|  |  |  |  |
| --- | --- | --- | --- |
| ***#*** | ***Line*** | ***Text*** | ***Checks for...*** |
| ***01*** | 113 | false | Client exits closed state. |
| ***02*** | 131 | false | Client exits listen state. |
| ***03*** | 138 | inack == seq | Message received has correct ack value. |
| ***04*** | 139 | msg == SYN\_ACK || msg == SYN | Message is of a valid type. |
| ***05*** | 160 | false | Client exits syn\_sent state. |
| ***06*** | 169 | inack == seq | Message received has correct ack value. |
| ***07*** | 170 | msg == RST | Message is of a valid type. |
| ***08*** | 181 | false | Client exits syn\_received state. |
| ***09*** | 205 | Inack == seq | Message received has correct ack value. |
| ***10*** | 206 | msg == ACK | Message is of a valid type. |
| ***11*** | 218 | false | Client exits established state. |
| ***12*** | 225 | inack == seq | Message received has correct ack value. |
| ***13*** | 226 | msg == ACK || msg == FIN | Message is of a valid type. |
| ***14*** | 248 | false | Client exits fin\_wait\_1 state. |
| ***15*** | 257 | inack == seq | Message received has correct ack value. |
| ***16*** | 258 | msg == FIN | Message is of a valid type. |
| ***17*** | 263 | false | Client exits fin\_wait\_2 state. |
| ***18*** | 268 | false | Client exits close\_wait state. |
| ***19*** | 277 | inack == seq | Message received has correct ack value. |
| ***20*** | 278 | msg == ACK | Message is of a valid type. |
| ***21*** | 281 | false | Client exits closing state. |
| ***22*** | 286 | false | Client exits last\_ack state. |
| ***23*** | 298 | false | Client exits time\_wait state. |
| ***24*** | 340 | false | Server exits closed state. |
| ***25*** | 349 | msg == SYN | Message is of a valid type. |
| ***26*** | 356 | false | Server exits listen state. |
| ***27*** | 365 | msg == SYN | Message is of a valid type. |
| ***28*** | 372 | false | Server exits syn\_sent state. |
| ***29*** | 381 | msg == ACK | Message is of a valid type. |
| ***30*** | 382 | inack == seq | Message received has correct ack value. |
| ***31*** | 386 | false | Server exits syn\_received state. |
| ***32*** | 393 | msg == FIN || msg == DATA | Message is of a valid type. |
| ***33*** | 394 | inack == SEQ | Message received has correct ack value. |
| ***34*** | 416 | false | Server exits established state. |
| ***35*** | 421 | false | Server exits fin\_wait\_1 state. |
| ***36*** | 426 | false | Server exits fin\_wait\_2 state. |
| ***37*** | 439 | false | Server exits close\_wait state. |
| ***38*** | 444 | false | Server exits closing state. |
| ***39*** | 453 | msg == ACK | Message is of a valid type. |
| ***40*** | 454 | inack == seq | Message received has correct ack value. |
| ***41*** | 458 | false | Server exits last\_ack state. |
| ***42*** | 463 | false | Server exits time\_wait state. |

|  |  |  |
| --- | --- | --- |
| ***#*** | ***Never...*** | ***Tests For...*** |
| ***01*** | c\_close\_wait | Client never enters close\_wait state. |
| ***02*** | c\_last\_ack | Client never enters last\_ack state. |
| ***03*** | s\_fin\_wait\_1 | Server never enters fin\_wait\_1 state. |
| ***04*** | s\_fin\_wait\_2 | Server never enters fin\_wait\_2 state. |
| ***05*** | s\_closing | Server never enters closing state. |
| ***06*** | s\_time\_wait | Server never enters time\_wait state. |
| ***07*** | c\_exit && <>!c\_exit | Client never leaves exit state. |
| ***08*** | s\_exit && <>!s\_exit | Server never leaves exit state. |
| ***09*** | c\_closed && !(c\_closed U (c\_syn\_sent || c\_exit)) | Client transitions from closed to syn\_sent or exit state. |
| ***10*** | c\_listen && !(c\_listen U (c\_closed || c\_syn\_sent)) | Client transitions from listen to closed or syn\_sent state. |
| ***11*** | c\_syn\_sent && !(c\_syn\_sent U (c\_established || c\_syn\_received)) | Client transitions from syn\_sent to established or syn\_received state. |
| ***12*** | c\_established && !(c\_established U (c\_fin\_wait\_1)) | Client transitions from established to fin\_wait\_1 state. |
| ***13*** | c\_fin\_wait\_1 && !(c\_fin\_wait\_1 U (c\_fin\_wait\_2 || c\_closing || c\_time\_wait)) | Client transitions from fin\_wait\_1 to fin\_wait\_2 or closing or time\_wait state. |
| ***14*** | c\_fin\_wait\_2 && !(c\_fin\_wait\_2 U (c\_time\_wait)) | Client transitions from fin\_wait\_2 to time\_wait state. |
| ***15*** | c\_closing && !(c\_closing U c\_time\_wait) | Client transitions from closing to time\_wait state. |
| ***16*** | c\_time\_wait && !(c\_time\_wait U c\_closed) | Client transitions from time\_wait to closed state. |
| ***17*** | s\_closed && !(s\_closed U (s\_listen || s\_syn\_sent || s\_exit)) | Server transitions from closed to listen or syn\_sent or exit state. |
| ***18*** | s\_listen && !(s\_listen U s\_syn\_received) | Server transitions from listen to syn\_received state. |
| ***19*** | s\_syn\_sent && !(s\_syn\_sent U s\_established) | Server transitions from syn\_sent to established state. |
| ***20*** | s\_syn\_received && !(s\_syn\_received U s\_established) | Server transitions from syn\_recieved to established state. |
| ***21*** | s\_established && !(s\_established U s\_close\_wait) | Server transitions from established to close\_wait state. |
| ***22*** | s\_close\_wait && !(s\_close\_wait U s\_last\_ack) | Server transitions from close\_wait to last\_ack state. |
| ***23*** | s\_last\_ack && !(s\_last\_ack U s\_closed) | Server transitions from last\_ack to closed state. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Line | Code should... | Assertion | Caught |
| 01 | 467 | Let process end | LTL\_08 | yes |
| 02 | 384 | Use a goto to switch states. | IMM\_31 | yes |
| 03 | 350 | Increment ack. | IMM\_03 | yes |
| 04 | 193 | Go to fin\_wait\_1 state. | LTL\_12 | yes |
| 05 | 198 | Send DATA instead of SYN. | IMM\_32 | yes |
| 06 | 193 | Client should never enter close\_wait state. | LTL\_01 | yes |
| 07 | 437 | Server should never enter closing state. | LTL\_05 | yes |
| 08 | 106 | Client should send SYN instead of ACK. | IMM\_25 | yes |
| 09 | 134 | Client state should be SYN\_SENT, never LAST\_ACK | LTL\_02 | yes |