in the form as shown below. The coding "NN" represents the assigned keyword or I/F code, in hexadecimal form.

NN	KEYWORD or I/F NAME	Keyword or I/F descriptive text	
	Format:	<COMMAND>	
		<PARAMETER NAME 0>	[Parameter description;
		...	Parameter value coding, scale or range;
		<PARAMETER NAME n>	Parameter definitions and explanations.]

In the practices listed in annex B, keywords are listed numerically, in hexadecimal notation. Keyword numbers are reserved as follows:

Keywords 00_h – $1F_h$: System service subset;
Keywords 20_h – $3F_h$: Common message subset;
Keywords 40_h – FF_h: Virtual machine type-specific subset.

Approved July 1, 1993

3 Summary of keywords, mnemonics and information field (I/F) names

Hex	Keyword	(Mnemonic)	Hex	I/F name	(Mnemonic)
20	CNOP	(CNOP)	20		
21	CRESET	(CRST)	21	VIRTUAL MACHINE TYPE	(VTYP)
22	READ	(READ)	22	EQUIPMENT TYPE	(ETYP)
23	I/F ITEM RESPONSE	(IFRE)	23	TIME STANDARD	(TIME)
24	TIMELINE SOURCE	(TSCE)	24	TIMELINE TIME	(TTIM)
25	STARTUP RESPONSE	(STRE)	25	HIGH-RES TIMELINE TIME	(HTIM)
26	EXECUTE PROCEDURE	(EXPR)	26	EVENT BUFFER STATUS	(EBST)
27	DEFINE EVENT	(DEEV)	27	VIRTUAL MACHINE STATUS	(VMST)
28	CLEAR EVENT	(CLEV)	28		
29	ERROR	(CERR)	29	FAULT STATUS	(FTST)
2A			2A		
2B			2B		
2C			2C		
2D	FAILURE	(FAIL)	2D		
2E			2E		
2F	TIMELINE STOP	(TSTP)	2F		
30	TIMELINE RUN	(TRUN)	30		
3E	USER DEFINED	(UDEF)	3E	USER DEFINED	(UDND)
3F	EXTENSION	(CEXT)	3F	EXTENSION	(CIEX)

NOTE – The ability to perform command 29_h (ERROR) is mandatory for every virtual machine.

EXTENSION SET

Hex	Keyword	(Mnemonic)
00		
01		
02		
03	FUNCTION POLL	(FNPL)
04	FUNCTION RESPONSE	(FNRE)
05	FIELD POLL	(FDPL)
06	FIELD RESPONSE	(FDRE)
07	UPDATE	(UDAT)
08	CYCLE	(CYCL)
09	MUTE	(MUTE)
0A	SIMULTANEOUS READ	(SIRD)
0B	DEFINE PROCEDURE	(DEPR)
0C	DELETE PROCEDURE	(DLPR)
0D	RECALL PROCEDURE	(REPR)
0E	PROCEDURE RESPONSE	(PRRE)
0F	RECALL EVENT	(REEV)
10	EVENT RESPONSE	(EVRE)
11	SIMULTANEOUS READ RESPONSE	(SRDR)
FF	EXTENSION	(EXEX)

4 Keywords

Hex Keyword

20 CNOP — Virtual machine no operation.

Format: <CNOP>

21 CRESET — Directs the destination virtual machine to assume standard values of all preselectable functions. (Same status as power up).

Format: <CRESET>

22 READ — Directs the virtual machine to transmit the instantaneous content of the specified information field.

Format: <READ>
<I/F NAME>

NOTE – Several <I/F NAMES> may be wrapped in a BEGIN/END construct.

23 I/F ITEM RESPONSE — Response to READ, UPDATE, or CYCLE commands.

Format: <I/F ITEM RESPONSE>
<I/F NAME>
<I/F VALUE> (Length varies according to the I/F NAME)

NOTE – Several <I/F NAME> AND <I/F VALUE> pairs may be wrapped in a BEGIN/END construct.

24 TIMELINE SOURCE — Directs the virtual machine to select the source of the timeline.

Format: <TIMELINE SOURCE>
<SOURCE IDENT>

00_h INTERNAL – Internal clock incremented by an unspecified source ("tick")

01_h EXTERNAL – External reference time

25 STARTUP RESPONSE — Indicates that the controlled device has been started up.

Format: <STARTUP RESPONSE>
<MODE>

1-byte special binary number:
00_h = coldstart
01_h = warmstart

NOTES
1 This response is generated automatically whenever the controlled device is powered up.
2 "Coldstart" means that all internal stores are cleared; all functions are set to their default conditions.

26 EXECUTE PROCEDURE

Directs the virtual machine to execute immediately the procedure name.

Format: <EXECUTE PROCEDURE>
<PROCEDURE NAME> — NAME is in the range 01_h to FF_h. 00_h is reserved.

27 DEFINE EVENT

Prepares an event; i.e., a function which shall be executed at the instant of coincidence of a specified trigger time with the content of a specified I/F time.

Format: <DEFINE EVENT>
<EVENT NAME> — 8-bIts
<I/F NAME OF TRIGGER SOURCE>
<TRIGGER VALUE>
<COMMAND> — Specifies the function.

NOTES
1 In order to implement a procedure in an event, EXECUTE PROCEDURE shall be used for the COMMAND, and the procedure shall have been predefined.
2 The TRIGGER SOURCE I/F NAME is a TIMELINE or a type-specific time information field.
3 All functions contained within a procedure defined as an event must be executed by the virtual machine at the trigger time specified by the event.
4 The virtual machine shall clear an event on execution.
5 The EVENT NAME is unique for each event.
6 Where mutually exclusive commands are given inadvertently at the same time through the use of events constructs, such events shall be cleared by the virtual machine and an error message returned.

28 CLEAR EVENT

Clears one or all events previously established.

Format: <CLEAR EVENT>
<EVENT NAME> — ($<00_{h>}$ is all events)

29 ERROR

Advises the controlled virtual machine that the previous string has not been understood by or cannot be performed by the controlled virtual machine.

Format: <ERROR>
<EXEC CODE> — 8-bits:
00 = Parse error.
01 = Cannot do by design.
02 = Insufficiently equipped.
03 = Buffer overflow.
04 = Invalid keyword.
05 = Invalid keyword argument.
FE = See FAULT STATUS information field for more information.
FF = Unspecified
<BYTE COUNT> — 8-bits, not including the byte count.
<OFFENDING STRING> — Truncated not to exceed an overall ERROR message length of 256 bytes.

2D FAILURE

Warns of a catastrophic failure of the specific machine; i.e., a failure which requires intervention by the local operator.

Format: <FAILURE>

2F TIMELINE STOP

If the timeline is internal, stops the timeline from incrementing.

Format: <TIMELINE STOP>

30 TIMELINE RUN

If the timeline is internal, starts the timeline incrementing from the time indicated.

Format: <TIMELINE RUN>
<TIMELINE VALUE> (type TIME)

3E USER DEFINED

Identifies USER DEFINED commands.

Format: <USER DEFINED>
<BYTE COUNT> 16-bit binary unsigned number. Specifies the length of the command, in bytes, not including the byte count itself.
<RAW DATA> (Length varies according to the byte count.)

3F EXTENSION

Directs the virtual machine to enter the common message extension set for the following single command only. The virtual machine shall then resume execution of the basic command set.

Format: <EXTENSION>
<EXTENSION SET COMMAND> (1 or more bytes)

EXTENSION SET

00
01 } RESERVED
02

03 FUNCTION POLL

Directs the virtual machine to indicate which of the keywords contained in the command set are supported by its type-specific machine. BEGIN and END are excluded from the keywords. The existence of the function poll command assumes the existence of the BEGIN/END construct.

Format:
<FUNCTION POLL>
<BEGIN>
<KEYWORD 1>
<KEYWORD 2>
<KEYWORD ...>
<END>

04 FUNCTION RESPONSE

Contains the list of supported keywords in response to a FUNCTION POLL command.

Format:
<FUNCTION RESPONSE>
<BEGIN>
<KEYWORD 1>
<KEYWORD 2>
<KEYWORD ...>
<END>

05 FIELD POLL

Directs the virtual machine to indicate which I/F names contained in the parameter list are supported by the type-specific machine information field.

Format:
<FIELD POLL>
<BEGIN>
<I/F NAME 1>
<I/F NAME 2>
<I/F NAME ...>
<END>

06 FIELD RESPONSE

Contains the list of supported I/F names from those indicated in a FIELD POLL command.

Format:
<FIELD RESPONSE>
<BEGIN>
<I/F NAME 1>
<I/F NAME 2>
<I/F NAME ...>
<END>

07 UPDATE

Directs the virtual machine to respond immediately with the contents of the information field, and then automatically whenever its contents change.

Format: <UPDATE>
<I/F NAME> (Hex)

NOTES
1 The single <I/F NAME> may be replaced by several names wrapped into a BEGIN/END construct.
2 The default condition is MUTEd.
3 When an information field value has changed a number of times in the period between bus-controller polls, only the most recent value is transmitted at the next poll.

08 CYCLE

Directs the virtual machine to transmit periodically, as specified, the instantaneous contents of the specified information field.

Format: <CYCLE>
<TIME INTERVAL> (type TIME)
<I/F NAME> (Hex)

NOTES
1 The single <I/F NAME> may be replaced by several names wrapped in a BEGIN/END construct.
2 The default condition is MUTEd.
3 When an information field value has changed a number of times in the period between bus-controller polls, only the most recent value is transmitted at the next poll.

09 MUTE

Directs the virtual machine to switch off all responses previously initiated by CYCLE or UPDATE commands.

Format: <MUTE>

0A SIMULTANEOUS READ

Directs the virtual machine to read simultaneously the contents of the specified information fields.

Format: <SIMULTANEOUS READ>
<BEGIN>
<I/F NAME>
<I/F NAME>
.
.
<END>

0B DEFINE PROCEDURE

Directs the virtual machine to assemble a block of virtual machine commands for subsequent execution.

Format:

<DEFINE PROCEDURE>	
<PROCEDURE NAME>	(Hex) in the range 01_h – FF_h. 00_h is reserved.
<BYTE COUNT>	16-bits, not including the byte count.
<COMMAND 1> <COMMAND 2> <COMMAND …>	The procedure.

NOTES
1 All functions contained within a procedure which is used within an event must be executed by the virtual machine at the trigger time specified by the event, even if actions must be taken in advance.
2 Procedures are retained until receipt of a DELETE PROCEDURE or CRESET command.

0C DELETE PROCEDURE

Directs the virtual machine to delete a command block previously defined.

Format:

<DELETE PROCEDURE>	
<PROCEDURE NAME>	(Hex) (00_h deletes all procedures)

0D RECALL PROCEDURE

Directs the virtual machine to transmit, but not execute or delete, the specified procedure for checking purposes.

Format:

<RECALL PROCEDURE>	
<PROCEDURE NAME>	(Hex) (00_h recalls all procedures)

0E PROCEDURE RESPONSE

Response to RECALL PROCEDURE command.

Format:

<PROCEDURE RESPONSE>	
<PROCEDURE NAME>	
<BYTE COUNT>	16-bits, not including the byte count.
<COMMAND 1>	
<COMMAND 2>	
<COMMAND …>	

0F RECALL EVENT

Causes an EVENT RESPONSE from the controlled virtual machine containing the data of an event already established.

Format: <RECALL EVENT>
<EVENT NAME> (00_h recalls all events)

10 EVENT RESPONSE

Contains the data of an event already established.

Format: <EVENT RESPONSE>
<EVENT NAME>
<I/F NAME of TRIGGER SOURCE>
<TRIGGER VALUE> (type TIME)
<COMMAND> Function caused by trigger condition.

11 SIMULTANEOUS READ RESPONSE

Response to SIMULTANEOUS READ with all specified information fields.

Format: <SIMULTANEOUS READ RESPONSE>
<BEGIN>
<I/F NAME 1>
<I/F VALUE 1>
<I/F NAME 2>
<I/F VALUE 2>
.
.
.
<END>

FF EXTENSION

Directs the virtual machine to enter the further extension set for the following single command only. The virtual machine shall then resume execution of the basic set.

Format: <EXTENSION>
<EXTENSION SET COMMAND>

5 Information fields

20 RESERVED

21 VIRTUAL MACHINE TYPE — Contains the virtual machine name and hence defines the type-specific machine command set.

Format: <VIRTUAL MACHINE TYPE>
<VIRTUAL MACHINE NAME> (8-bit binary unsigned number)

NOTE – The content of VIRTUAL MACHINE NAME shall be defined explicitly in each virtual machine dialect; the virtual machine name for a wholly USER-DEFINED virtual machine is 01_h.

22 EQUIPMENT TYPE — Contains the data to identify the specific product, including hardware/software revision level.

Format: <EQUIPMENT TYPE>
<BYTE COUNT> 8-bits; not including the byte count itself.
<ISO/IEC 646 printing characters>

NOTE – The ISO/IEC characters shall contain three fields: (1) manufacturer identification, (2) product identification, and (3) revision level (in that order). Each field shall be terminated by OD_h.

23 TIME STANDARD — Contains the nominal field rate to be used, or in use.

Format: <TIME STANDARD>
<NAME> 8-bit binary unsigned number:
00_h is undefined
01_h is “48"
02_h is ”50"
03_h is “60"

24 TIMELINE TIME — Contains the timeline time value.

Format: <TIMELINE TIME>
<TIMELINE TIME VALUE> Standard “time” format.

25 HIGH-RES TIMELINE TIME — Contains the time of a high-resolution timeline where applicable.

Format: <HIGH-RES TIMELINE TIME>
<TIME VALUE> 6-byte expression in “high-resolution time” format.

26 EVENT BUFFER STATUS — Tallies the event buffer status.

Format: <EVENT BUFFER STATUS>
<STATUS REPORT> — Space remaining in bytes. 16-bit number.

27 VIRTUAL MACHINE STATUS — Tallies the virtual machine status.

Format: <VIRTUAL MACHINE STATUS>
<STATUS REPORT> — 00_h = OFF; 01_h = not available; 02_h = available

29 FAULT STATUS — Tallies faults in the system and their reasons where detected by internal diagnostics (as applicable).

Format: <FAULT STATUS>
<PARAMETER GROUP COUNT> — 1-byte number specifying the number n of parameter groups following:

<ERROR CODE>	1-byte code	1st detected
<FAULTY ITEM #>	1-byte number	fault
. . .		
<ERROR CODE>	1-byte code	nth detected
<FAULTY ITEM #>	1-byte number	fault

NOTES
1 Parameter group count = 0 means: no fault.
2 The error codes and faulty item numbers are user-defined.

3E USER DEFINED — Identifies USER DEFINED information fields.

Format: <USER DEFINED>
<BYTE COUNT> — 16-bIt binary unsigned number. Specifies the length of the information field in bytes, not including the byte count itself.
<RAW DATA> — (Length varies according to the byte count.)

3F EXTENSION — Directs the virtual machine to enter the common message I/F name extension set for the following single I/F name only. The virtual machine shall then resume access to the basic I/F name set.

Format: <EXTENSION>

Annex A (informative)
General concepts

The following text contains a general explanation of some of the concepts used in the formulation of the common message set. It constitutes tutorial information and is intended to assist in the understanding of the specifications in previous portions of this practice.

A.1 Commands and responses

The message language is subdivided into two varieties of message which differ only in the direction of information flow between controlling and controlled virtual machines.

– COMMANDS are messages from a controlling to a controlled virtual machine;

– RESPONSES are messages from a controlled to a controlling virtual machine; responses are generally transmitted in reaction to a command.

A.2 State machine and information transfer

The virtual machine controlled by the message language is considered to be a STATE MACHINE. The message set can be regarded as being of two types:

– Messages which change the state of the virtual machine (e.g., the VTR messages STOP and PLAY). These commands reside mainly in the type-speciic message set and comprise commands which are mutually exclusive (e.g., the tape motion commands [TMCs] in the VTR set);

– Messages (commands and responses) which do not change the state, but which only carry information to or from the virtual machine. As information transfer is a general requirement of all types of virtual machines, general principles are applied to these tasks; therefore, these messages reside mainly in the common message set.

A.3 Information fields (I/F)

Items of information which are maintained by and held within a controlled virtual machine, and which may be needed by its controlling virtual machine, are arranged in a virtual array of INFORMATION FIELDS, in a manner similar in concept to a data base.

Each information field is identified by a unique descriptor called the INFORMATION FIELD NAME. This name, coded as a binary value, is used as an address within all commands referencing the field. Therefore, the information field name is used as a parameter name in these commands.

The format of information field data within each message, as transmitted over the remote control system, is predefined for each item by the information field name. Each message set requires its own array of information fields. The complete field array of a specific virtual machine comprises the field array specified in the common message set, together with that of the type-specific message set.

A typical example of a command requiring an information field is READ, which directs the virtual machine to transmit the content of one or more information fields, as specified within the command. The SIMULTANEOUS READ command directs the virtual machine to read simultaneously the instantaneous values of a number of specified information fields. In response to this command, all specified fields will be read as a “snapshot” and will, therefore, be “frozen” during the read period.

It is essential to be able to PRESET the values of certain items held within information fields. However, since the preset function could indirectly change the state of the virtual machine (e.g., presetting a tape-timer), the PRESET command is contained within the type-specific command set; information fields to which it relates are then individually specified.

A.4 Error and failure messages

An ERROR message advises a controlling virtual machine that the command as identified cannot be performed. The reason for the inability to perform the action is contained within an EXEC CODE transmitted as a parameter to the ERROR keyword. The string which caused the error message is then appended to the EXEC CODE preceded by a byte count.

A special EXEC CODE is available which directs that an information field (FAULT STATUS) should be read to obtain more information on the error. In the event of failure of the specific machine (i.e., a failure requiring the attendance of an operator), a single byte FAILURE message is transmitted.

A.5 Inquiry concept

Although ideally every virtual machine should respond to the complete message set, it is the responsibility of each manufacturer to determine the degree of conformance of his product. To enable a controlling virtual machine to determine the facilities supported by a remote-controlled virtual machine, two inquiry commands are provided:

– FUNCTION POLL to identify supported commands;
– FIELD POLL to identify supported information fields.

The associated responses are FUNCTION POLL RESPONSE and FIELD POLL RESPONSE. Virtual machines that do not support these inquiry commands must respond to any unknown command with ERROR.

A.6 Standard and extension keywords

Because of the limited code space available, each message set (system service, common, type-specific) contains an extension keyword which opens an additional code space of 256 additional keywords.

Frequently used keywords will preferably reside in the standard set. For keywords that are used less frequently, the additional overhead of one byte is acceptable; such keywords have been put in the extension set from the beginning, thus making room in the standard set for future applications.

A.7 Procedures

A group of commands which are to be executed in sequence on one or more occasions may be combined into a procedure using the command DEFINE PROCEDURE. Once defined, a procedure can be called simply by the command EXECUTE PROCEDURE as often as wanted until cancelled by the DELETE PROCEDURE command. It is possible to define more than one procedure at a time using different procedure names coded as binary numbers.

The command RECALL PROCEDURE and the associated response PROCEDURE RESPONSE may be used to inspect currently-defined procedures.

A.8 Timeline concept

In order to allow for synchronous processes in and among several virtual machines, a timescale common to all virtual machines is provided which may be referenced by certain commands. This timescale is called the TIMELINE. The default resolution of the timeline shall be frames. As an option, a higher resolution may be specified using the ESbus high-resolution time format.

The timeline may be derived externally by a locally defined reference time (e.g., derived from a central time code generator and distributed over separate lines), or it may be generated internally by a built-in clock, the "machine internal clock," that gets only its "ticks" from an external source available to all machine internal clocks of the system (e.g., the vertical pulse in television applications). Either one of the two possibilities may be selected by the TIMELINE SOURCE command.

When the machine internal clock is selected as the timeline source, the timeline may be stopped by the TIMELINE STOP command and restarted by the TIMELINE RUN command which also specifies the start value. The TIMELINE RUN command is also issued by the bus controller in response to the system service command REQUEST TIME TRANSMISSION; this allows for exact synchronization of all timelines of the system. The current status of the timeline may be accessed through the information fields TIMELINE TIME or HI-RES TIMELINE TIME.

A.9 Events

An event specifies a command that will be executed on occurrence of a specified trigger condition. The trigger condition arises when a specified trigger value coincides with the content of a specified trigger source. Any information field of the specific virtual machine may serve as a trigger source. However, the most important trigger source is the timeline. This allows for time-synchronous events in different virtual machines; e.g., synchronizing the transports of several VTRs.

The command DEFINE EVENT is used to specify an event. The event is cleared by the occurrence of the trigger condition or by a CLEAR EVENT command.

The command RECALL EVENT and the associated response EVENT RESPONSE may be used to inspect pending events.

Using EXECUTE PROCEDURE as the command within an event specification allows for a sequence of commands to be programmed for execution on a trigger condition. The accuracy with which an event is defined and executed shall depend on the type of timeline in use (default or high-resolution).

It is important to note that the controlled virtual machine, once programmed with an event, is responsible for taking care of all necessary actions for the correct execution of that event. This is the case even if actions have to be taken in advance of the occurrence of the trigger condition. For example, when executing an event that programs an edit entry on the timeline, the virtual machine (in this instance, a VTR) must apply the necessary switch commands for the erase head a certain number of frames in advance of the trigger time.

A.10 Tasks with repeated responses

In order to reduce overhead on the remote control system, commands are provided which may be used to instruct a controlled virtual machine to transfer the content of an information field repeatedly, either whenever the content changes (UPDATE command) or when a specified time period is over (CYCLE command).

However, caution must be exercised in the use of multiple UPDATE commands where the values of the specified information fields are changing rapidly; bus congestion may occur. Additionally, when an information field value has changed a number of times in the period between bus controller polls, only the most recent value is transmitted at the next poll in response to either the UPDATE or CYCLE commands. This will minimize the risk of bus congestion.

NOTE – Repeated transmissions must be consistent with the requirements of the supervisory protocol (i.e., a transmission can take place only following a tributary poll and a subsequent service request to the bus contoller).

A.11 Dialect identification

The information field VIRTUAL MACHINE TYPE of the common message set contains a code which defines the type of the type-specific message set. Every dialect has an associated code which is defined in the specification of that message set. The type of virtual machine and the dialect understood by it can be interrogated by READing this information field.

Annex B (informative)
Bibliography

ANSI/SMPTE 207M-1992, Television — Digital Control Interface — Electrical and Mechanical Characteristics

SMPTE RP 113-1992, Supervisory Protocol for Digital Control Interface

SMPTE RP 138-1992, Control Message Architecture

SMPTE RP 139-1992, Tributary Interconnection

SMPTE RP 163-1992, Television — System Service Messages

SMPTE RP 170-1993, Video Tape Recorder Type-Specific Messages for Digital Control Interface

SMPTE RP 171-1993, Type-Specific Messages for Digital Control Interface of Analog Audio Tape Recorders

ISO/IEC 646:1991, Information Technology — ISO 7-Bit Coded Character Set for Information Interchange

ISO 2022:1986, Information Processing — ISO 7-Bit and 8-Bit Coded Character Sets — Code Extension Techniques