NCR Systems Engineering B.V.
SUBMISSION TO IEEE 802.4L
<u>THROUGH-the-air Token Bus Physical Layer</u>

THEO KLEIJNE MARCH 1989
NCR SYSTEMS ENGINEERING
ZADELSTEDE 1-10
3431 JL NIEUWEGEIN-HOLLAND
(0)3402-76444

SUBJECT: Start and End Delimiter Implementation.

Start Delimiter - End Delimiter.

1. Introduction.

On MAC level: start delimiter N N 0 N N 0 0 0

end delimiter N N 1 N N 1 I E

Requirements for the Start and End Delimiter on the medium are:

- distinguishable from data on the medium;
- robust, Hamming distance of 4.

In DQPSK, data dibits are transmitted on the medium.

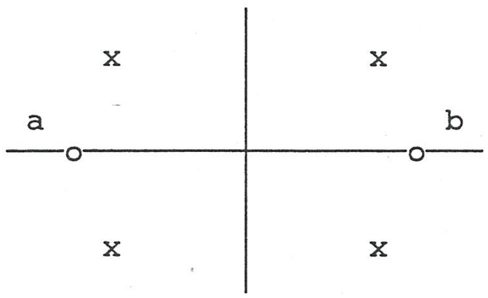

Fig. 1: location of the phase points
x = data dibit, 00 - 01 - 10 - 11
o = non_data dibit's a and b

The start delimiter is a unique octet while the end delimiter contains one information bit and one error bit, resulting in four different end delimiter octets.

2. Proposal.

It is proposed to built delimiters with the phase points a and b. To make it sufficiently robust, it is proposed to send 3 * 4 phase points in a particular order, to provide the delimiter function. This is equivalent in time with 3 octets. The I and E information in the end delimiter will be contained in a fourth octet.
Detection is done by means of a correlation.
It is calculated what the reliability is of recognition of k phase points out of 12. See section 3.

Thus a start delimiter will use three octet times on the medium and the end delimiter will use four octet times on the medium.

To prevent buffering problems between multiple frames in one transmission, the pre-amble period between frames, generated by the MAC, should be sufficiently long to provide the required time for the physical layer to implement the delimiters.

A start delimiter will be: |s s s s|s s s s|s s s s|
s = a or b

An end delimiter will be: |e e e e|e e e e|e e e e|x x x y|
e = a or b
x = 01 data
y = IE (information bit and error bit

An optimal code will be selected for both delimiters to achieve a reliable correlation result and to provide an hamming distance of 4 for both delimiters. This hamming distance means that any 4 errors will not result in any other valid code word.

3. Delimiter performance

In discussing delimiter performance, two probabilities have to be considered:

- P1 = P(incorrect delimiter|delimiter send)
  The probability that less then k of the n delimiter dibits are received correctly.

- P2 = P(correct delimiter|data send)
  The probability that k or more correct non_data symbols are detected out of data.

- P3 = P(detect start delimiter as end delimiter and v.v.)

In these calculations synchronization is assumed. When the detection mechanism is one (or two) octet(s) out of sync, the probability that a delimiter is detected is always neglectable compared to P2 because of the correlation mechanism. So:

P(correct delimiter|combined data and delimiter) << P2

Setting the requirements for P1 and P2 the following was used: the probability that a frame error is caused by a delimiter error (P1 or P2) has to be neglectable compared to the probability that a frame is incorrect because of a data error.
This means, when M is the frame length in bits:

P1 << P(data)*M	P(data) = FER[1] per bit P(data)*M = error probability of frame
P2*M/2n << P(data)*M	In one frame there are M/2n blocks of n dibits

Considering a P(data) of 10**-8 then both P1 and P2 have to be smaller then 10**-8 * M.
Table 1 shows the calculation results of P1 and P2 for n=4 and k=3.

---

1 Frame Error Ratio

	n=4	n=8	n=12	n=16	
k=3	8.0E-3				=P1
	3.2E-6				=P2
k=4		3.7E-6			=P1
	--	5.2E-7			=P2
k=6			6.8E-8		=P1
			5.8E-10		=P2
k=7			2.1E-6		=P1
			4.7E-12		=P2
k=8				1.9E-9	=P1
				7.0E-13	=P2

TABLE 1. Calculation results.

k = number of phase points to be detected
n = number of phase points transmitted
P1 = probability that the delimiter is not received correctly
P2 = probability that the delimiter is correctly detected out of data