FR, FR_MAC, FR_MAC_INV and frame processing

(Or how to open a can of worms...)

IEEE 802.5 Meeting ‘EE’

Document 11-11

Neil Jarvis

SilCom UK (A Microvitec Group Company)

OK, so what's the problem?

- What should a MAC do with the following frame?
  - AC-FC-DA-SA-VL-VC-FCS
    - FC indicates MAC frame
    - DA = any_recognised_address
    - VL = 2
    - VC = 00
    - VI is missing, but first byte of FCS is X'09'
    - Valid FCS and no code violations

Transitions that may fire in a DTR Station for this frame.

- R3501: FR & FSJC=1 & FSOP=1 & FIPTXIS=0
  => M_UNITDATA.indication
- R3540: FR_MAC(DA=any_recognized_address & DC=0 & VI=09)
  => DISCARD_PDU
- R3630: FR_MAC_INV(ERR_COND=SHORT_MAC & SC_PRESENT & SC<>RS)
  => TXI_RSP_PDU(…)
- And others depending on other errors in VL, VC and VI.

Why the ambiguity?

- FR_MAC is defined as:
  - A Valid MAC frame is received. Which is further defined as:
  - A frame with FF bits set to 00, containing no code violations and a valid FCS is received.
- No mention of frame verification
  - But FR_MAC is used with criteria involving
    - DA, SA, VC, VI, A and C bits.
  - These imply some verification…

OK, so what are you proposing?

- An attempt at making the definitions for FR, FR_MAC etc. be consistent with the state transition tables.
- An attempt to redraw/redefine the frame processing flow diagrams in clause 10 to match these new definitions and state transitions.

Definitions I: FR

- Criteria Used
  - DA, P, RI_PRESENT and RI_NOT_PRESENT
- Old Definition
  - A frame has been received and meets the frame receive criteria specified in 4.3.2
- New Definition
  - A frame has been received and meets the frame receive criteria specified in 4.3.2, and is long enough to contain at least AC-FC-DA-SA-FCS.

Definitions II: FR_WITH_ERR

- Criteria Used
  - E
- Old Definition
  - A frame is received with errors (see 4.3.2)
- New Definition
  - A frame is received with errors (see 4.3.2), or is not long enough to contain AC-FC-DA-SA-FCS.

Definitions III: FR_MAC

- Criteria Used
  - DA, SA, VC (DC and SC), VI, A and C
- Old Definition
  - A valid MAC frame is received.
- New Definition
  - A valid MAC frame is received, and VL, VC and VI have passed verification. No subvectors have been verified.

Definitions IV: FR_LLC

- Criteria Used
  - DA and A
- Old Definition
  - A valid LLC frame is received.

Definitions V: FR_MAC_INV

- Criteria Used
  - SC and ERR_COND
- Old Definition
  - A valid MAC frame has been received which has failed verification (see 10.3.6)

Definitions VI: FR_xxx

- Criteria
  - DA, SA, SC, SC_PRESENT, SC_NOT_PRESENT, CORR_PRESENT, CORR_NOT_PRESENT, SV data, A, C, M, R, and E.
- Old Definition
  - A verified xxx MAC frame is received.

Frame Processing: Phase I

- If frame is not valid as defined in 4.3.2 (or 9.1) or is not long enough to contain AC-FC-DA-SA-FCS
  - then FR_WITH_ERR() and exit.

Frame Processing: Phase II

- Verify and parse DA
- Verify and parse SA
- FR()
- If FF bits of FC are 01
  - then FR_LLC() and exit

Frame Processing: Phase III

- Parse VL, VC and VI
  - May generate ERR_COND
    - SHORT_MAC, VL_LTH_ERR, VI_UNK, RI_INVALID, SC_INVALID
- If ERR_COND
  - then FR_WITH_ERR() and exit
  - else FR_MAC()

Frame Processing: Phase IV

- If DA not any_recognised_address or DC is not me
  - then exit

Frame Processing: Phase V

- Parse SVs
  - May generate ERR_COND
    - VL_LTH_ERR, LONG_MAC, SV_MISSING, SV_UNK, SV_LTH_ERR
- If ERR_COND
  - then FR_MAC_INV()
  - else FR_xxx()
- Exit

What about that bad MAC frame?

- With the new definitions, the frame is unambiguously dealt with by the following transition:
  - R3630:
    FR_MAC_INV(ERR_COND=SHORT_MAC
    & SC_PRESENT & SC<>RS)
    => TXI_RSP_PDU(…)

Gosh, this is all just marvelous. Now what?

- Too late to be normative?
- Fix flow charts in clause 10 to be not quite so wrong (FR_MAC shown in wrong place).
- I propose writing a new informative annex with these flow charts and some descriptive words
- Or I could just give up and become a consultant...