Printing Protocol Definition for SBP-2 / IEEE1394

Proposal Draft Ver. 0.3~~b~~d
~~July 18~~August 5, 1997

SEIKO EPSON Corporation

EPSON RESEARCH CENTER
EPSON Software Development Center

Fumio Nagasaka
Jason Scott Sanmiya

Table of contents

1. Scope and Purpose

1.1. Scope

This proposal defines a printer access protocol for reliable connectivity over the High Performance Serial Bus IEEE1394. The SBP-2 transport protocol defines a general transport language on the IEEE1394 bus. This proposal defines a printer-specific implementation of SBP-2.

This proposal defines the following parts of SBP-2

- SBP-2 Printing, Command Set
- SBP-2 Printing, Status Block
- Host timeout handling
- SBP-2 Printing, Configuration ROM

1.2. Purpose

This printing definition provides the following features.

- Fast data transfer from host to printer via IEEE1394
- Simultaneous connectivity of multiple hosts to a single printer
- Availability of multiple printers to any host on the IEEE1394 bus

2. Normative References

2.1. Approved references

- ISO/IEC 13213:1994, Control and Status Register (CSR) Architecture for Microcomputer Buses

2.2. References under development

~~–~~Serial Bus Protocol 2 (SBP-2) Working Draft, X3T10 Project 1155D Rev.2e~~1~~, ~~July 17,1996~~July 23, 1997
<ftp://ftp.symbios.com/pub/standards/io/t10/drafts/sbp2/>

- IEEE Std. 1394-1995, Standard for a High Performance Serial Bus

3. Definitions and notation

4. Background of this protocol (informative)

4.1. Problems raised through 1394 PWG discussions

4.1.1. Two logins for bi-directional communication

Do we need to have 2 logins to establish bi-directional communication ?

4.1.2. Lacks of prioritized ORB

When a host device invokes a status request ORB while the printer is printing, the status request ORB will be added at the tail of linked list of ORBs. Then can the printer fetch the status request ORB earlier than other ORBs ?[1]

4.1.3. Login scheduling policy for race condition

Many hosts can invoke login requests to print while the target printer is printing the document from a host. When the printing finished, can the host device which was waiting longer than other hosts login earlier ?

4.2. Solutions of this proposal

4.2.1. Two logins for bi-directional communication

SBP-2 allows Normal command ORBs to have the communication buffer which shall be used for input or output. This printing protocol uses Normal command ORBs to have bi-directional communication between a host device and a printer. In other words, the printer may write reverse directional data on the buffer which specified by the host device and allocated on the hosts device's memory space. Thus the printing session does not need to perform two logins to establish bi-directional communication.

4.2.2. Lacks of prioritized ORB

This protocol invokes two logins to perform one printing session. As section 4.2.1 described, each one login can facilitate bi-directional communication. However, two logins are necessary according to the flowing reason.

When a printer cause some trouble, for example due to lacks of paper, a host device need to send status request to the printer and examine the origin of the trouble. According to the returned result, the host device is able to send some commands, for example paper tray change command, and as the result the host keeps better usability. This means that status requests and commands shall have higher priority than printing data.

A host device invokes primary login and secondary login to print documents. The primary login allows the host device to allocate logical connection to the target device. At this login the host device shall specify unsolicited status enable. Using this connection the host device may request status ORBs or command ORBs. While the printer is printing a document from a host device, other host

device need to wait to start printing session. The printer shall write unsolicited status to solicit printing on the STATUS_FIFO of the waiting host device, when the printer finished previous printing session. Then the host device may request secondary login to start printing. The secondary login may cause second fetch agent which seeks linked list of data ORBs. In other words, this protocol implicitly cause to generate two fetch agents in the target device. The primary fetch agent has higher priority of execution. And priority of execution is a design issue of the implementation of SBP-2.

4.2.3. Login scheduling policy for race condition

As described in previous section, a host device invokes two logins for one print session. In the case of many hosts are connected to the target device through IEEE 1394 high performance serial bus, hosts device may invoke a primary login, while a host device is printing document to the target printer. While the target printer is printing, if other host device invokes secondary login , it will be rejected by the target. However the target device records invoked logins, and the target will reply unsolicited status to solicit login, when the target becomes ready to start printing session. According to this scheme of connection, the target device can solicit the host the host device that was waiting while most longest time.

5. Model (informative)

5.1. Forms of communication between host and printer

This protocol identifies three forms of communication between the host and the printer: status requests, commands, and data. This communication is accomplished by means of SBP-2 ORBs. Status ORBs query printer status and print job status. Command ORBs send print commands such as paper-tray-change and print-head-cleaning. Data ORBs point to host memory buffers from which the printer can retrieve print data.

Status ORBs and command ORBs have higher execution priority than data ORBs. To facilitate these two levels of execution, this protocol requires the host to perform two SBP-2 logins. The primary login initiates the *status/command session*, and the secondary login initiates the *data session.*

5.2. Print job FSM

The following finite state machine illustrates the print job process.

S0, no print activity – The host is not communicating with the printer.

S0:S1 – The host initiates communication with the printer by performing an SBP-2 log-in, thus starting the *status/communication session.* The host is entered in the printer's print-job queue, behind all other pending jobs.

S1, print job pending – The host must wait for the printer to signal that its print job is active. During this waiting phase, the host's *status ORBs* will be processed in a timely manner. The host's *command ORBs* will be completed in a timely manner but with the error PRINT_JOB_NOT_ACTIVE. This prevents pending hosts from unduly affecting active print jobs.

S1:S0 – The host can cancel its pending print job at any time by logging out of its *status/command session.* Note that this allows a host to access the first available printer by placing its print job in every queue on the IEEE1394 bus. When one job becomes active, the host logs out from the other printers.

S1:S2 – The printer informs the host that it's print job is at the front of the queue by means of an unsolicited status write to the *status/command session's status block.* The host, upon receiving this message, will perform a second SBP-2 log-in to initiate the *data session.*

S2, print job active – The printer retrieves printing data through the data session's *linked list of data ORBs.* The printer retrieves *status ORBs* and *command ORBs* through the *status/command's linked list of ORBs.* Unsolicited status is written to the *status/command status block.* The host signals the end of its print job by appending a terminal ORB (defined below) to the linked-list of data ORBs and to the linked-list of status/command ORBs.

S2:S0 – The host logs out when the dummy ORBs have been processed. The printer can terminate the active print job if the host stops responding. A host is considered unresponsive if it *times-out* by not providing printer data, or by not resetting the printer's unsolicited status enabled register. If there are no jobs in the printer's queue, an unresponsive host will not be logged out.

5.3. Active print job fetch process

The following diagram illustrates an active print job. It corresponds to S2 in the FSM above.

To perform logins and logouts the host writes to the Management_Agent register. This process is well documented in the SBP-2 specifications.

To send a status request or a command, the host appends a Status ORB or a Command ORB to the end of the Status/Command session's linked list of ORBs. Then the host rings the Status/Command session's doorbell by writing any value to the Status/Command doorbell register. The Status/Command doorbell has a higher priority than the Data doorbell, so the printer will interrupt data processing to answer the Status/Command ORB. The printer notifies the host that this ORB has been processed, by writing to the Status/Command Status Block. The host can then free the memory and buffer used by the ORB.

Data ORBs are made available to the printer in the same manner, except that Data ORBs are appended to the Data session's linked list of ORBs, and the printer provides ORB completion notification by writing to the Data Status Block. The host can free the memory and buffer used by the ORB after completion notification has been received.

The host *always* reports unsolicited status in the Status/Command status block.

6. SBP-2 Printing, Command Set

6.1. General format of command block ORBs

SBP-2 command block ORBs have the following general format.

The first five quatlets are defined by the SBP-2 standard. The sixth, seventh, and eighth quatlets are specific to this printing protocol's command set.

The notify bit, "n", is set to 1 for all command block ORBs. Thus, the printer must write to the *status/command status block* upon completion of a status ORB or command ORB, and to the *data status block* upon completion of a data ORB.

Transfer speed, "spd", and maximum payload, "max_payload", are not limited by this protocol.
Page tables, specified by the "p" and "page_size" fields, should be used as described in the SBP-2 specifications.

Protocol version gives the current version of the Epson IEEE1394 Printing Protocol Definition. Higher versions shall be compatible with previous versions.

Bits specified by "r" are reserved for future standardization.

ORB_SUBTYPE gives one of the three forms of printer communication that this protocol recognizes, and shall be one of the following:

Value	ORB_SUBTYPE
0	Status ORB
1	Command ORB
2	Data ORB
3	Terminal ORB
4-15	reserved for future ~~standardizatoin~~standardization

The subtype_specific_code gives the command to be performed. For instance, an ORB_SUBTYPE of 0 (status ORB) with a subtype_specific_code of 0 (standard_status_request), requests basic status information from the printer.

Note that the *status/command linked-list of ORBs* should contain only Status ORBs and Command ORBs (ORB_SUBTYPE=0 or 1), and the *data linked-list of ORBs* should contain only Data ORBs (ORB_SUBTYPE=2). Later revisions of this protocol may expand the variety of ORB_SUBTYPES.

For an explanation of the other fields, please see the SPB-2 specification.

6.2. Status ORBs

A host can send status ORBs while its print job is pending or active. Status ORBs have higher execution priority than Data ORBs. Therefore, the printer shall interrupt Data ORB processing to process the status ORB. This ensures low latency status replies. Status ORBs have the same execution priority as command ORBs. So, if numerous status ORBs and command ORBs arrive simultaneously, the printer shall process them in a first come first served manner, before resuming Data ORB processing.

Status ORBs have the following format.

The direction bit, “d”, is set to 1 to indicate that the buffer pointed to by data_descriptor is an output buffer. Note that some status requests use the status block and therefore do not require an output buffer.

ORB_SUBTYPE is set to 0 (STATUS_ORB).

Status request shall be one of the following:

Value	Status Request	Output buffer required
0	Standard status request	No
1	Full diagnostic request	Yes
2	Print job status	Yes
3+	Reserved for future standardization	

If the specified status_request requires a buffer, then the output buffer shall be pointed to by data_descriptor. Otherwise, the data_descriptor, direction bit “d”, page table bit “p”, page_size, and data_size fields are ignored.

For an explanation of the other fields, please see the SBP-2 specification.

6.3. Command ORBs

Command ORBs allow the host to control printer functions. Command ORBs will be processed by the printer only when the host's print job is active. If the host's job is pending, the error "JOB_NOT_ACTIVE" is returned in the *status/command status block*.

Command ORBs have higher execution priority than Data ORBs. Thus, upon ringing the *status/command session's doorbell*, the printer shall interrupt Data ORB processing to fetch and process the Command ORB. Command ORBs have the same execution priority as Status ORBs. Therefore, the printer shall process multiple Command ORBs and Status ORBs in the order that they are received.

The printer must notify the host of command ORB completion as soon as possible. The printer *should not* wait for the command function to finish (e.g. wait for the page feed) before notifying completion, as this would tie up printer resources, preventing the host from querying status. If the printer encounters an error in the command function, it must notify the host by using an unsolicited status write.

Command ORBs have the following format.

Command ORBs may or may not use an input buffer, and may or may not use an output buffer, depending on the type of command. Most likely the input buffer will point to a ASCII string containing a command in one of several print job languages.

The direction bit, "d", is set to 1 since the upper-most data_descriptor will always point to the output buffer, if one is required, and the lower-most data_descriptor will always point to the input buffer, if one is required.

The page table bit "p", page size, and data_size fields affect the data_descriptor for the output buffer. Usually, a buffer is not used, and these fields are ignored.

The input buffer must be contiguous, i.e. it cannot be paged.

ORB_SUBTYPE equals CMD_ORB (1).

Command_request shall take one of the following values:

Value	Command	Output buffer required
0	Reset	No
1	Paper feed	No
2	Self clean	No
3	Change paper tray	No
4+	reserved for future standardization	

Notify and protocol_version are as described above. For an explanation of the other fields, please see the SBP-2 specification.

6.4. Data ORBs

Data ORBs provide a pointer to a buffer containing print data. This protocol sets no restrictions on data type or data format. Appending a dummy ORB to the linked list of ORBs signals the end of the print job. Data ORBs have the following format.

Data_descriptor points to the buffer containing print data. The data buffer corresponds to SBP-2 specifications. For reference, if “p” (page table) is 0, then data_descriptor points to the buffer which contains printing data, and data_size gives the size of this buffer. If p is 1, then data_descriptor points to the page table, and data_size gives the number of entries in the page_table. For a complete explanation, please see the SBP-2 specification.

The direction bit, “d”, equals 0 to specify that the data buffer is an output buffer.

ORB_SUBTYPE equals DATA_ORB (2).

The value of data_type shall be one of the following:

Value	Data_type
0	ASCII text
1	RAW
2	Post Script
3+	Reserved for future ~~standardizatoin~~standardization

Notify and protocol_version are as described above. For an explanation of the other fields, please see the SBP-2 specification.

7. Status Block

7.1. Status block format

The printer writes to a status block to notify the host of ORB completion. The printer also uses the status block for unsolicited status reporting.

The *status/command session* and the *data session* each have their own status blocks, hereafter referred to as the *status/command status block* and *data status block*, respectively. To simplify driver development, all unsolicited status shall be reported in the status/command status block.

Status blocks in this protocol have the following format.

The first two quatlets are defined in the SBP-2 specifications and deal with transport layer error handling. The third and last quatlet is specific to this protocol's method of status reporting.

Len = 2 (size of status block = len + 1 quatlets).

Protocol version specifies the version of this protocol definition.

Error_cause is defined as one of the following:

Value	Error cause
0	No error
1	Printer firmware error
2	Printer hardware error
3	Host driver error
4+	Reserved for future ~~standardation~~standardization

Error_number further qualifies the error_cause, and shall take one of the following values.

Error_cause value	Error_number value	Error_number meaning
0	0	No error, print job active
0	1	No error, print job pending
1	0	Internal communication error (~~eg.~~e.g. 1284.4)
2	0	Out of paper
2	1	Paper jam
2	2	Error in self clean
2	3	Error in paper tray change
2	Etc.	(an error for every printer command)
3	0	Print data not supplied—request faster delivery
3	1	Print data no supplied—terminating print job
3	2	Print job not active (used to decline a command ORB)
4+		reserved for future ~~standardizatoin~~standardization

7.2. Completed ORB notification

When a status write occurs to notify the host of ORB completion, the status block shall have the following form.

Src = 0 or 1. 0 implies, “The status block pertains to an ORB identified by ORB_Offest; at the time the ORB was most recently fetched by the target the next_ORB field did not contain a null pointer”. 1 implies, “The status block pertains to an ORB identified by ORB_offset; at the time the ORB was most recently fetched by the target the next_ORB field was null”.

Resp = response status (REQUEST COMPLETE 0, TRANSPORT FAILURE 1, ILLEGAL REQUEST 2). Will be 0 unless an SBP-2 error has ~~occured~~occurred.

d = 1 if the session has moved to the “dead” state. See SBP-2 specifications for detail.

Sbp_status = additional information that qualifies the response status in resp.

ORB_offset = offset of the corresponding ORB

Protocol version, error cause, and error number are as defined above.

If there were no problems completing the ORB, resp=0 (REQUEST COMPLETE) and sbp_status = 0 (no additional sense to report), error_cause = 0 (no errors), error_number = 0 or 1 (no error, print job active/pending).

7.3. Unsolicited status writes

Unsolicited status is always reported in the *status/command* session. The host’s method of checking for unsolicited status is OS dependent. The host driver must reset the unsolicited status mechanism by writing any value to the unsolicited status enabled register in the printer. The status block shall have the following form.

Src = 2 (the status block is unsolicited and contains device status information)
Resp = 3 (sbp_status meaning is VENDOR_DEPENDENT)
Sbp_status = reserved.
ORB_offset is ignored
Len, protocol version, error cause, and error number are as defined above

8. Timeouts

8.1. Why timeouts are necessary

IEEE1394 provides an environment in which numerous hosts can access the same printer. Thus, if the active host stops responding, it's print job must be terminated to allow the next host in the queue to print. If there are no other hosts in the queue, timeouts will still be handled, but the current print job shall not be terminated as described in "terminating the current host's print job".

8.2. Conditions for printer to be considered unresponsive

The host assumes the printer has stopped responding if one of the following situations occur:
1. the printer has not returned job result for each data ORB.
2. the printer has not reply the answer when the host invokes status request
3. the printer has not fetched ORBs from the host device

8.2.1 Printer does not return job result

T.B.D.

8.2.2 Printer does not reply for status request

T.B.D.

8.2.3 Printer does not fetch ORB

T.B.D.

8.3. Conditions for host to be considered unresponsive

The printer assumes the host has stopped responding if one of the two following situations occur:
1. the host has ~~not~~ stopped providing print data
2. the host has not reset the printer's unsolicited status enable register.

~~8.2.1.~~8.3.1. Host does not provide print data

The host has not provided print data, or a dummy ORB to signify the end of print data.
The printer reaches the end of the linked list of data ORBs, and the last ORB is not a dummy ORB. This implies that the host is not providing data to the printer fast enough, so the printer sends ~~unsolisited~~unsolicited status to the *status/control session's* status block. This unsolicited status will have an error_cause value of 3 (host_driver_error), and an error_number of 0 (print data not supplied—request faster delivery). This signals the host driver to increase the ~~frequecy~~frequency with which printer data is supplied. If the printer waits at the end of the printing data linked list for 5 seconds, it assumes that the host has stopped responding.

~~8.2.2.~~8.3.2. Host does not respond to unsolicited status write

The printer has an error and the active host is not responding to unsolicited status writes

The host must respond to an unsolicited status write by resetting the unsolicited status enabled register in the printer. If, after 5 seconds, it has not, then the printer assumes that the host has stopped responding.

~~8.2.3.~~8.3.3. Terminating the current host's print job

If the current host has stopped responding, and if there are other hosts waiting for printer resources, the printer will terminate the current host's print job. If it can (if its unsolicited status register is set), the printer will write ~~unsolisited~~unsolicited status with values of error_cause=3 (host driver error) and error_number=1 (print data not supplied—terminating print job). Then the printer hardware will reset (paper feed, mode reset, etc), and the next job will be activated by writing ~~unsolisited~~unsolicited status with error_cause=0 (no error) and error_number=0 (no error, print job active) to the next host on the queue.

8.4. Connection assurance while printing

This section outlines how to keep connection while the host device is printing.

8.4.1 Host device provides Tickle ORBs.

The host device shall provide specific data ORB which means the data session is available, when the host device expects it will not be able to provide data to print more than 5 seconds. In this situation the host shall place a tickle ORB in linked list of the data ORBs.
The data format of the Tickle ORB is shown below.

8.4.2 Printer provides Tickle status

When the printer expects it will not be able to fetch data ORBs from the host device more than 5 seconds, the printer shall return an unsolicited tickle status to the host device.
The data format of the tickle status is shown below.

9. Configuration ROM

In accordance to the SBP-2 protocol, the printer must contain numerous implementation specific entries in its configuration ROM. These are outlined below.

9.1. Command_Set_Spec_ID

Command_Set_Spec_ID entry (see 7.3.3 and 7.4.1 in SBP-2 specs)
Specifies the organization responsible for the command set definition for the ~~thargettarget~~.
0x38 | command_set_spec_ID
command_set_spec_ID is obtained from the IEEE/RAC.

9.2. Command_Set

Command_Set entry (see 7.3.4 and 7.4.1...)
Indicates that this printer uses the Epson SBP-2 Printer Protocol Definition.
0x39 | command_set value (<-- ASCII?)

9.3. Command_Set_Revision

Command_Set_Revision entry (see 7.3.5 and 7.4.3 in SBP-2 spec)
Specifies the version of this, Epson's SBP-2 Protocol Definition, implemented by the hardware.
0x3B | command_set_version

9.4. Logical_Unit_Characteristics

Logical_Unit_Characterisics entry (see 7.3.7 ...)
Specifies characteristics of the target implementation.
0x3A | q=0 (target implements basic task management model defined in SBP-2) | o=1 (ordered completion of tasks) | i=0 (no isochronous operations) | 5 bits reserved | login_timeout= ???x500millisec | ORB_size = 6 (quadlets to fetch)

9.5. Bus Info Block

T.B.D.