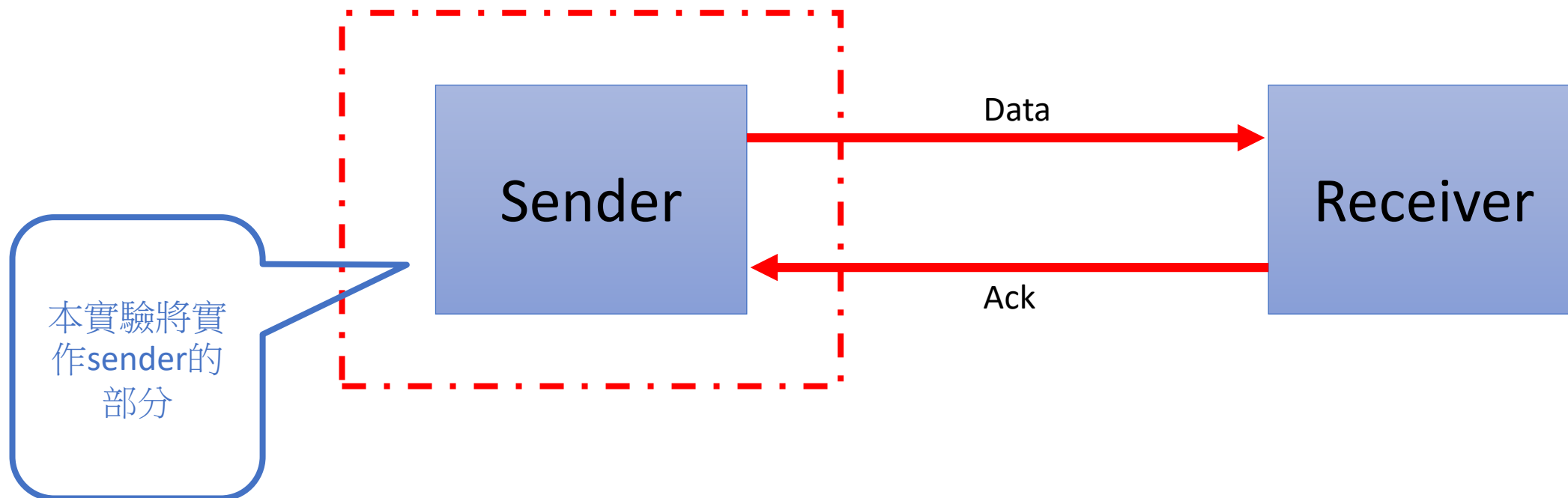


Lab 2: Go-Back-N RDT

HackMD: <https://hackmd.io/@KentShen/H1pxm9LVc>

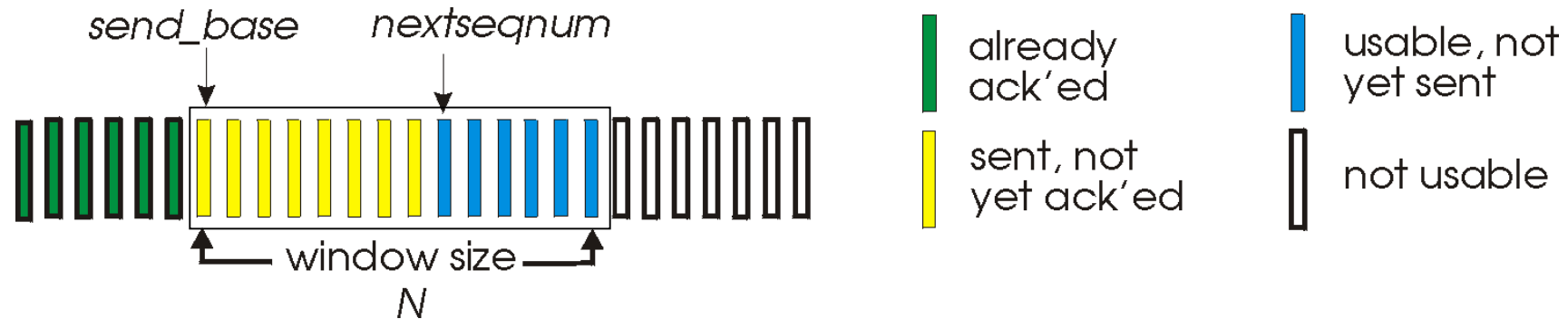
實驗場景

- 由sender端傳data到receiver，接著receiver會回應ACKs
- Receiver端運行go-back-n，並會檢查是否收到所有data
- Python 3.8.10



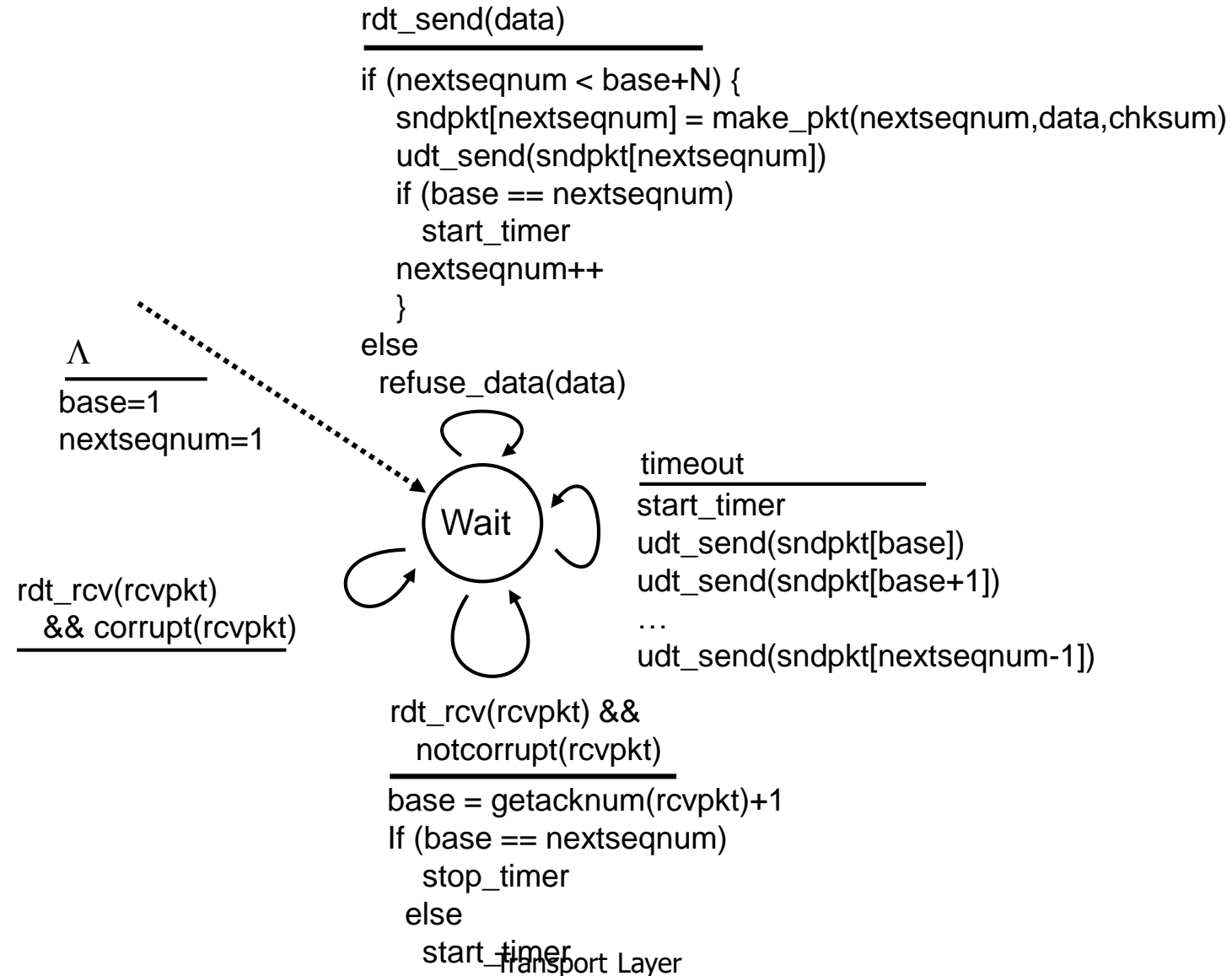
Go-Back-N: sender

- k-bit seq # in pkt header
- “window” of up to N, consecutive unack’ed pkts allowed

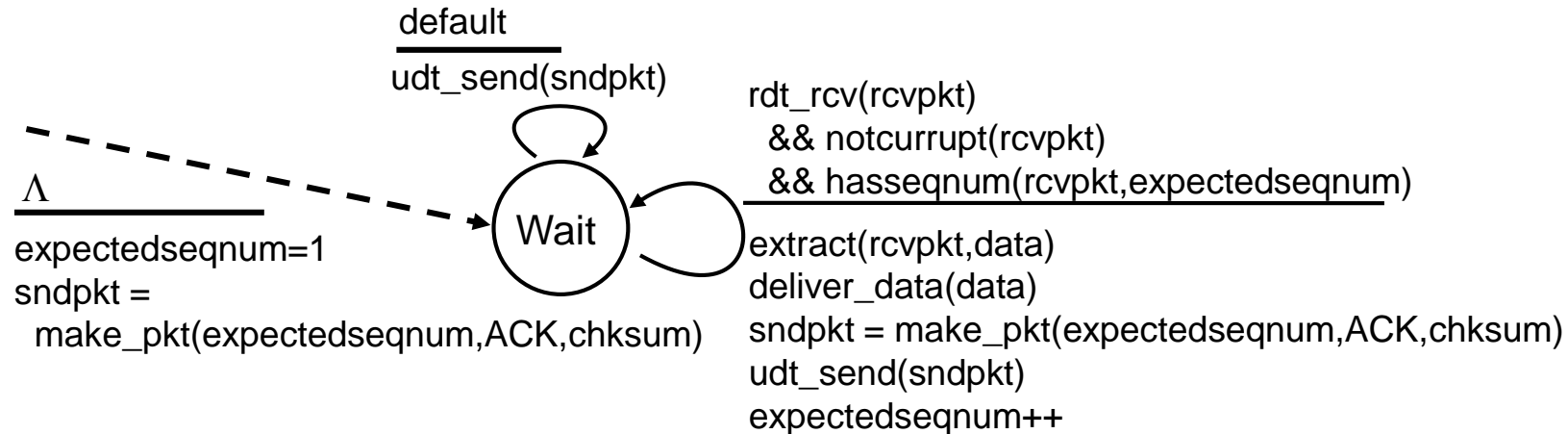


- ACK(n): ACKs all pkts up to, including seq # n -
“cumulative ACK”
 - may receive duplicate ACKs (see receiver)
- timer for oldest in-flight pkt
- *timeout(n)*: retransmit packet n and all higher seq # pkts in window

GBN: sender extended FSM



GBN: receiver extended FSM



ACK-only: always send ACK for correctly-received pkt
with highest *in-order* seq #

- may generate duplicate ACKs
- need only remember **expectedseqnum**
- out-of-order pkt:
 - discard (don't buffer): *no receiver buffering!*
 - re-ACK pkt with highest in-order seq #

Sender端設定

`send_base = 0` → `send_base`指標位置

`next_seq_num = 0` → `next_seq_num`位置

`cwnd_size = 3` → window的大小為3

`num_pkt = 10` → 總共要送出10個封包

Sender端需完成功能

- Send_base及next_seq_num要在正確的位置
- Window內的data需接連送出
- 設定timer的時間為5秒
- Timeout時需要重傳window內所有的data
- Sender結束前需送出10筆資料

成果繳交

- 繳交檔案：`sender.py`
- 繳交時間：May 5, 2022