

TO: T10 Membership
FROM: Paul A. Suhler, Certance
DATE: 14 November 2003
SUBJECT: T10/03-367r3, ADT Annex: Error detection and recovery action examples

Revision 3:

- Changes from 3 November working group meeting (T10/03-378r0)

Revision 2:

- Various changes for 3 November working group meeting

Revision 1:

- Changes to clarify transmission of multiple IUs without receiving Acknowledgement IUs
- Fixed incorrect labels
- Added P4:Initiating Recovery state and added labels for sender actions.
- This is not entirely consistent with 03-355r2, which does not yet show a P3:P2 transition upon receipt of an Initiate Recovery IU.

Revision 0:

- Initial version

Background

In the process of preparing proposals for error detection and recovery, we found that drawings made comprehension easier. This document offers some such as an informative annex for ADT.

Changes

Add the following as an Annex.

Annex A

(Informative)

Error detection and recovery action examples**A.1 Introduction**

This informative annex diagrams various error detection and recovery procedures for ADT devices conforming to this standard. The conventions for the diagrams are shown in table A.1.

Table A.1 – Diagram Drawing Conventions

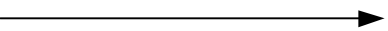
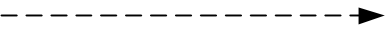
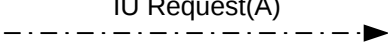
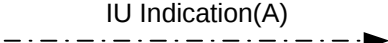
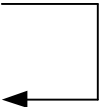
Drawing Convention	Meaning
	Acknowledged or Unacknowledged IU
	IU with error
	Acknowledgement IU
	Service request from upper layer of sender
	Service indication to upper layer of receiver
	Service confirmation to upper layer of sender
	Time-out value exceeded
	IU received is processed to transmit IU
X	IU lost or dropped
PR	Value of the Pending Recovery (PR) bit in the NAK IU.
EFN	Value of the port's Expected Frame Number counter
NFTS	Value of the port's Next Frame To Send counter.
a = b	Counter a is set to the value of expression b
(a == b)	Expression a equals expression b
(a != b)	Expression a does not equal expression b
(condition) => action	Because the condition is true, the action is performed

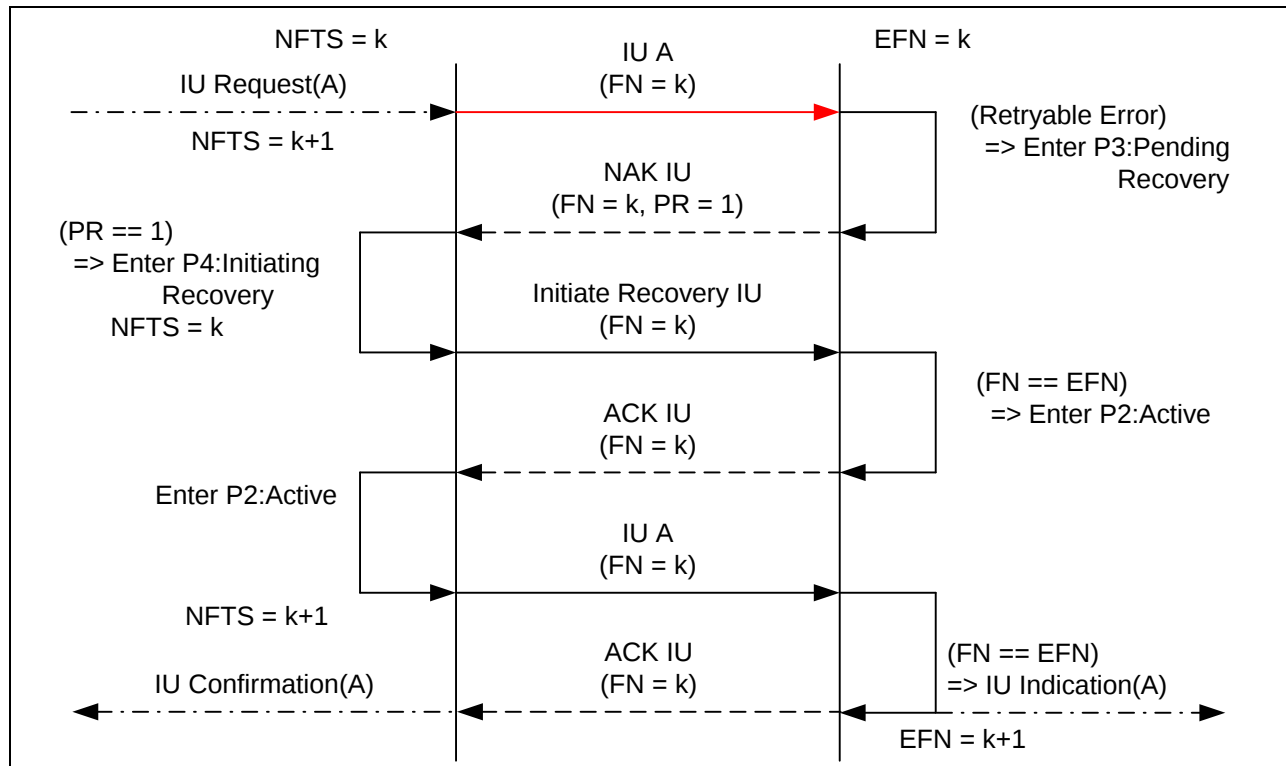
Figure A.1 – Receiver-Detected Retryable Error

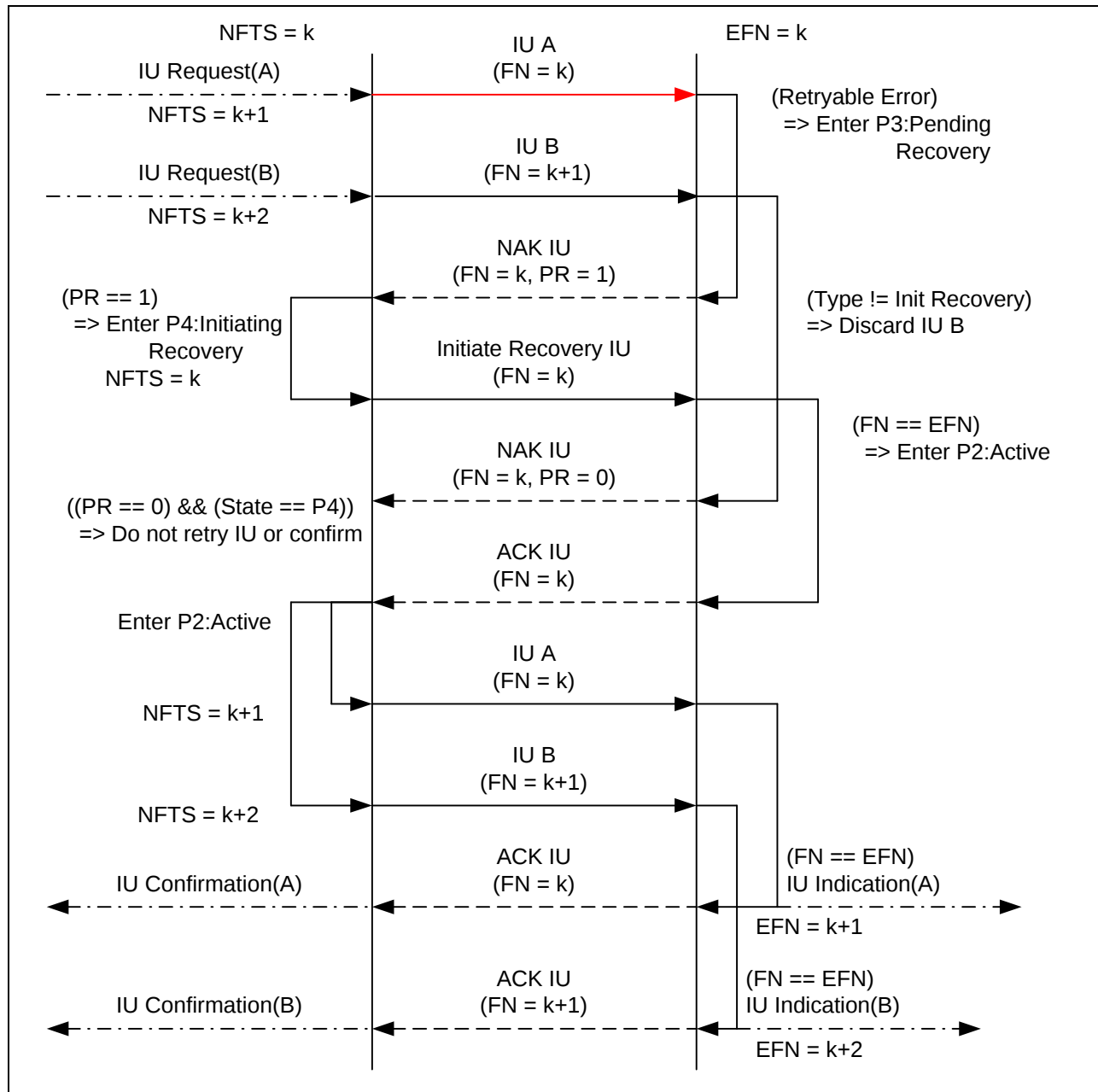
Figure A.2 – Receiver-Detected Retryable Error with Multiple Active IUs

Figure A.3 – Lost IU with No Further Traffic

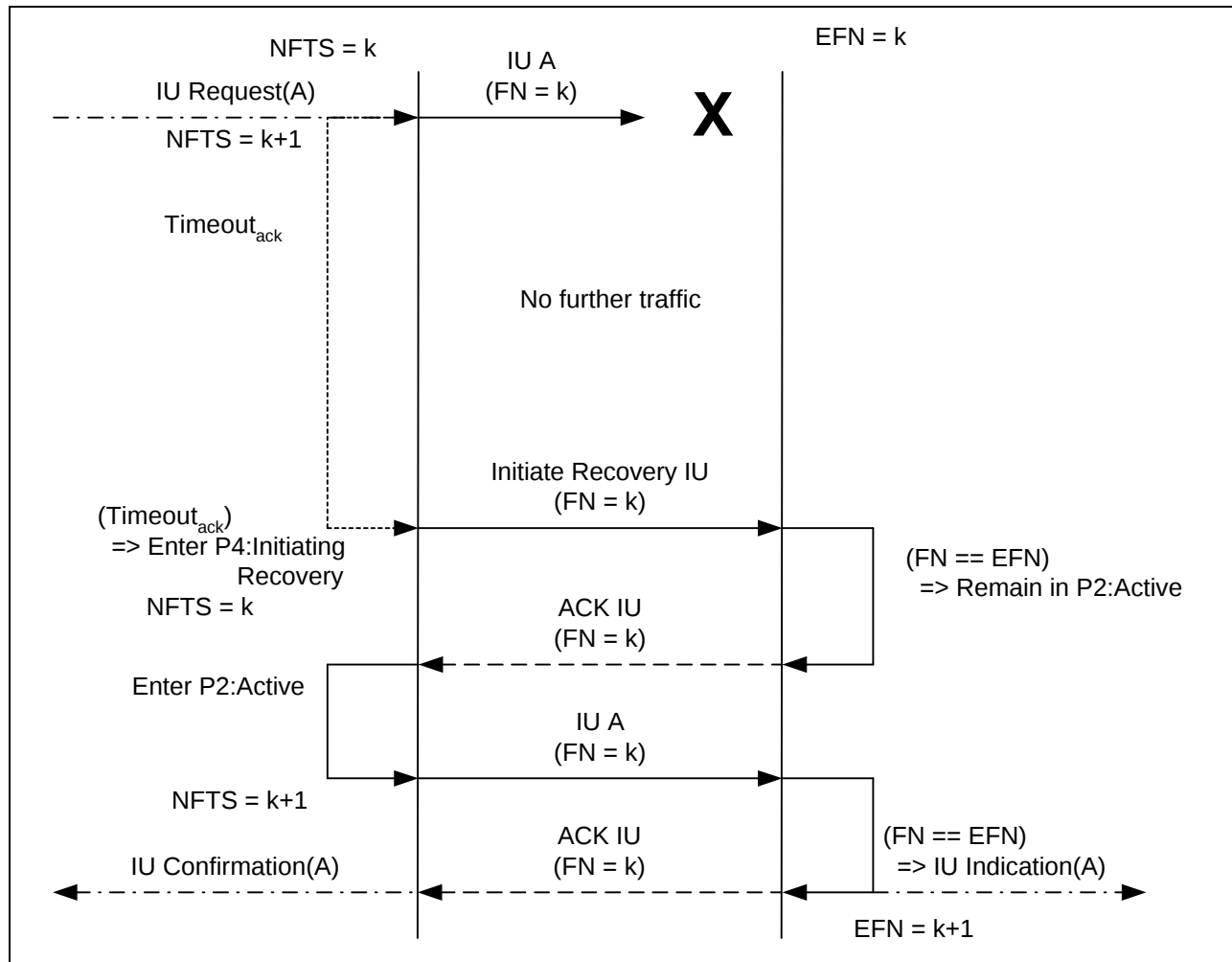


Figure A.4 – Lost ACK with Recovery Driven by Out-of-Order ACK

Note: The requirement for Acknowledgement IUs to be sent in the same order as the original IUs are received allows detection of the missing acknowledgement for FN k. There is no need to wait for timer expiration.

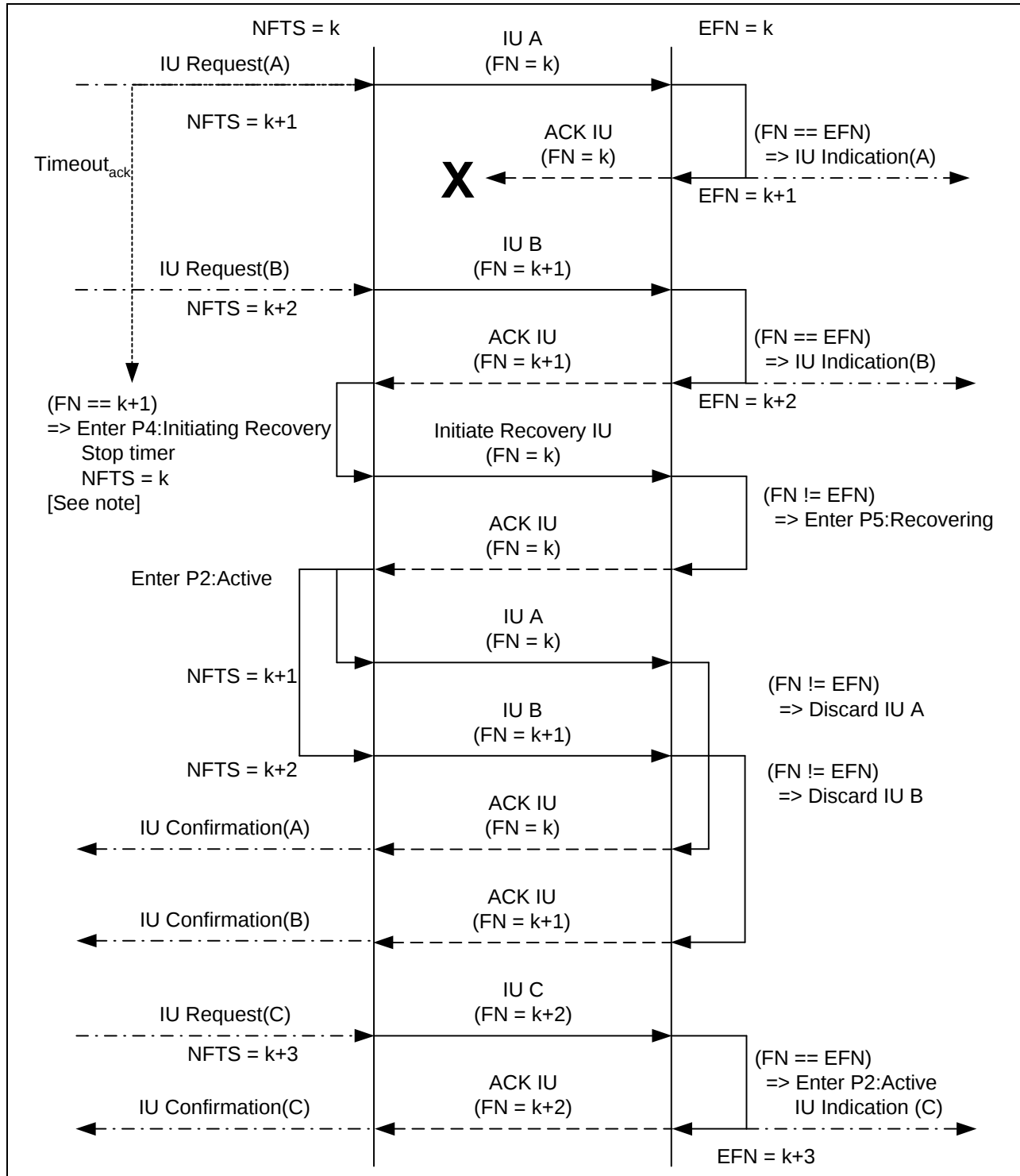


Figure A.5 – Lost IU with Recovery Driven by Out-of-Order NAK

Note: The requirement for Acknowledgement IUs to be sent in the same order as the original IUs are received allows detection of the missing acknowledgement for FN k. There is no need to wait for timer expiration.

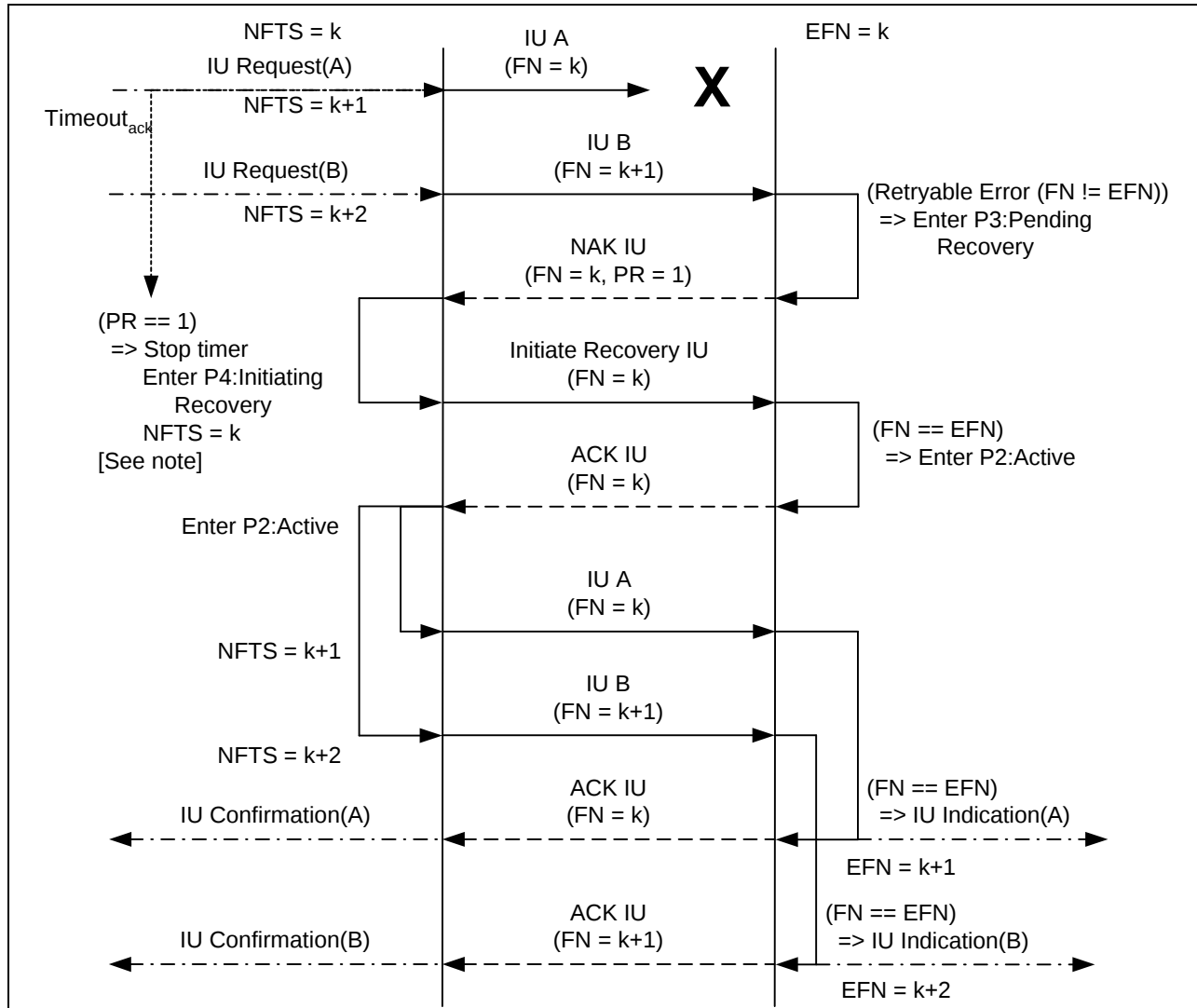


Figure A.6 – Lost NAK with Recovery Driven by Timeout

Note: In this example – unlike the previous ones – the sender does not use an out-of-order Acknowledgement IU to infer that an earlier Acknowledgement IU was lost. Instead, it waits for the $\text{Timeout}_{\text{ack}}$ on the earlier Acknowledgement IU.

This diagram would also apply similarly if IU A received an ACK instead of a non-retryable NAK.

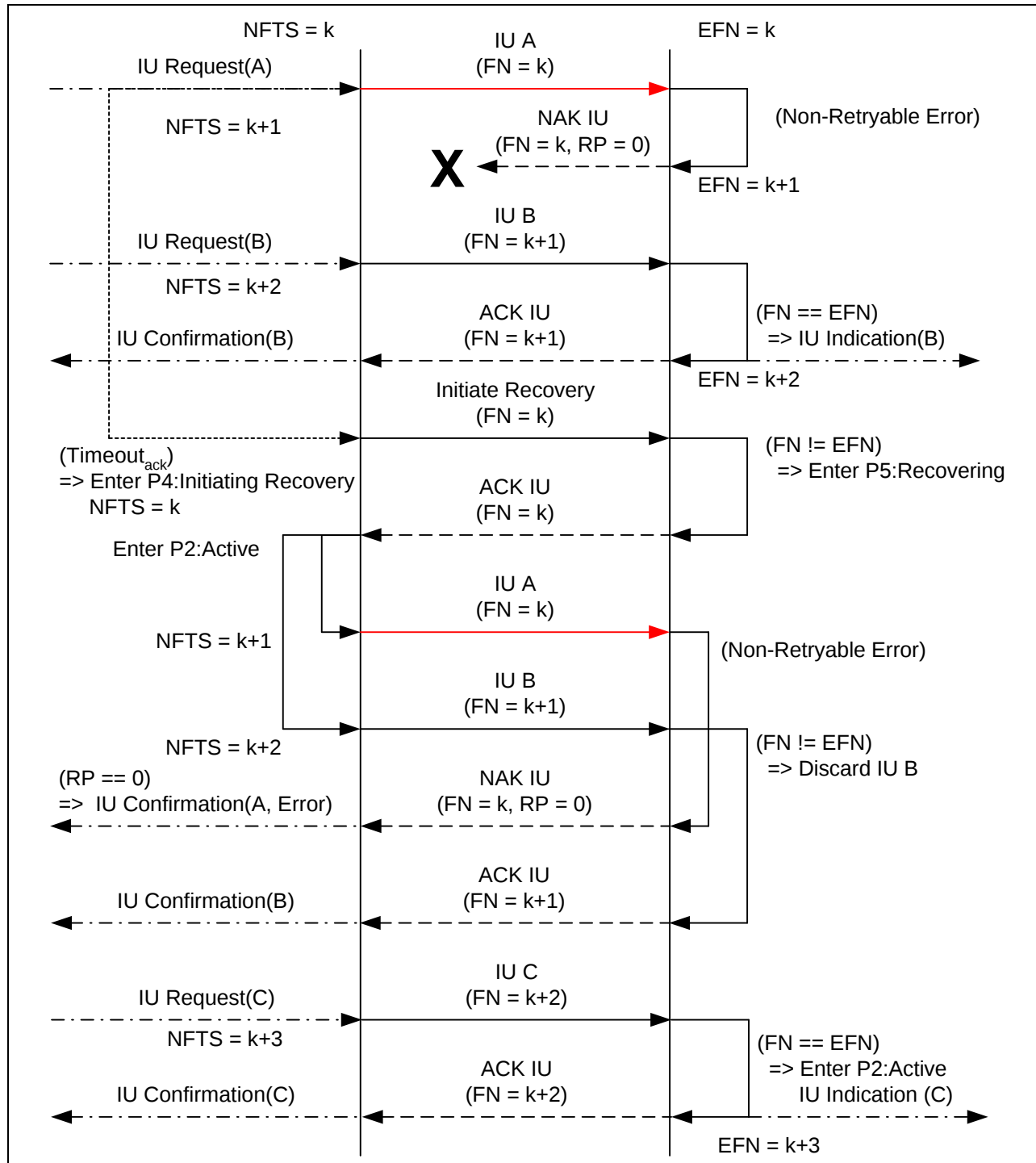


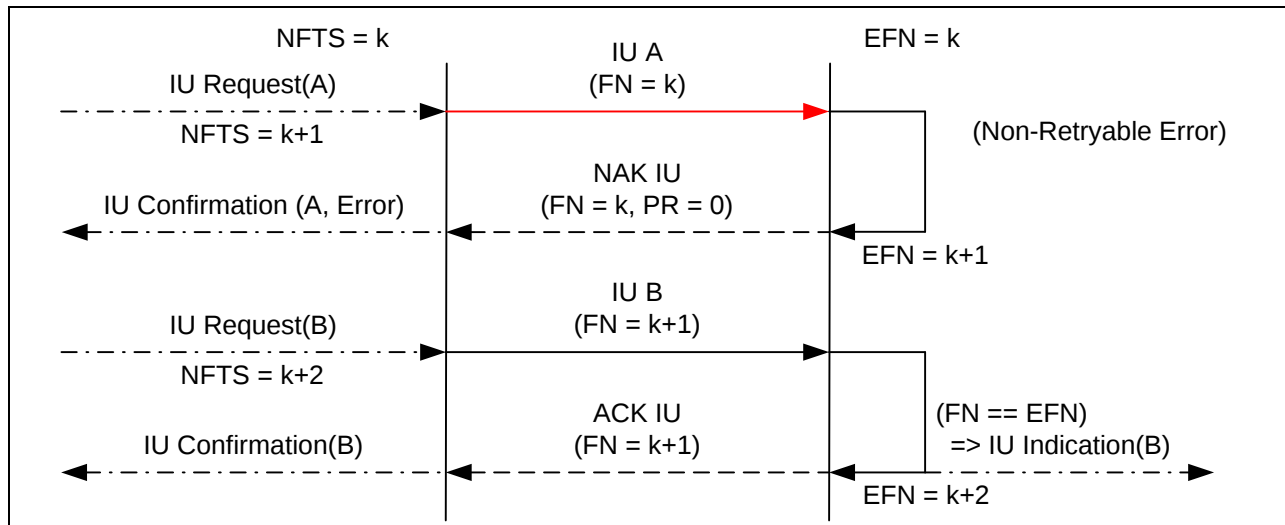
Figure A.7 – Non-Retryable Error

Figure A.8 – Lost ACK with Errors on Next IU

