RP 191-1996

SMPTE RECOMMENDED PRACTICE

Routing Switcher Type-Specific Messages for Remote Control of Broadcast Equipment

Introduction

This clause contains a general explanation of some of the concepts used in the formulation of the routing switcher type-specific message set. It constitutes tutorial information and is intended to assist in the understanding of the specifications in clause 6 of this practice. A working knowledge of the following ESbus topics is assumed:

- ESbus system overview;
- control message architecture;
- supervisory protocol;
- tributary interconnection;
- electrical and mechanical characteristics;
- system service and common messages.

Conventions: Acronyms and abbreviations are shown in uppercase characters; e.g., information field – I/F.

Message keywords and names of information fields are shown in uppercase characters; e.g., CONNECT CROSSPOINT; SOURCE POINTER.

These command keywords and information field names are used within the text of this practice to imply requested action, information field identity, and, in turn, the information field contents of the virtual machine. To assist in readability of this practice, these terms are used in the context of the presentation material; e.g., “If this I/F is PRESET, ...” (PRESET in this context refers to a command contained within the type-specific message set.)

Terms with special meaning to this or related documents are shown with leading uppercase characters; e.g., Virtual Machine; Common Messages.

1 Scope of the dialect

This dialect is intended to be used for the external control of routing switcher systems.

2 Multidimensional information fields

The controlled elements in a routing switcher are in the crosspoints. The crosspoints are arranged in a multidimensional way; i.e., to identify an individual crosspoint, it is usually necessary to specify the following characteristics of its location:

- its row (1.dimension);
- its column (2.dimension);
- its level (3.dimension);
- its matrix (4.dimension).

(The latter applies only to a switcher consisting of several matrices.)

In ESbus dialects, all kinds of status data are maintained in Information Fields (I/Fs). Each type of information has its corresponding I/F associated with a unique I/F Name. In routing switcher systems, nearly all status data are related to crosspoints. Because crosspoints are arranged in a multidimensional way, and each crosspoint is the carrier of status data, the I/Fs describing routing switcher data must also be multidimensional. The particular item of information belonging to one crosspoint is just an element of the whole Information Field of a certain type. This requires additional descriptors which point to the location of this element within the field; i.e., to the row, column, level, and matrix.

When such an element of an I/F is tallied, these descriptors are simply carried as parameters in the format. When such an element is accessed, however,

Approved September 1, 1996

a different mechanism is required due to the fact that the Common commands which access I/Fs (e.g., READ, UPDATE, CYCLE) allow only for specifying the I/F name. No additional descriptor information is permitted in the format. Such additional information must, therefore, be transmitted in advance by presetting one or more pointers which predefine the parameters necessary for multidimensional access.

The pointers themselves also reside in Information Fields of their own and thus can be PRESET as any other presettable I/F. This gives the advantage that the pointer information need only be transmitted when it really changes, not in advance of every I/F access.

The names of the relevant pointers are:

- MATRIX POINTER;
- LEVEL POINTER;
- SOURCE POINTER;
- DESTINATION POINTER.

NOTE – In order to PRESET a multidimensional I/F, no pointers are required since the whole format of an I/F appears within the PRESET command, including the descriptors

Because of the special requirements of routing switchers, one of these dimensions may not be specified; e.g., the I/F DESTINATIONS-TO-SOURCE does not use the DESTINATION POINTER because it shows all connections between a specified source and any destination in the form of a list.

3 Wildcard characters

In order to facilitate access to a whole array (row, column, level, matrix), of one type of a multidimensional status information, a wildcard character is introduced (FF_h to $FFFF_h$, respectively).

A pointer, preset to the wildcard, indicates (when an I/F access requiring this pointer is made) that the information corresponding to a full available range of the pointer is desired. For example, if the LEVEL POINTER is preset to FF_h, a subsequent READ of the I/F CROSSPOINT STATUS will result in a multiple I/F RESPONSE message (either in many single messages or in one message using the BEGIN/END construct or in combination) tallying the crosspoint status of all existing levels.

A second application of the wildcard concerns certain commands (e.g., CONNECT CROSSPOINT), where it can be used as a normal parameter with the same effect.

The description of the commands and information fields in clause 6 contains detailed instructions as to when and how wildcards can be used.

4 Procedures and events

As with all Dialects, the complete Routing Switcher Dialect, by definition, consists of both Common Messages and the Routing Switcher Type-Specific Dialect described in this practice.

The elements of the Common Messages prove very useful for switcher applications and are, therefore, recommended. The concept of Procedures provided by the common messages can be used to predefine a lengthy set of commands (by the DEFINE PROCEDURE command) pending the arrival of the EXECUTIVE PROCEDURE command, which causes the entire Procedure to be performed.

The concept of Events provided by the Common Messages can be used to predefine a command (by the DEFINE EVENT command) to be executed at a certain point in time. As the time scale (usually the Timeline) is used, an individual software clock running in each controlled Virtual Machine is preset by the bus controller (by the System Service command REQUEST TIME TRANSMISSION).

Both concepts can easily be combed by first defining a procedure, then defining an event with EXECUTE PROCEDURE as the command to be carried out on the timeline.

Details about these facilities are described in the documents about System Service and Common Messages mentioned above.

5 Sample command sequences

The following message sequences show the application of the dialect (including the Common Messages). The commands are also shown encoded into their hexadecimal codes.

The PRESET commands signed with a "*" sign may be omitted if the corresponding I/F has already been preset by a preceding PRESET command.

5.1 Disconnecting all crosspoints in level #10 matrix #2:

```
<DISCONNECT CROSSPOINT>                                  <43>
    <MATRIX = 2>                                         <02>
    <LEVEL = 10>                                         <0A>
    <SOURCE = wildcard>                                  <FFFF>
    <DESTINATION = wildcard>                             <FFFF>
```

5.2 Requesting the status of all crosspoints in level #2 of matrix #1:

```
<PRESET>                                          *      <60>
    <MATRIX POINTER>                                     <41>
        <MATRIX = 1>                                     <01>
<PRESET>                                          *      <60>
    <LEVEL POINTER>                                      <42>
        <LEVEL = 2>                                      <02>
<PRESET>                                          *      <60>
    <SOURCE POINTER>                                     <43>
        <SOURCE = wildcard>                              <FFFF>
<READ>                                                   <22>
    <DESTINATIONS-TO-SOURCE>                             <46>
```

The response from the controlled virtual machine may be:

```
<I/F ITEM RESPONSE>                                      <23>
    <BEGIN>                                              <01>
        <DESTINATION-TO-SOURCE>                          <46>
            <MATRIX = 1>                                 <01>
            <LEVEL = 2>                                  <02>
            <SOURCE = 0>                                 <0000>
                <PARAMETER COUNT = 1>  (one connection)  <0001>
                <DESTINATION = 3>                        <0003>
        <DESTINATION-TO-SOURCE>                          <46>
            <MATRIX = 1>                                 <01>
            <LEVEL = 2>                                  <02>
            <SOURCE = 1>                                 <0001>
                <PARAMETER COUNT = 0>  (no connection)   <0000>
        <DESTINATION-TO-SOURCE>                          <46>
            <MATRIX = 1>                                 <01>
            <LEVEL = 2>                                  <02>
            <SOURCE = 2>                                 <0002>
                <PARAMETER COUNT = 4>  (multiple connection) <0004>
                <DESTINATION = 2>                        <0002>
                <DESTINATION = 4>                        <0004>
                <DESTINATION = 9>                        <0009>
                <DESTINATION = 17>                       <0011>
        <DESTINATION-TO-SOURCE>                          <46>
            <MATRIX = 1>                                 <01>
            <LEVEL = 2>                                  <02>
            <SOURCE = 31>                                <001F>
                <PARAMETER COUNT = 1>                    <0001>
                <DESTINATION = 1>                        <0001>
                <DESTINATION = 4>                        <0004>
                <DESTINATION = 9>                        <0009>
                <DESTINATION = 17>                       <0011>
```

5.3 Instructing the controlled Virtual Machine to tally any changes in the crosspoint status of matrix #3:

```
<PRESET>                                          *      <60>
    <MATRIX POINTER>                                     <41>
        <MATRIX = 3>                                     <03>
```

<PRESET>	*	<60>
<LEVEL POINTER>		<42>
<LEVEL = wildcard>		<FF>
<PRESET>	*	<60>
<DESTINATION POINTER>		<44>
<DESTINATION = wildcard>		<FFFF>
<UPDATE>		<3F07>
<SOURCES-TO-DESTINATION>		<47>

On a change, the response from the controlled Virtual Machine may be:

<I/F ITEM RESPONSE>	<23>
<SOURCES-TO-DESTINATION>	<47>
<MATRIX = 3>	<03>
<LEVEL = 2>	<02>
<DESTINATION = 9>	<0009)
<PARAMETER COUNT = 1>	<0001>
<SOURCE = 11>	<000B>

5.4 Defining a procedure that establishes a default configuration of connections:

<DEFINE PROCEDURE>	<3F0B>
<NAME = 1>	<01>
<BYTE COUNT = XX>	<00XX>
<CONNECT CROSSPOINT>	<42>
<MATRIX = 1>	<01>
<LEVEL = 1>	<01>
<SOURCE = 0>	<0000>
<DESTINATION = 12>	<000C>
<CONNECT CROSSPOINT>	<42>
<MATRIX = 1>	<01>
<LEVEL = 1>	<01>
<SOURCE =1>	<0001>
<DESTINATION = 10>	<000A>
<CONNECT CROSSPOINT>	<42>
<MATRIX = 1>	<01>
<LEVEL = 1>	<01>
<SOURCE = 2>	<0002>
<DESTINATION = 2>	<0002>
<CONNECT CROSSPOINT>	<42>
<MATRIX = 1>	<01>
<LEVEL = 1>	<01>
<SOURCE = 3>	<0003>
<DESTINATION =7>	<0007>
. . .	

Once defined, the procedures may be carried out as often as desirable simply by commanding:

<EXECUTE PROCEDURE>	<26>
<NAME = 1>	<01>

5.5 Setting a crosspoint at 9:00 o'clock on the timeline:

<DEFINE EVENT>	<27>
<NAME = 3>	<03>
<I/F NAME OF TRIGGER SOURCE = TIMELINE>	<24>
<TRIGGER VALUE = 09:00:00:00>	<09000000>
<CONNECT CROSSPOINT>	<42>
<MATRIX = 1>	<01>
<LEVEL = 1>	<01>
<SOURCE = 3>	<0003>
<DESTINATION = 7>	<0007>

6 Routing switcher type-specific messages

Virtual machine type is 05_h.

GENERAL NOTES

1 All parameters described below as 1-byte numbers or 2-byte numbers are binary-coded unsigned numbers.

2 Parameters which can be used with Wildcard Characters are indicated below by FF_h = all or $FFFF_h$ = all, respectively.

3 In all cases, the temporal order or EVENTS must be preserved. Mutually exclusive commands actuated by the EVENT construct, which are placed on the EVENT cue at the same trigger point, will cause both events to cancel.

6.1 Index

Numerical Index of Keywords, Information Field Names, and Mnemonics

	MESSAGE KEYWORDS	MNEMONIC		I/F FIELD NAMES	MNEMONIC
40	Not used		40	COMMAND ERROR STATUS	CEST
41	Not used		41	MATRIX POINTER	MPOI
42	CONNECT CROSSPOINT	CONC	42	LEVEL POINTER	LPOI
43	DISCONNECT CROSSPOINT	DISC	43	SOURCE POINTER	SPOI
44	LOCK CROSSPOINT	LCKC	44	DESTINATION POINTER	DPOI
45	UNLOCK CROSSPOINT	UNLC	45	Not used	
46	SPECIFIC MUTE	SPMT	46	DESTINATION-TO-SOURCE	DTOS
47	Not used		47	SOURCES-TO-DESTINATION	STOD
48	TEST CROSSPOINTS	TESC	48	CROSSPOINT STATUS	CSTA
49	Not used		49	SRC SIGNAL STATUS	SSTA
4A	SET CROSSPOINT EXCLUSION	SCEX	4A	LEVEL CONFIGURATION	LECO
4B	CLEAR CROSSPOINT EXCLUSION	CCEX	4B	LEVEL STATUS-MISSING BLOCKS	LSMB
4C	Not used		4C	LEVEL STATUS-ADDED BLOCKS	LSAB
4D	Not used		4D	Not used	
4E	Not used		4E	SOURCE NAME	SNAM
4F	Not used		4F	DESTINATION NAMES	DNAM
50	Not used		50	LOCKED DEST-TO-SOURCE	LDTS
51	Not used		51	LOCKED SOURCES-TO-DEST	LSTD
52	Not used		52	EXCLUSION DEST-TO-SOURCE	EXDS
53	Not used		53	EXCLUSION SOURCES-TO-DEST	EXSD
54	Not used		54	Not used	
55	Not used		55	Not used	
56	Not used		56	Not used	
57	Not used		57	Not used	
58	CHOP	CHOP	58	DESTINATIONS-IN-CHOP	DICH
59	Not used		59	CHOP SOURCE	CSRC
5A	SET CHOP RATE	SCHR	5A	CHOP RATE STATUS	CHRS
5B	SET CHOP TIME	SCHT	5B	CHOP TIME STATUS	CHTS
5C	Not used		5C	Not used	
5D	Not used		5D	Not used	
5E	Not used		5E	Not used	
5F	Not used		5F	Not used	
60	PRESET	PRST	60	Not used	

6.2 Keywords

40 Not used

41 Not used

42 CONNECT CROSSPOINT causes the crosspoint between the specified source (row) and the specified destination (column) in the specified level of the specified matrix to be connected.

Format:	<CONNECT CROSSPOINT>		
	<MATRIX>	1-byte number	(FF_h = all)
	<LEVEL>	1-byte number	(FF_h = all)
	<SOURCE>	2-byte number	
	<DESTINATION>	2-byte number	($FFFF_h$ = all)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the crosspoints between specified source and destination in the specified level of all available matrices are connected.

2 If the level is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in all available levels of the specified matrix are connected.

3 If the destination is addressed with the wildcard $FFFF_h$, the crosspoints between the specified source and all available destinations in the specified level of the specified matrix are connected.

4 More than one parameter may use the wildcard at the same time.

43 DISCONNECT CROSSPOINT causes the crosspoint between the specified source (row) and the specified destination (column) in the specified level of the specified matrix to be disconnected.

Format:	<DISCONNECT CROSSPOINT>		
	<MATRIX>	1-byte number	(FF_h = all)
	<LEVEL>	1-byte number	(FF_h = all)
	<SOURCE>	2-byte number	($FFFF_h$ = all)
	<DESTINATION>	2-byte number	($FFFF_h$ = all)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the crosspoints between specified source and destination in the specified level of all available matrices are disconnected.

2 If the level is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in all available levels of the specified matrix are disconnected.

3 If the source is addressed with the wildcard $FFFF_h$, a crosspoint that might be set between a source and the specified destination in the specified level of the specified matrix is disconnected.

4 If the destination is addressed with the wildcard $FFFF_h$, the crosspoints between the specified source and all available destinations in the specified level of the specified matrix are disconnected.

5 More than one parameter may use the wildcard at the same time; e.g., if both source and destination are addressed with the wildcard >$FFFF_h$, all crosspoints in the specified level of the specified matrix are disconnected.

44 LOCK CROSSPOINT causes the crosspoint between the specified source (row) and the specified destination (column) in the specified level of the specified matrix to be locked in its current state (either connected or disconnected).

Format:	<LOCK CROSSPOINT>		
	<MATRIX>	1-byte number	(FF_h = all)
	<LEVEL>	1-byte number	(FF_h = all)
	<SOURCE>	2-byte number	($FFFF_h$ = present state)
	<DESTINATION>	2-byte number	($FFFF_h$ = all)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the crosspoints between specified source and destination in the specified level of all available matrices are locked in their respective states.

2 If the level is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in all available levels of the specified matrix are locked in their respective states.

3 If the source is addressed with the wildcard $FFFF_h$, the crosspoints between all available sources and the specified destination are locked in their respective states.

4 If the destination is addressed with the wildcard $FFFF_h$, the crosspoints between the specified source and all available destinations in the specified level of the specified matrix are locked in their respective states.

5 More than one parameter may use the wildcard at the same time.

45 UNLOCK CROSSPOINT causes the crosspoint between the specified source (row) and the specified destination (column) in the specified level of the specified matrix to be unlocked and available for a change in state.

Format:	<UNLOCK CROSSPOINT>		
	<MATRIX>	1-byte number	(FF_h = all)
	<LEVEL>	1-byte number	(FF_h = all)
	<SOURCE>	2-byte number	($FFFF_h$ = all)
	<DESTINATION>	2-byte number	($FFFF_h$ = all)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the crosspoints between specified source and destination in the specified level of all available matrices are unlocked.

2 If the level is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in all available levels of the specified matrix are unlocked.

3 If the source is addressed with the wildcard $FFFF_h$, a crosspoint that might be locked between a source and the specified destination in the specified level of the specified matrix is unlocked.

4 If the destination is addressed with the wildcard $FFFF_h$, the crosspoints between the specified source and all available destinations in the specified level of the specified matrix are unlocked.

5 More than one parameter may use the wildcard at the same time; e.g., if both source and destination are addressed with the wildcard >$FFFF_h$, all crosspoints in the specified level of the specified matrix are unlocked.

46 SPECIFIC MUTE directs the controlled Virtual Machine to switch off all responses previously initiated by a CYCLE or UPDATE command for the specified Information Field.

Format: <SPECIFIC MUTE>
<I/F NAME>

NOTES

1 This command supplements the MUTE command of the Common Message set, which is a general mute for all I/Fs.

2 This command requires the same predefinitions of the pointers as the UPDATE or CYCLE command it is intended to cancel.

47 Not used

48 TEST CROSSPOINT causes the crosspoint between the specified source (row) and the specified destination (column) in the specified level of the specified matrix to be tested. The test result may be interrogated by READing the Information Field CROSSPOINT STATUS.

Format: <TEST CROSSPOINT>

<MATRIX>	1-byte number	(FF_h = all)
<LEVEL>	1-byte number	(FF_h = all)
<SOURCE>	2-byte number	($FFFF_h$ = all)
<DESTINATION>	2-byte number	($FFFF_h$ = all)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the crosspoints between specified source and destination in the specified level of all available matrices are tested.

2 If the level is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in all levels of the specified matrix are tested.

3 If the source is addressed with the wildcard $FFFF_h$, the crosspoints between all sources and the specified destination in the specified level of the specified matrix are tested.

4 If the destination is addressed with the wildcard $FFFF_h$, the crosspoints between the specified source and all destinations in the specified level of the specified matrix are tested.

5 More than one parameter may use the wildcard at the same time; e.g., if both source and destination are addressed with the wildcard >$FFFF_h$, all crosspoints in the specified level of the specified matrix are tested.

49 Not used

4A SET CROSSPOINT EXCLUSION causes connection of the crosspoint between the specified source (row) and the specified destination (column) in the specified level of the specified matrix to be disallowed. Previously connected crosspoints that become excluded are disconnected after execution.

Format: <SET CROSSPOINT EXCLUSION>

<MATRIX>	1-byte number	(FF_h = all)
<LEVEL>	1-byte number	(FF_h = all)
<SOURCE>	2-byte number	($FFFF_h$ = all)
<DESTINATION>	2-byte number	($FFFF_h$ = all

NOTES

1 If the matrix is addressed with the wildcard FF_h, the crosspoints between specified source and destination in the specified level of all available matrices cannot be connected.

2 If the level is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in all available levels of the specified matrix cannot be connected.

3 If the source is addressed with the wildcard $FFFF_h$, the crosspoints between the specified destination and all sources in the specified level of the specified matrix cannot be connected.

4 If the destination is addressed with the wildcard $FFFF_h$, the crosspoints between the specified source and all available destinations in the specified level of the specified matrix are connected.

5 More than one parameter may use the wildcard at the same time.

4B CLEAR CROSSPOINT EXCLUSION causes connection of the crosspoint between the specified source (row) and the specified destination (column) in the specified level of the specified matrix to be allowed. There is no effect on previously connected crosspoints.

Format:	<CLEAR CROSSPOINT EXCLUSION>		
	<MATRIX>	1-byte number	(FF_h = all)
	<LEVEL>	1-byte number	(FF_h = all)
	<SOURCE>	2-byte number	($FFFF_h$ = all)
	<DESTINATION>	2-byte number	($FFFF_h$ = all)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in the specified level of all available matrices can be connected.

2 If the level is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in all available levels of the specified matrix can be connected.

3 If the source is addressed with the wildcard $FFFF_h$, the crosspoints between the specified destination and all sources in the specified level of the specified matrix can be connected.

4 If the destination is addressed with the wildcard $FFFF_h$, the crosspoints between the specified source and all available destinations in the specified level of the specified matrix can be connected.

5 More than one parameter may use the wildcard at the same time.

58 CHOP causes the switcher to switch rapidly between the present source (row) selection and the new source (row) selection for the specified destination (column) in the specified level of the specified matrix. The switching will take place for the default time determined by the switcher system, unless modified by a SET CHOP TIME command. The switching will occur at the default rate for the switcher system, unless modified by a SET CHOP RATE command. At the end of the chop time, the selected output will return to the state present before the chop command started operation. A chop command can be terminated by a CONNECT CROSSPOINT COMMAND or a DISCONNECT CROSSPOINT COMMAND, which will end the chop and perform the new command for the selected crosspoints.

Format:	<CHOP>		
	<MATRIX>	1-byte number	(FF_h = all)
	<LEVEL>	1-byte number	(FF_h = all)
	<SOURCE>	2-byte number	
	<DESTINATION>	2-byte number	($FFFF_h$ = all)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in the specified level of all available matrices will enter the chop mode.

2 If the level is addressed with the wildcard FF_h, the crosspoints between the specified source and destination in all available levels of the specified matrix will enter the chop mode.

3 If the destination is addressed with the wildcard $FFFF_h$, the crosspoints between the specified source and all available destinations in the specified level of the specified matrix will enter the chop mode.

4 More than one parameter may use the wildcard at the same time.

5A SET CHOP RATE sets the chop rate or speed to a new value for the specified level of the specified matrix.

Format:	<SET CHOP RATE>		
	<MATRIX>	1-byte number	(FF_h = all)
	<LEVEL>	1-byte number	(FF_h = all)
	<CHOP RATE>	4-byte time value	(0 = no change)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the chop rate for the specified level within all matrices will be set.

2 If the level is addressed with the wildcard FF_h, the chop rate for all levels within the specified matrix will be set to the new value.

3 The chop rate value is in the standard four-byte TIME format. If the chop rate value is set to zero, the chop duration will not be changed. Any other value will indicate the number of hours, minutes, seconds, and frames of time chop function will take to complete a cycle. This assumes both sources are switched for equal amounts of time. Thus a one-frame rate will have each source present for one field at a time in an interlaced scan system. A two-frame rate will have each source present for one frame at a time.

4 More than one parameter may use the wildcard at the same time.

5B SET CHOP TIME sets the chop duration time to a new value for the specified level of the specified matrix.

Format:	<SET CHOP TIME>		
	<MATRIX>	1-byte number	(FF_h = all)
	<LEVEL>	1-byte number	(FF_h = all)
	<CHOP DURATION>	4-byte time value	(0 = no change)

NOTES

1 If the matrix is addressed with the wildcard FF_h, the chop time for the specified level within all matrices will be set.

2 If the level is addressed with the wildcard FF_h, the chop time for all levels within the specified matrix will be set to the new value.

3 The chop duration value is in the standard four-byte TIME format. If the chop duration value is set to zero, the chop duration will not be changed. Any other time value will indicate the number of hours, minutes, seconds, and frames of time during which the chop function will continue.

4 More than one parameter may use the wildcard at the same time.

. . .

60 PRESET presents the named Information Field to the given value.

Format: <PRESET>
<PERMITTED INFORMATION FIELD NAME>
<VALUE> format and coding defined by the I/F NAME (see 6.3, Information Fields)

Permitted Information Field names for Routing Switchers are:

MATRIX POINTER
LEVEL POINTER
SOURCE POINTER
DESTINATION POINTER
SOURCE NAME
DESTINATION NAME

6.3 Information fields

NOTES

1 The items of the Information Field are accessed by the Common Messages:

READ, UPDATE, CYCLE, or SIMULTANEOUS READ

These commands use the format:

<KEYWORD><PARAMETER NAME>

where the PARAMETER NAME uses the Information Field Name specified below.

Though several Parameter Names may be grouped together by means of a BEGIN/END construct, the command does not allow for carrying additional parameters in order to specify which item of a multidimensional Information Field shall be accessed.

Such additional information must, therefore, be transmitted in advance by presetting one or more pointers; e.g., MATRIX POINTER, LEVEL POINTER, SOURCE POINTER, DESTINATION POINTER.

2 The items of the information field are tallied by the Common Messages:

I/F RESPONSE or SIMULTANEOUS READ RESPONSE

These commands use the format:

<KEYWORD><PARAMETER NAME><PARAMETER VALUE>

where the PARAMETER VALUE carries the Information Field contents specified below.

The parameters that subdefine the individual item of a mutlidimensional Information Field (e.g., MATRIX POINTER, LEVEL POINTER, etc.) are carried within the Parameter Values. Therefore, the content of the Pointers is of no effect in the instant when an Information Field is tallied.

Several names/values may be grouped together by means of a BEGIN/END construct.

3 Multidirectional Information Fields and their corresponding Pointers are individually noted in the description below.

40 COMMAND ERROR STATUS tallies the faults in a failed command, giving the reasons why it could not be executed.

Format: <COMMAND ERROR STATUS>

<PAR GROUP CNT>	1-byte number specifying the number of parameter groups following	
<ERROR CODE>	1-byte code	) first detected
<FAULTY ITEM #>	1-byte number	) fault
. . .		
<ERROR CODE>	1-byte code	) nth detected
<FAULTY ITEM #>	1-byte number	) fault

NOTE – Faulty item #1 contains the first byte of the 2-byte output number. Faulty item #2 contains the second byte of the 2-byte output number. The error codes are:

00 Not used
01 Not used
02 Not used
03 Not used
04 Invalid Level, Input, or Output
05 Not used
06 Not used
07 Output is Locked
08 Take is Excluded
09 Output is Password Protected
0A Not used
0B Not used
0C Not used
0D Point-to-Point Switching Conflict
0E Not used
0F Not used
10 Not used
11 Not used
12 Not used
13 Output is Protected
14 Not used
15 Not used
16 Not used
17 Not used
18 Not used
19 No Paths Left for a Pathfinding Switch

41 MATRIX POINTER defines a pointer to a matrix.

Format: <MATRIX POINTER>

<MATRIX>	1-byte number	(FF_h = all matrices)

42 LEVEL POINTER defines a pointer to a level of the matrix specified in the MATRIX POINTER I/F.

Format: <LEVEL POINTER>

<LEVEL>	1-byte number	(FF_h = all levels)

NOTE – If this field is PRESET to the wildcard FF_h, a READ command referring to this pointer will result in multiple I/F RESPONSES for all available levels.

43 SOURCE POINTER defines a pointer to a source (row) in the level (specified in the LEVEL POINTER I/F) of the matrix specified in the MATRIX POINTER I/F.

Format: <SOURCE POINTER>

<SOURCE>	2-byte number	($FFFF_h$ = all sources)

NOTE – If this field is PRESET to the wildcard $FFFF_h$, a READ command referring to this pointer will result in multiple I/F RESPONSES for all available sources.

44 DESTINATION POINTER defines a pointer to a destination (column) in the level (specified in the LEVEL POINTER I/F) of the matrix specified in the MATRIX POINTER I/F.

Format: <DESTINATION POINTER>

<DESTINATION>	2-byte number	($FFFF_h$ = all destinations)

NOTE – If this field is PRESET to the wildcard $FFFF_h$, a READ command referring to this pointer will result in multiple I/F RESPONSES for all available destinations.

45 Not used

46 DESTINATIONS-TO-SOURCE indicates all destinations (columns) in a specified level of a specified matrix that are currently connected to a specified source (row).

Format: <DESTINATIONS-TO-SOURCE>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<SOURCE>	2-byte number specifying the source
<PARAMETER CNT>	2-byte number specifying the number n of parameters following
<DESTINATION 1>	2-byte number specifying the first destination connected to the source
. . .	
<DESTINATION n>	2-byte number specifying the nth destination connected to the source

NOTES

1 Parameter Count = 0 means: no connection.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and SOURCE POINTER to be PRESET in advance.

47 SOURCES-TO-DESTINATION indicates all sources (rows) in a specified level of a specified matrix that are currently connected to a specified destination (column).

Format: <SOURCES-TO-DESTINATION>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<DESTINATION>	2-byte number specifying the destination
<PARAMETER CNT>	2-byte number specifying the number n of parameters following
<SOURCE 1>	2-byte number specifying the first source connected to the destination
. . .	
<SOURCE n>	2-byte number specifying the nth source connected to the destination

NOTES

1 Parameter Count = 0 means: no connection.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and DESTINATION POINTER to be PRESET in advance.

48 CROSSPOINT STATUS gives a list of the bad crosspoints corresponding to a specified destination (column) in a specified level of a specified matrix.

Format: <CROSSPOINT STATUS>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<DESTINATION>	2-byte number
<PARAMETER CNT>	2-byte number specifying the number n of parameters following
<SOURCE 1>	2-byte number specifying the first of a list of bad crosspoints
. . .	
<SOURCE n>	2-byte number specifying the nth and last one of a list of bad crosspoints

NOTES

1 Parameter Count = 0 means: no connection.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and DESTINATION POINTER to be PRESET in advance.

49 SOURCE SIGNAL STATUS indicates the signal quality of a specified source in a specified level of a specified matrix.

Format: <SOURCE SIGNAL STATUS>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<SOURCE>	2-byte number
<CODE>	1-byte special binary code:
	00_h = good
	else = bad (details may be reported using user-defined codes)

NOTE – Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and SOURCE POINTER to be PRESET in advance.

4A LEVEL CONFIGURATION describes the start-up configuration in the specified level of the specified matrix by first defining the rectangular limits of the level and then detailing crosspoint blocks (typically card-related) which do not exist.

Format: <LEVEL CONFIGURATION>

<MATRIX>	1-byte number	
<LEVEL>	1-byte number	
<FIRST SRC>	2-byte number	rectangular limits of the level
<LAST SRC>	2-byte number	
<FIRST DEST>	2-byte number	
<LAST DEST>	2-byte number	
<PAR GROUP CNT>	1-byte number specifying the number n of parameter groups following	
<SRC# of BLOCK BEG>	2-byte number	data of first nonexistent block
<SRC# of BLOCK END>	2-byte number	
<DEST# of BLOCK BEG>	2-byte number	
<DEST# of BLOCK END>	2-byte number	
. . .		

<SRC# of BLOCK BEG>	2-byte number	
<SRC# of BLOCK END>	2-byte number	data of nth
<DEST# of BLOCK BEG>	2-byte number	nonexistent block
<DEST# of BLOCK END>	2-byte number	

NOTE – Accessing this I/F requires the I/F's MATRIX POINTER and LEVEL POINTER to be PRESET in advance.

4B LEVEL STATUS – MISSING BLOCKS reports blocks of crosspoints (typically card-related) in the specified level of the specified matrix that are detected by the controlled Virtual Machine as missing relative to the start-up configuration.

Format: <LEVEL STATUS – MISSING BLOCKS>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<PAR GROUP CNT>	1-byte number specifying the number n of parameter groups following

<SRC# of BLOCK BEG>	2-byte number	
<SRC# of BLOCK END>	2-byte number	data of first
<DEST# of BLOCK BEG>	2-byte number	faulty block
<DEST# of BLOCK END>	2-byte number	
. . .		
<SRC# of BLOCK BEG>	2-byte number	
<SRC# of BLOCK END>	2-byte number	data of nth
<DEST# of BLOCK BEG>	2-byte number	faulty block
<DEST# of BLOCK END>	2-byte number	

NOTES

1 Parameter Group Count = 0 means: no faulty blocks.

2 Accessing this I/F requires the I/F's MATRIX POINTER and LEVEL POINTER to be PRESET in advance.

4C LEVEL STATUS – ADDED BLOCKS reports blocks of crosspoints (typically card-related) in the specified level of the specified matrix that are detected by the controlled Virtual Machine as having been added relative to the start-up configuration.

Format: <LEVEL STATUS – ADDED BLOCKS>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<PAR GROUP CNT>	1-byte number specifying the number n of parameter groups following

<SRC# of BLOCK BEG>	2-byte number	
<SRC# of BLOCK END>	2-byte number	data of first
<DEST# of BLOCK BEG>	2-byte number	new block
<DEST# of BLOCK END>	2-byte number	
. . .		
<SRC# of BLOCK BEG>	2-byte number	
<SRC# of BLOCK END>	2-byte number	data of nth
<DEST# of BLOCK BEG>	2-byte number	new block
<DEST# of BLOCK END>	2-byte number	

NOTES

1 Parameter Group Count = 0 means: no faulty blocks.

2 Accessing this I/F requires the I/F's MATRIX POINTER and LEVEL POINTER to be PRESET in advance.

4D Not used

4E SOURCE NAME contains the name of the specified source in the specified level of the specified matrix.

Format:	<SOURCE NAME>	
	<MATRIX>	1-byte number
	<LEVEL>	1-byte number
	<SOURCE>	2-byte number
	<CHARACTER CNT>	2-byte number specifying the number n of characters following
	<CHARACTER 1>	1-byte ASCII code, first character
	. . .	
	<CHARACTER n>	1-byte ASCII code, nth character

NOTES

1 This I/F can be PRESET by the controlling Virtual Machine and is to be used only for displaying the names at the switcher, where applicable.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and SOURCE POINTER to be PRESET in advance.

4F DESTINATION NAME contains the name of the specified destination in the specified level of the specified matrix.

Format:	<DESTINATION NAME>	
	<MATRIX>	1-byte number
	<LEVEL>	1-byte number
	<DESTINATION>	2-byte number
	<CHARACTER CNT>	2-byte number specifying the number n of characters following
	<CHARACTER 1>	1-byte ASCII code, first character
	. . .	
	<CHARACTER n>	1-byte ASCII code, nth character

NOTE – This I/F can be PRESET by the controlling Virtual Machine and is to be used only for displaying the names at the switcher, where applicable.

50 LOCKED DESTINATIONS-TO-SOURCE indicates all destinations (columns) in a specified level of a specified matrix whose crosspoint to the specified source is locked.

Format:	<LOCKED DESTINATIONS-TO-SOURCE>	
	<MATRIX>	1-byte number
	<LEVEL>	1-byte number
	<SOURCE>	2-byte number specifying the source
	<PARAMETER CNT>	2-byte number specifying the number n of parameters following
	<DESTINATION 1>	2-byte number specifying the first destination whose cross-point to the specified source is locked
	. . .	
	<DESTINATION n>	2-byte number specifying the nth destination whose cross-point to the specified source is locked

NOTES

1 Parameter Count = 0 means: no connection.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and SOURCE POINTER to be PRESET in advance.

51 LOCKED SOURCES-TO-DESTINATIONS indicates all sources (rows) in a specified level of a specified matrix whose crosspoint to the destination is locked.

Format: <LOCKED SOURCES-TO-DESTINATIONS>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<DESTINATION>	2-byte number specifying the destination
<PARAMETER CNT>	2-byte number specifying the number n of parameters following
<SOURCE 1>	2-byte number specifying the first source whose crosspoint to the specified destination is locked
. . .	
<SOURCE n>	2-byte number specifying the nth source whose crosspoint to the specified destination is locked

NOTES

1 Parameter Count = 0 means: no connection.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and DESTINATION POINTER to be PRESET in advance.

52 EXCLUSION DESTINATIONS-TO-SOURCE indicates all destinations (columns) in a specified level of a specified matrix that cannot be connected to the specified source.

Format: <EXCLUSION DESTINATIONS-TO-SOURCE>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<SOURCE>	2-byte number specifying the source
<PARAMETER CNT>	2-byte number specifying the number n of parameters following
<DESTINATION 1>	2-byte number specifying the first destination that cannot be connected to the source
. . .	
<DESTINATION n>	2-byte number specifying the nth destination that cannot be connected to the source

NOTES

1 Parameter Count = 0 means: no connection.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and SOURCE POINTER to be PRESET in advance.

53 EXCLUSION SOURCES-TO-DESTINATION indicates all sources (rows) in a specified level of a specified matrix that cannot be connected to the specified destination.

Format: <EXCLUSION SOURCES-TO-DESTINATION>

<MATRIX>	1-byte number
<LEVEL>	1-byte number
<DESTINATION>	2-byte number specifying the destination
<PARAMETER CNT>	2-byte number specifying the number n of parameters following
<DESTINATION 1>	2-byte number specifying the first source that cannot be connected to the destination
. . .	
<DESTINATION n>	2-byte number specifying the nth source that cannot be connected to the destination

NOTES

1 Parameter Count = 0 means: no connection.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and DESTINATION POINTER to be PRESET in advance.

58 DESTINATIONS-IN-CHOP indicates all destinations (columns) in a specified level of a specified matrix that are currently in the chop mode.

Format: <DESTINATIONS-IN-CHOP>

<MATRIX>	1-byte number
(LEVEL>	1-byte number
<———>	2-byte space holder (contents not used)
<PARAMETER CNT>	2-byte number specifying the number n of parameters following
<DESTINATION 1>	2-byte number specifying the first destination connected to the source
. . .	
<DESTINATION n>	2-byte number specifying the nth destination connected to the source

59 CHOP SOURCE indicates all sources (rows) in a specified level of a specified matrix that are currently in chop mode to a specified destination (column).

Format: <CHOP SOURCE>

<MATRIX>	1-byte number
(LEVEL>	1-byte number
<DESTINATION>	2-byte number specifying the destination
<PARAMETER CNT>	2-byte number specifying the number n of parameters following
<SOURCE 1>	2-byte number specifying the first source chopping to the destination
. . .	
<SOURCE n>	2-byte number specifying the nth source chopping to the destination

NOTES

1 Parameter Count = 0 means: no chop to this destination.

2 Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and DESTINATION POINTER to be PRESET in advance.

5A CHOP RATE STATUS indicates the chop rate in a specified level of a specified matrix.

Format:	<CHOP RATE STATUS>	
	<MATRIX>	1-byte number
	<LEVEL>	1-byte number
	<CHOP RATE>	4-byte number in TIME format

NOTE – Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and SOURCE POINTER to be PRESET in advance.

5B CHOP TIME STATUS indicates the chop duration in a specified level of a specified matrix.

Format:	<CHOP TIME STATUS>	
	<MATRIX>	1-byte number
	<LEVEL>	1-byte number
	<CHOP RATE>	4-byte number in TIME format

NOTE – Accessing this I/F requires the I/F's MATRIX POINTER, LEVEL POINTER, and SOURCE POINTER to be PRESET in advance.