International Telegraph and Telephone
Consultative Committee
(CCITT)
Period 1985-1988

Doc # 252
May 1987
Original: English

Question: 4/XV, Specialists Group

STUDY GROUP XV - CONTRIBUTION

Source: AT&T

Title: Proposed Algorithm for Zero Byte Replacement with n x 384 kbit/s Video Conferencing

ABSTRACT

In order to avoid all zero bytes with the n x 384 kbit/s video codec standard, the algorithm described in this contribution has been developed. It provides for the substitution of all zero bytes with bytes that contain a minimum of one "1". The algorithm will allow for encryption of all of the data with the exception of the eighth bit position of the first time-slot. It could easily be implemented as an option on video codecs that can be activated or deactivated depending on the existence of bit sequence independence.

General Discussion of Algorithm

The zero substitution algorithm requires that the n x 384 kbit/s frame structure be divided into sectors. Figure 1 shows the sector layout for values of n from 1 to 4. It is assumed that there is no need for this algorithm for applications where n = 5. Depending on the value of n, varying numbers of bits of the application channel of the audio and service channel (See Figure 3) are required for use as flag bits. One bit is required for each sector.

Specific details of the decoder algorithm are provided in this document. It is assumed that in providing details of the decoder, there is sufficient information to design an encoder/decoder pair.

NOTE: Only 7 bits of time-slot 1 should be coded. This is the audio and control time slot and the eighth bit is for framing and other uses including the application channel. Part of the application channel is used for zero byte coding.

In executing the zero byte coding algorithm, bit eight of time slot 1 must not be altered. Zero byte coding for sub-blocks 1 through 8 would operate in the same way as with other sub-blocks except the operation is restricted to the first seven bits of the byte in time-slot 1.

Decoder for replacing Zero Bytes

For the purposes of zero code substitution, the 80 rows composing a n x 384 kbit/s frame are divided into 8 equal sub-blocks of 10 or 20 rows each as shown in Figure 1. These sub-blocks are numbered 1 through 20 and one flag bit each is assigned using bits 61 through 80 of the application channel depicted in the frame structure of the audio and service channel (See Figure 3). Bit position 61 is associated with sub-block 1, bit position 62 with sub-block 2, etc. The algorithm for decoding the incoming data must function as follows:

Step 1: If the flag bit for a sub-block is "1", go to Step 2 (flag bit equal to "1" indicates that there is at least one zero byte in this sub-block);

Otherwise, go to Step 5.

Step 2: Read byte position 1* (the numerical identifier of the byte position within a sub-block is given in Figure 2);

If the address is 000001 (this indicates that byte 1 is the only zero byte) place all zeros (7 zeros in the first 7 bit positions if time-slot 1) in byte position 1, then go to Step 5;

Otherwise, go to Step 3.

Step 3: Read byte position in the first 7 bit positions indicated by address in bits 1 through 6/7*, place all zeros (7 zeros if time-slot 1) in current byte position (i.e. byte position given by previous read statement.

Step 4: If bit 7/8* is "0", go to Step 3;

If bit 7/8* is "1" and address is 000001 place all zeros (7 zeros in the first 7 bit positions if time-slot 1) in Byte Position 1 and go to Step 5;

If bit 7/8* is "1" and address is not equal to 000001, place the contents of the byte position given by the address in Byte Position 1, place all zeros (7 zeros in the first 7 bit positions if time-slot 1) in the byte position given by the address, then go to Step 5.

Step 5: Go to the next sub-block or end if last sub-block.

*Structure of Zero Substitiution Code Word

Bits 1 through 7 (bits 1 through 6 for sub-blocks 1 to 8) is the address of the next zero byte. Figure 2 provides the address of each byte within both small and large sub-blocks.

Bit 8 (bit 7 for sub-blocks 1 through 8) is a flag bit that indicates that the address given in bits 1 through 6/7 is the address of the last zero byte.

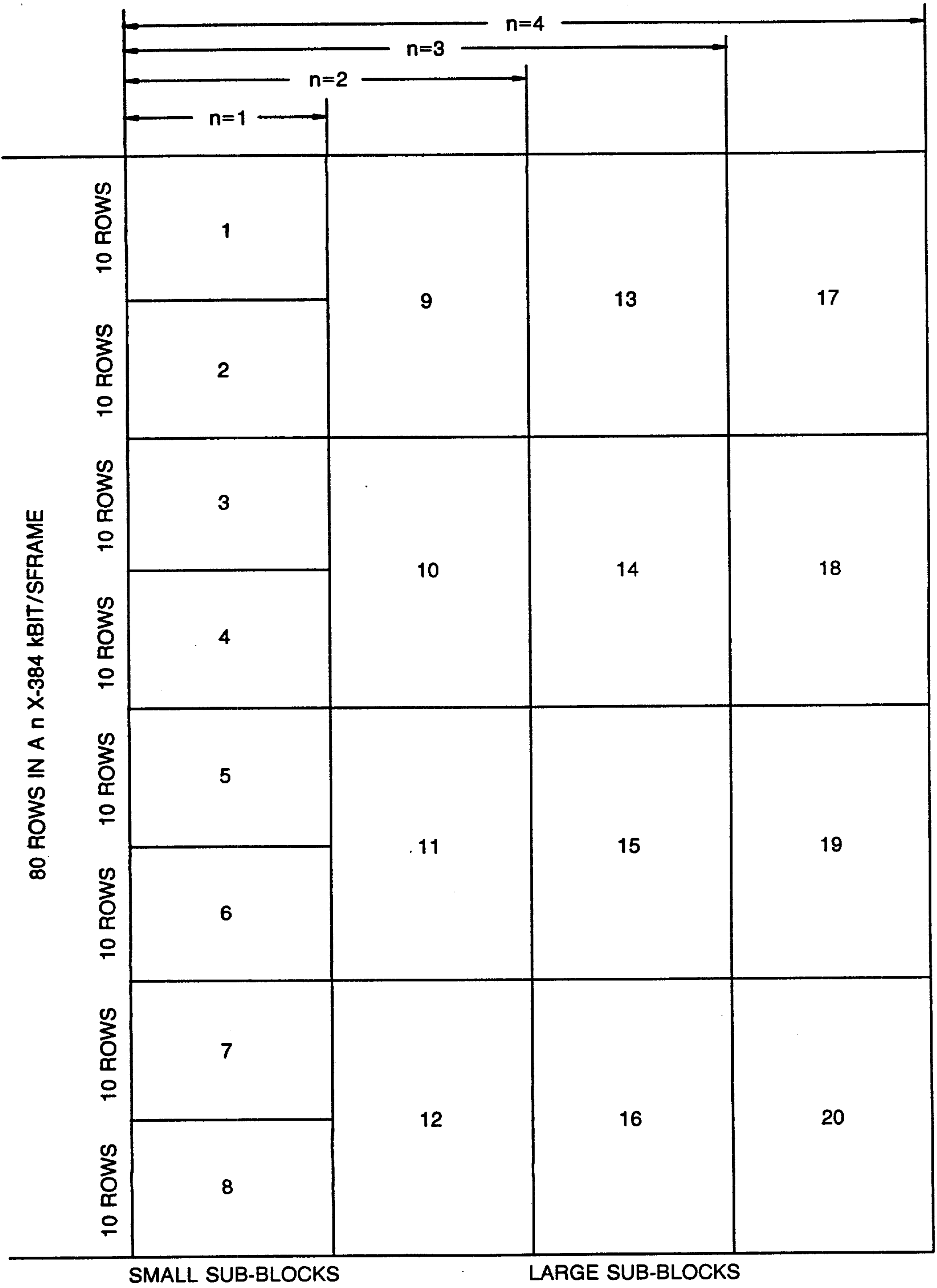


FIGURE 1

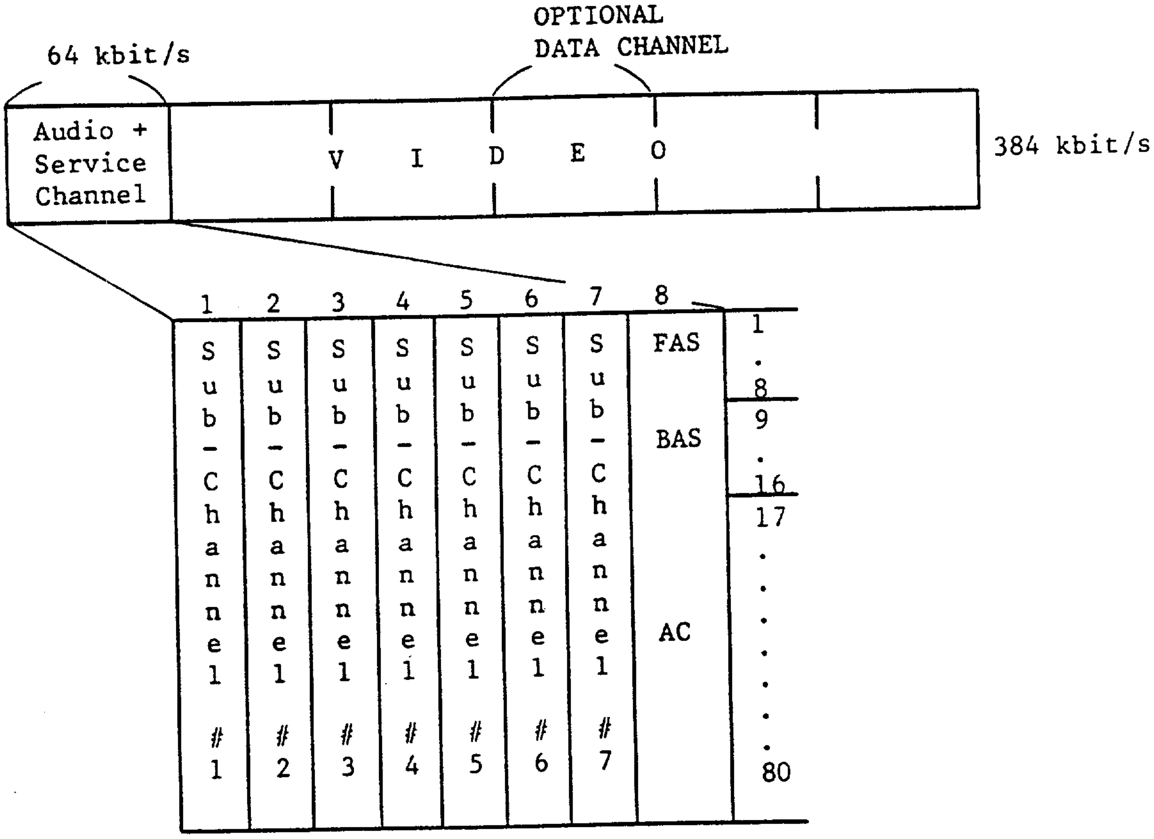
5BD5144

	TS1	TS2	TS3	TS4	TS5	TS6	TS7	TS8	TS9	TS10	TS11	TS12
ROW 1	1	2	3	4	5	6	1	2	3	4	5	6
2	7	8	9	10	11	12	7	8	9	10	11	12
3	13	14	15	16	17	18	13	14	15	16	17	18
4	19	20	21	22	23	24	19	20	21	22	23	24
5	25	26	27	28	29	30	25	26	27	28	29	30
6	31	32	33	34	35	36	31	32	33	34	35	36
7	37	38	39	40	41	42	37	38	39	40	41	42
8	43	44	45	46	47	48	43	44	45	46	47	48
9	49	50	51	52	53	54	49	50	51	52	53	54
10	55	56	57	58	59	60	55	56	57	58	59	60
11	1	2	3	4	5	6	61	62	63	64	65	66
12	7	8	9	10	11	12	67	68	69	70	71	72
13	13	14	15	16	17	18	73	74	75	76	77	78
14	19	20	21	22	23	24	79	80	81	82	83	84
15	25	26	27	28	29	30	85	86	87	88	89	90
16	31	32	33	34	35	36	91	92	93	94	95	96
17	37	38	39	40	41	42	97	98	99	100	101	102
18	43	44	45	46	47	48	103	104	105	106	107	108
19	49	50	51	52	53	54	109	110	111	112	113	114
20	55	56	57	58	59	60	115	116	117	118	119	120

SMALL SUB-BLOCKS

LARGE SUB-BLOCKS

FIGURE 2



FAS: Frame Alignment Signal (note 1)
BAS: Bitrate Allocation Signal
AC: Applicaion Channel

Note 1: The block termed as FAS contains also other information than for frame alignment purposes.

Fig. 3 Frame Structure for n x 384 kbit/s codec (in case of n = 1)