PART IV. IIASA DATA COMMUNICATION NETWORK DATA LINK CONTROL PROCEDURE

and

PART V. OPTIMIZATION OF THE FRAME RETRANSMISSION MECHANISM

J. H. Sexton

December 1975 WP-75-163

2361
Laxenburg
Austria

International Institute for Applied Systems Analysis

CSN 004
November 1975

IIASA DATA COMMUNICATION NETWORK

DATA LINK CONTROL PROCEDURE

J.H. Sexton

Introduction

This paper presents a definitive version of the low level line protocol to be used in the IIASA Network. A brief summary of this protocol was given in the paper CSN 003 circulated in September 1975. As stated there, it is planned to use the HDLC proposed draft international standard as described in the ISO documents TC97/DIS 3309 and TC97/SC6/1005. As far as possible we intend to keep to this standard. It is not proposed to repeat the contents of the above documents here. Instead the protocol will be defined by describing the subset to be used and those places where we are forced to deviate from the standard protocol. It will therefore be essential to read this paper in conjunction with the above mentioned papers, especially SC6/1005 which defines the protocol.

Frame Structure

Because many of our links have to be operated with existing hardware we are not able to employ the "bit-stuffing" technique described in DIS 3309. Therefore for those links we shall use synchronization characters. Our frames will therefore have the format shown below.

First Frame

SYN	SYN	SYN	SYN	ADDRESS	CONTROL	INFORMATION	CRC
1 byte	1	1	1	1	1 byte	0 or more bytes	2 bytes

Intermediate Frames

SYN	SYN	ADDRESS	CONTROL	INFORMATION	CRC
1	1	1	1	≥ 0	2 bytes

Last Frame

SYN	SYN	ADDRESS	CONTROL	INFORMATION	CRC	PAD
1	1	1	1	≥ 0	2 bytes	1

FIGURE 1

It will be seen that the first of a set of frames sent in one direction has four SYN characters at its head, while all others have two. And that only the last of such a set has an additional PAD character.

When it exists, the information field will be terminated by ETB.

To achieve transparency and avoid premature termination due to the chance occurrence of ETB in the data, all occurrences of SYN, DLE and ETB in the address, control and information fields will be immediately preceded by one DLE character, excluding of course the final ETB at the end of a non-empty information field.

The above mentioned control characters will have the values shown in the Table below.

Character	Octal Value (7 bit)
SYN	026
DLE	020
ETB	027
PAD	177

The frame formats shown will be used even for asynchronous links, since this will facilitate changes to synchronous modems.

Command Subset

We shall use only Normal Response Mode in half-duplex. If it is later decided to operate some links in full duplex, then a revision of this paper will be made.

Supervisory and Unnumbered Commands will be limited to the following set:

RR, RNR, SNRM, DISC

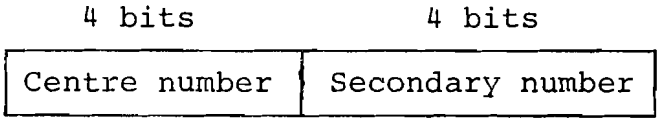
and Supervisory and Unnumbered Responses to:

RR, RNR, UA, CMDR.

We shall not use the extended modes.

Addressing

In agreement with ISO standard DIS 3309 we shall use the address field to identify the secondary, but will divide that field into two subfields as shown in Figure 2.



Address byte

Table of Centre Numbers

Centre	Number
IIASA	1
Bratislava	2
Budapest	3
Kiev	4
Moscow	5
Vienna	6
Warsaw	7

Figure 2

Cross Linking

To enable cross links to be detected, each secondary shall have two addresses, an even and odd pair of consecutive integers, e.g. 0 and 1. Frames transmitted by a primary will use the even address while those transmitted by a secondary will use the odd address. For local looped testing, secondaries will be permitted to transmit frames with the even address, but on those occasions only.

The centre number will always be that of the secondary.

Symmetry

To overcome the unsatisfactory assymmetry between primary and secondary stations we shall adopt the following solution. On each link there shall be two logical channels with a primary at each end,

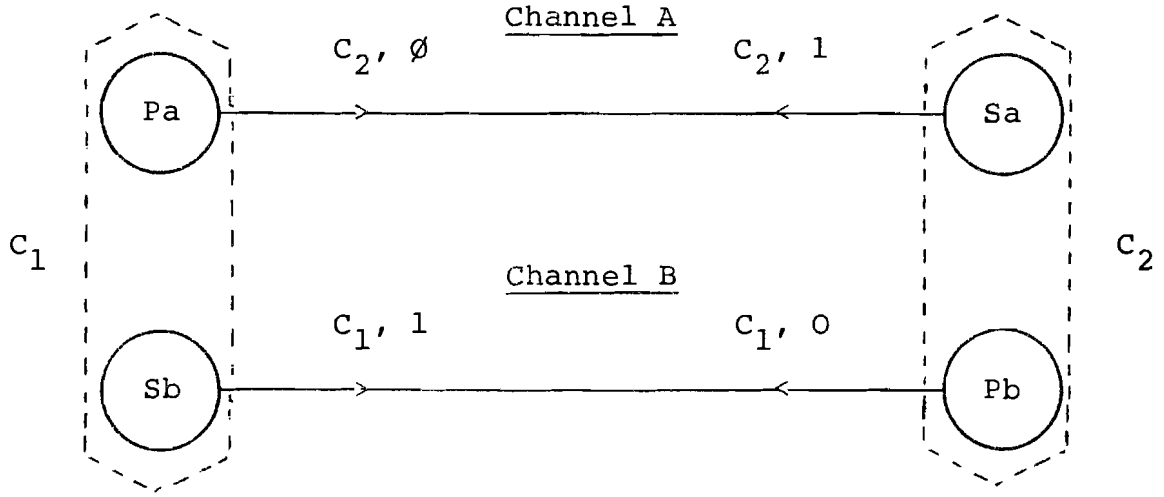


Figure 3

Primary Pa and Secondary Sa shall communicate with one another using the centre number for C_2, while Primary P_b and Secondary Sb shall communicate using the centre number for C_1. The Send and Receive Sequence Numbers shall be distinct for each channel in accordance with the standard.

In order to avoid confusion between the traffic on the two logical channels, the Primary and Secondary at a given centre shall cooperate according to the following rule.

If there is information to be sent then it shall be sent on the channel last used, unless the line has been quiescent for a certain time (T_1 a constant given later). When the line has been quiescent for this time a station wishing to restart activity will do so by transmitting from its Primary, which may or may not be the same logical channel as before. The clash which can occur in this quiescent situation is resolved by having a different inter-poll timeout T_2 for each primary. The centre which has the lower centre number will have the shorter timeout.

Symmetrical Recovery from Command Errors

The HDLC proposed draft does not provide primary stations with a means of informing secondary stations of protocol violations or some other exceptional conditions. This is a serious ommission from the protocol and we shall therefore make the following addition. Primary stations may transmit the CMDR response to report that one of the following conditions resulted from the receipt of a frame without FCS error from the secondary:

- the receipt of a response that is invalid or not implemented;
- the receipt of an I frame with an information field which exceeded the buffer size available;
- the receipt of an invalid N(R) count from the secondary.

The use of the information field as defined in section 5.3.2.2 of SC6/1005 will apply equally to this response when "command" is replaced by "response" and "secondary" is replaced by "primary." Secondary stations may use the information received in such a CMDR.

Since the above addition to the protocol may be considered a violation the secondary will respond to a CMDR with a CMDR. The primary should set the P bit in its CMDR and the secondary should set the F bit in its CMDR reply.

Recovery from a CMDR

Recovery from a CMDR is not defined in document 1005. For our purposes it will be as follows:

At the Secondary

Recovery is considered to have been effected when a DISC or SNRM is received or a legitimate command or information frame is received which "reverses" the error i.e.

if the CMDR was caused by an invalid command then a valid command "reverses" it;

if caused by an invalid N(R) then receipt of a valid N(R) "reverses" it;

if caused by too big an information field then an information field which is not too big "reverses" it.

At the Primary

Receipt of a CMDR from the secondary which rejects the CMDR from the primary.

It is recommended that all CMDRs be logged.

Optimization of the Frame Re-transmission Mechanism

It is proposed to follow section 6.2 paragraph a "REJ and POLL/FINAL BIT RECOVERY" from an "N(S) sequence exception," which requires that all following I frames be re-transmitted i.e. after a sequence error all previously transmitted frames will be re-transmitted and no optimization will be attempted for the present. The use of SREJ or other more overt methods will be postponed for future consideration.

Timeouts

The following timeouts are required by HDLC for the NRM case:

T_1 Active to Idle

A spacing condition on a line lasting for more than T_1 secs, will be taken to indicate the line is idle or quiescent and either primary is free to attempt transmission.

T_2 Inter-polling Time

This is the Primary reply timer, defined in Annex B, section 1.1.

T_3 Broken Line Condition

In the event that a primary is unable to solicit any response from a secondary after repeated attempts for a period of T_3 secs, the line may be presumed down.

Experimental Values

For experiment the following values are proposed:

T_1 1 sec.

T_2 2 sec for the centre with the lower number,
2.1 sec for the other centre.

T_3 1 minute.

States for a Logical Channel

Two states are defined:

"Working" and "Down."

Initially a channel is considered "Down."

Changes of State

"Down" to "Working"

At a primary: receipt of a valid UA in response to a SNRM command.

At a secondary: receipt of a valid frame other than SNRM.

"Working" to "Down"

At a primary: receipt of a valid UA in response to a DISC command. Timeout occurring, i.e. the broken line condition.

At a secondary: receipt of a valid DISC command.

The receipt of additional SNRM and DISC commands by a secondary should not be treated as errors. The secondary should always reply with UA and in the case of SNRM, re-initialize.

Use of Poll/Final Bit

The Poll bit will always be set on the last frame of a set of frames transmitted by the Primary.

The Final bit will always be set on an unnumbered response.

After transmitting a frame with P/F bit set, "Request to Send" must be removed, i.e. carrier must be suppressed.

Bit Numbering and References to Low Order Bits

The HDLC documents are written with the left-most bit numbered 1. While this is common in the USA, it is not so in Europe and can cause confusion. It is therefore recommended that the bit numbering in all diagrams be reversed and numbered from zero. Thus a single byte would be numbered:

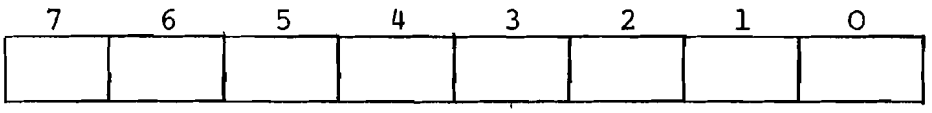


Figure 4

In any case left-most bits will be considered high order, i.e. they will have the highest value and will be transmitted first. Therefore references to left-most bits as low order in sections: 3.1 and 5.3.2.2 are to be treated as errors.

The values shown for fields, e.g. in section 5.2 are to be taken as correct.

CMDR Information Field

The option of padding the information field with 4 zero bits to 24 bits in all, will be taken, and it will be transmitted in the order shown below:

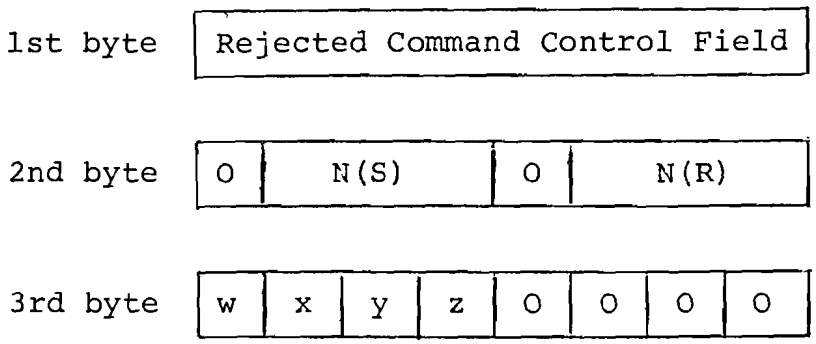


Figure 5

The Send State Variable V(S)

Section 3.2.3 is not sufficiently definite about what should be done when an error occurs and re-transmission is necessary.

The following is recommended:
V(S) is normally incremented with each new frame transmitted, but when re-transmission is necessary V(S) is put back to the receiver's N(R) and then continues to be incremented by one after each frame sent.

Miscellaneous Notes

The Abort capability mentioned in 3.3.1.1 will not be used.

Inter-frame time fill will not be used.

The use of the Parity bit is not defined, although it is recommended that it should be zero in control bytes.

The Cyclic Redundancy Check

The cyclic redundancy check will be computed after the insertion of DLE characters and will be generated for all bytes of the address, control and information fields, including the final ETB in a non-empty information field. All 8 bits of each byte will be included, whatever the use of the parity bit.

It shall be the remainder after the multiplication by x^{16} and then division (modulo 2) by the generating polynomial

$$x^{16} + x^{12} + x^5 + 1 \quad , \quad \text{i.e. CRC - CCITT} \quad .$$

Concluding Remarks

Any comments on the above notes should be addressed to the author. It is proposed to revise the decisions made here after a few months particularly concerning full-duplex operation and optimization strategies.

For background information refer to the earlier numbers in this series viz: CSN 001, CSN 002 and CSN 003. Another paper CSN 005 is available which explains some of the choices made here.

IIASA DATA COMMUNICATION NETWORK

HDLC

DATA LINK CONTROL PROCEDURE

OPTIMIZATION OF THE FRAME RETRANSMISSION MECHANISM

Introduction

It would be possible for a destination station to keep all correctly received frames and for a station that has to retransmit frames in error to omit re-transmitting some frames. The HDLC document TC97/SC6/1005 was studied to see whether an optimization along these lines is permitted.

The following conclusions were drawn:

1) The HDLC document is not only vague but also ambiguous;

2) It is explicitly forbidden in SDLC, IBM's version of HDLC;

3) The spirit of HDLC and the example of SREJ are in favour of optimization.

Referring to the HDLC 1005 paper:

Section 6.2 paragraph a "REJ and POLL/FINAL BIT RECOVERY" from an "N(S) sequence exception."

The word "may" in this paragraph confuses the meaning which is not altered, but clearer, if it is omitted, since one cannot re-transmit frames which have not been transmitted before. One possible explanation of the purpose of "may" in this sentence is that the author wished to make it clear that the number of I frames following could be zero.

This paragraph requires that all following I frames be re-transmitted. Thus explicitly forbidding the optimization that we were considering. However, section 6.2.1 is clearly meant to expand and clarify paragraph a.

Section 6.2.1 paragraph "POLL/FINAL BIT RECOVERY."

This covers our case, since we do not plan to use REJ. This differs in an important way from paragraph a, because the word "all" is not used. In its place is found the word "any." It could be argued that the meaning of "any" here is "all" and covers the zero case, as with the "all"--"may" combination in paragraph a. However, this is open to doubt and allows the following example scheme of use:

Following an "N(S) sequence exception." The sender re-transmits the first unacknowledged frame as one TUK* intending to send the remaining previously transmitted, but unacknowledged I frames in succeeding TUKs. If the receivers desire for re-transmission is satiated by this one frame he may update N(R) by more than one. In this case, at the sender, the set of "any previously transmitted, but unacknowledged, I frames" is reduced by more than one and he is now in a new state in which he is no longer obliged to fulfil his earlier intention of sending all the previously remaining transmitted but unacknowledged I frames.

Conclusion

I consider that what I have done above is no more than to make a logical analysis of what is written in document 1005, but we have no guarantee that what is written there expresses exactly what the authors had in mind. Therefore we had better make an intelligent guess than rely on such an analysis.

THE SEND STATE VARIABLE V(S)

The following notes in may opinion have no bearing on the above question except in so far as they illustrate the vagueness of the HDLC paper.

Section 3.2.3

When an error occurs and re-transmission is necessary it is not clear from this section whether V(S) is put back to its earlier value, since it speaks only about in-sequence information frames. The first re-transmitted frame, at least, is not in-sequence. One can imagine two cases:

a) V(S) is normally incremented with each new frame transmitted, but that when re-transmission is necessary, V(S) remains static, until new frames are reached when V(S) continues to be incremented.

   This is not in disagreement with section 3.2.4 because that section also talks about in-sequence frames, and frames to be re-transmitted could be kept in a buffer with their N(S) values as set when they were born.

---

*TUK is defined to mean all the frames transmitted in one direction between successive changes of direction of transmission, and is particularly meaningful in the half-duplex case, where TUKs alternate in direction.

b) V(S) is normally incremented as above, but when re-transmission is necessary V(S) is put back to the receivers N(R) and then continues to be incremented by one after each frame sent.

In either case the receiver will see no difference The second paragraph of section 6.2 says that retransmitted frames may contain an updated N(R) and/or P/Fbit. This is in favour of case b) above where re-generation of frame envelopes would be more straight forward.

Note

The sender has no way of knowing that an "N(S) sequence exception" has occurred except by comparing V(S) with the receivers N(R). This is unreliable since, referring to the section above describing the send state variable, if scheme A, were used, then using the above method to detect "N(S) sequence exception" would in some cases find spurious exceptions, i.e. cases which are not genuine.

Proposal

Let us adopt the SDLC approach for the present, throw away all frames received after an error and require that the source re-send them all. If later we are unhappy about this we can always change our minds.

John Sexton
Computer Science

22.10.75