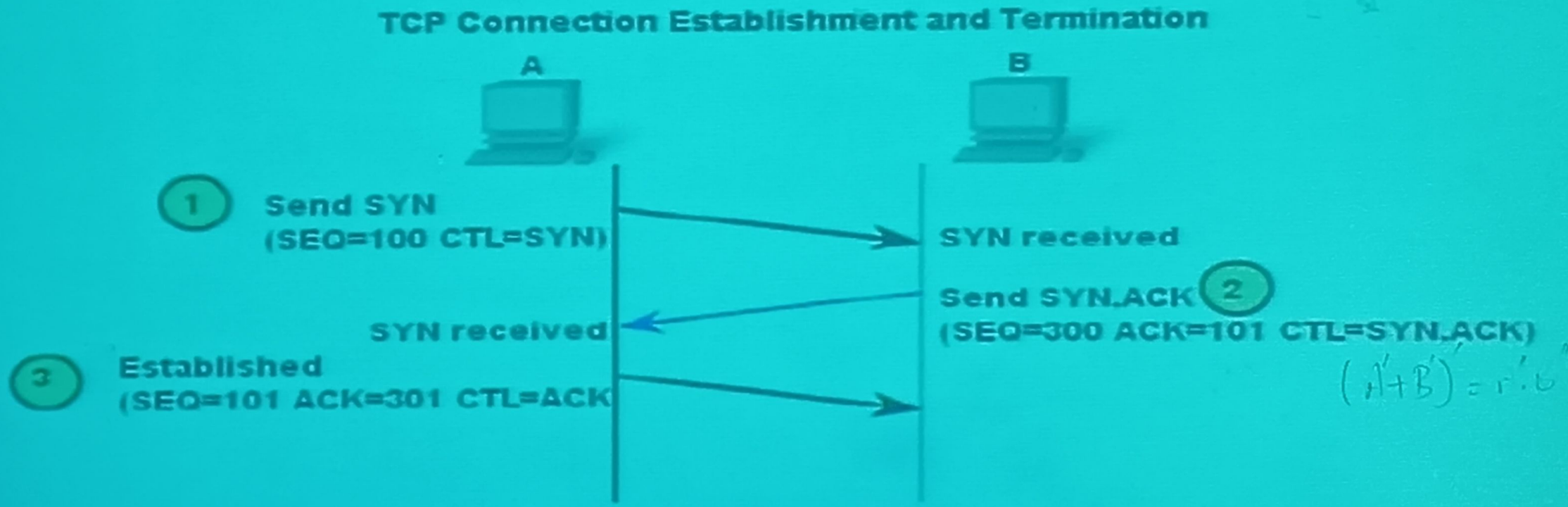
Connection Management

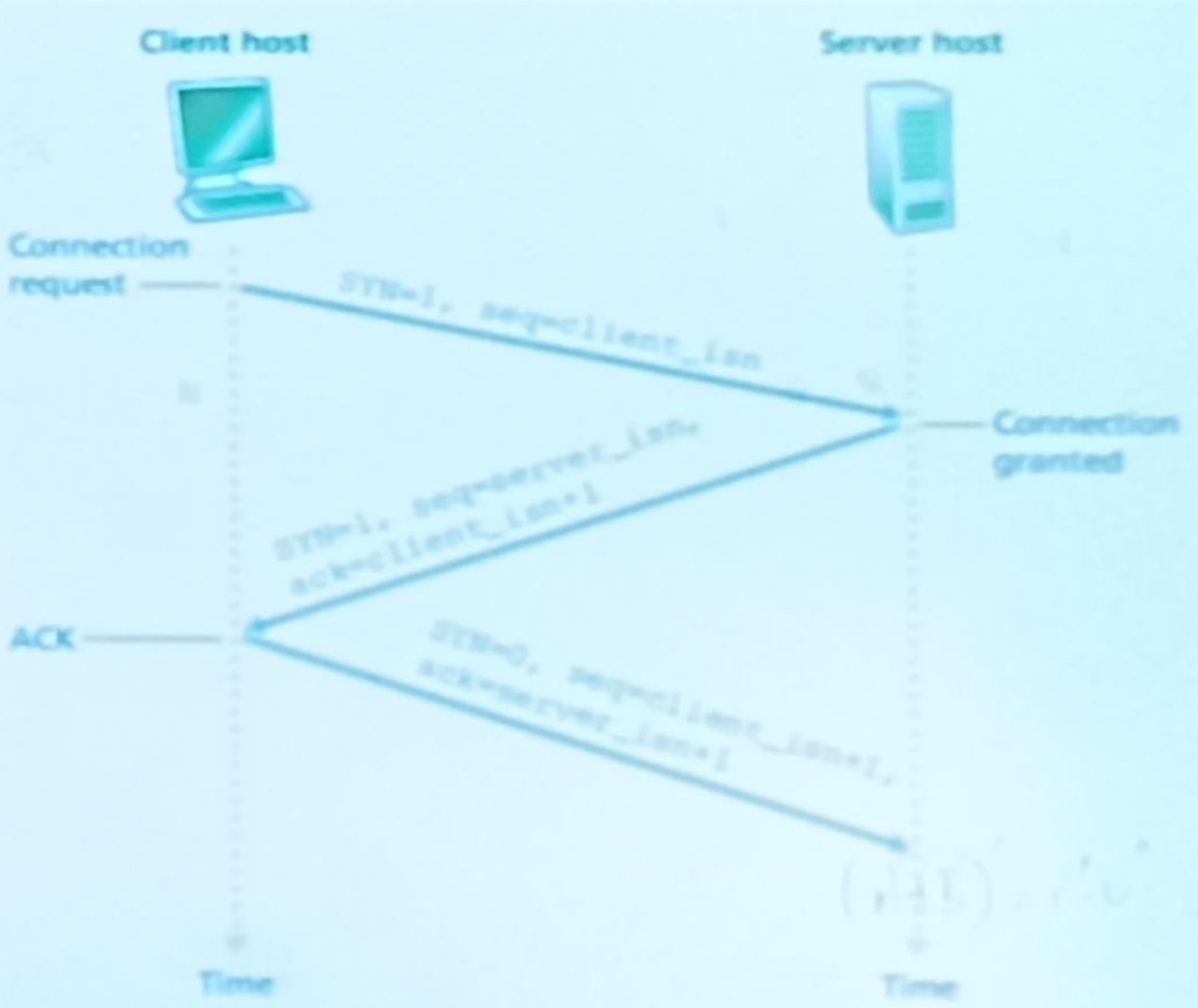
TCP communication works in Server/Client model. The client initiates the connection and the server either accepts or rejects it. Three-way handshaking is used for connection management.

Three Ways Handshake



TCP Connection Management: Establishing a new Connection

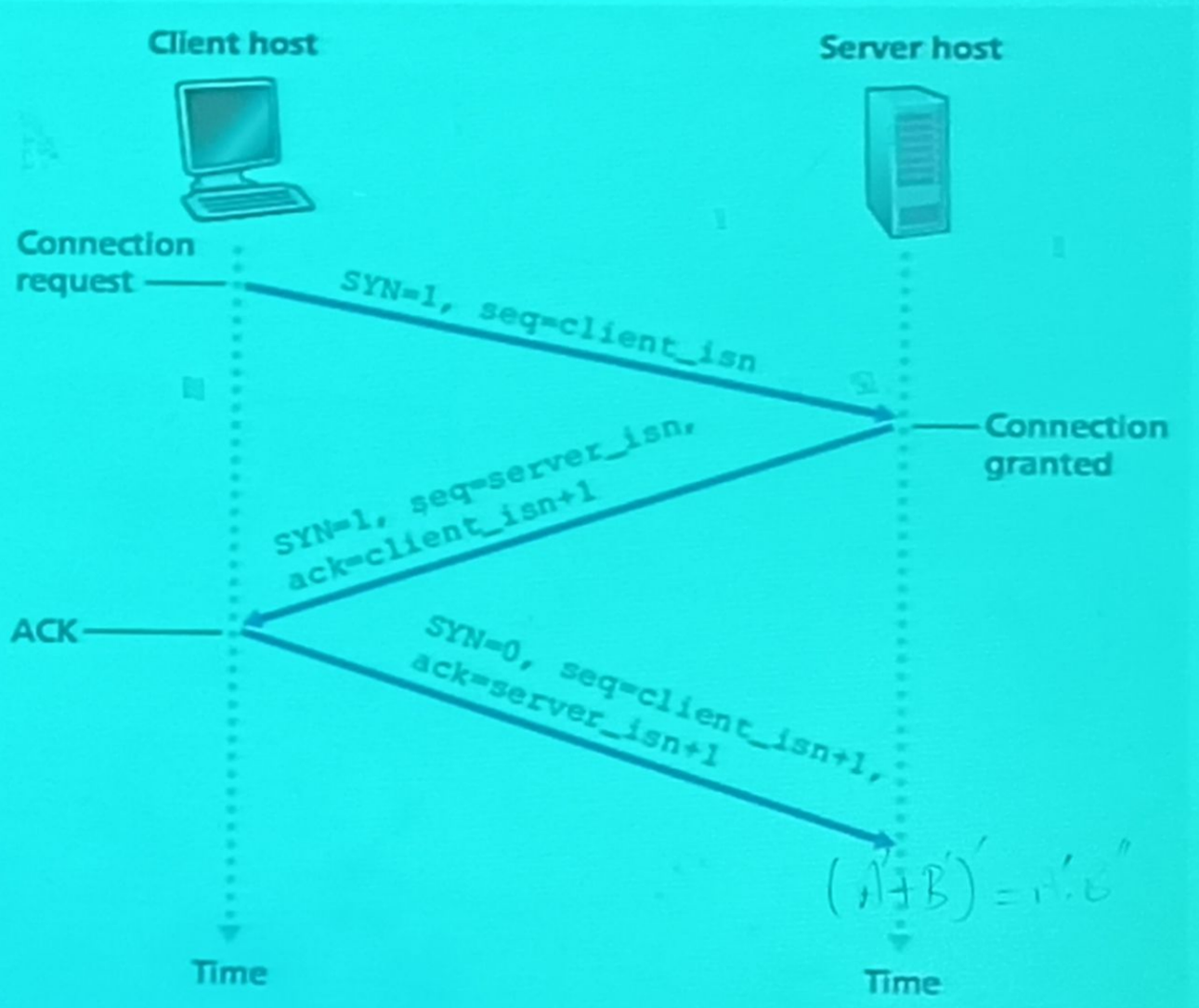
- ***Step 1.*** The client-side TCP first sends a special TCP segment to the server-side TCP. This special segment contains no application-layer data. But one of the flag bit (***SYN bit***) in the segment's header is set to 1. For this reason, this special segment is referred to as a TCP SYN segment.
- In addition, the client randomly chooses an initial sequence number (***client_isn***) and puts this number in the sequence number field of the initial TCP SYN segment.
- This segment is encapsulated within an IP datagram and sent to the server.



• TCP three-way handshake: segment exchange

TCP Connection Management: Establishing a new Connection

- ***Step 1.*** The client-side TCP first sends a special TCP segment to the server-side TCP. This special segment contains no application-layer data. But one of the flag bit (***SYN bit***) in the segment's header is set to 1. For this reason, this special segment is referred to as a TCP SYN segment.
- In addition, the client randomly chooses an initial sequence number (***client_isn***) and puts this number in the sequence number field of the initial TCP SYN segment.
- This segment is encapsulated within an IP datagram and sent to the server.



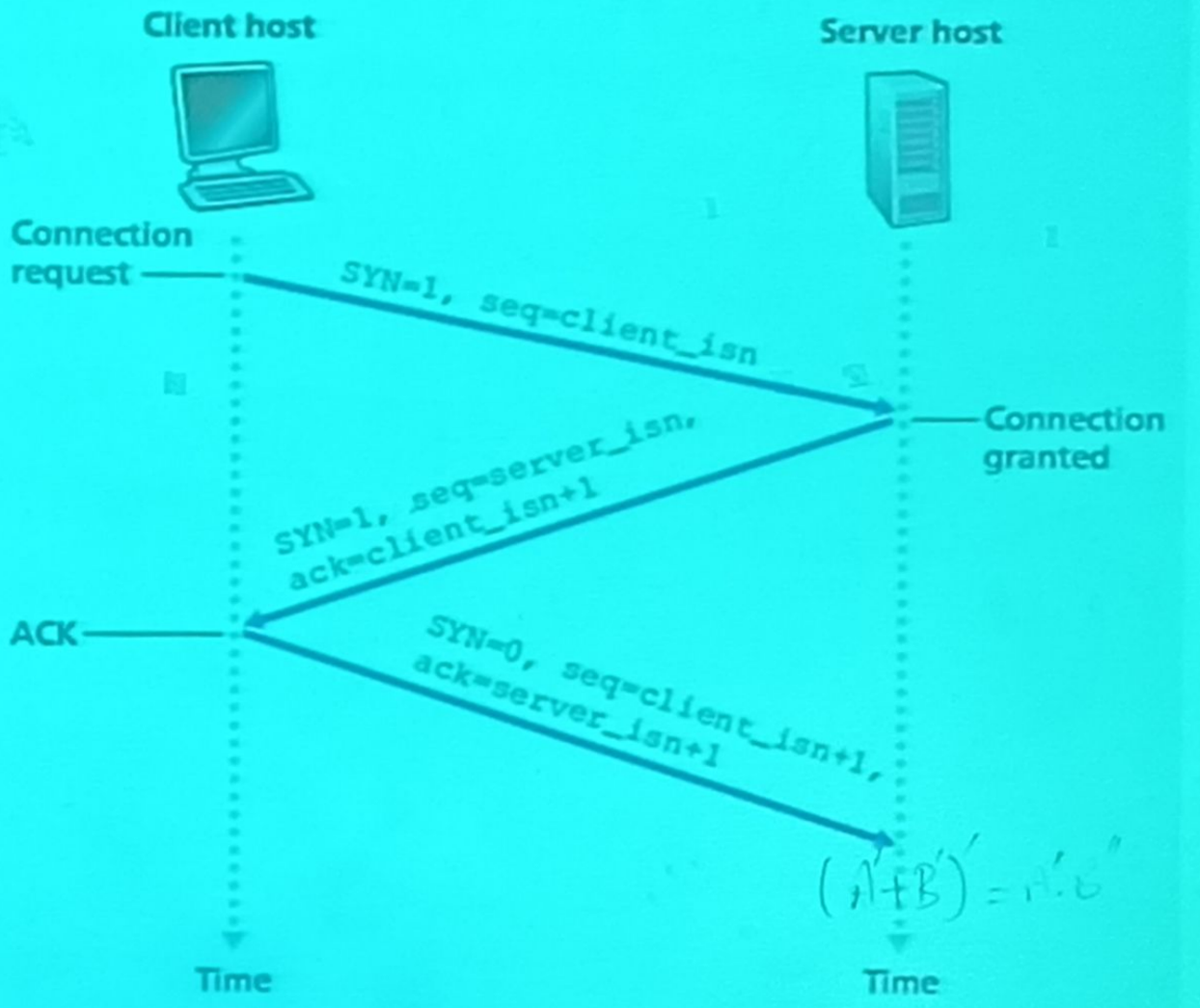
• TCP three-way handshake: segment exchange

TCP Connection Management: Establishing a new Connection

- **First**, the **SYN** bit in segment's header is set to **1**.
- **Second**, the acknowledgment number of the segment's header is set to ***client_isn+1***.
- **Finally**, the server chooses its own initial sequence number (***server_isn***) and puts this value in the sequence number field of the segment header.

- This connection-granted segment is saying, in effect, "I received your SYN packet to start a connection with your initial sequence number, **client_isn**. I agree to establish this connection. My own initial sequence number is **server_isn**."

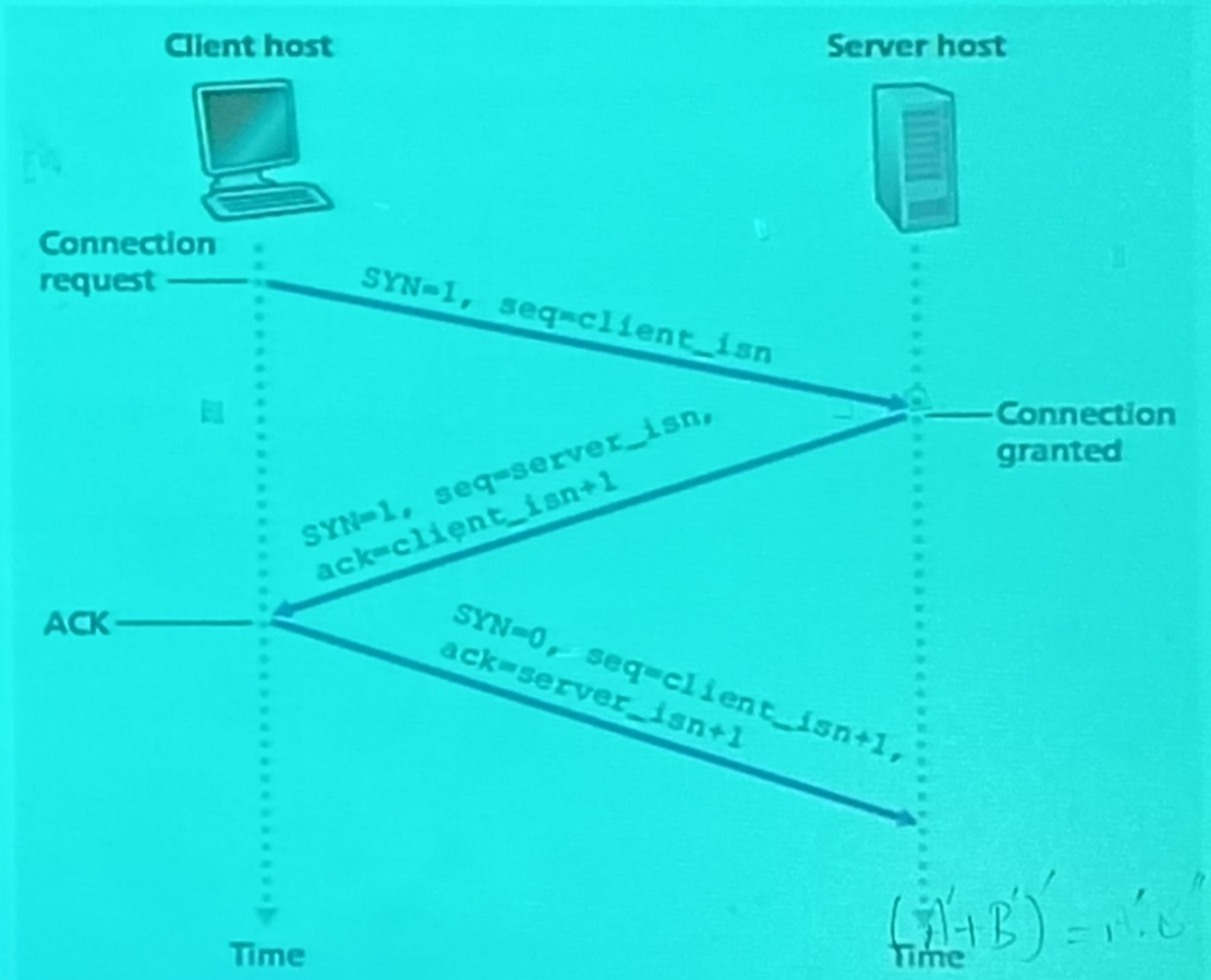
- The connection-granted segment is referred to as a **SYNACK segment.**



• TCP three-way handshake: segment exchange

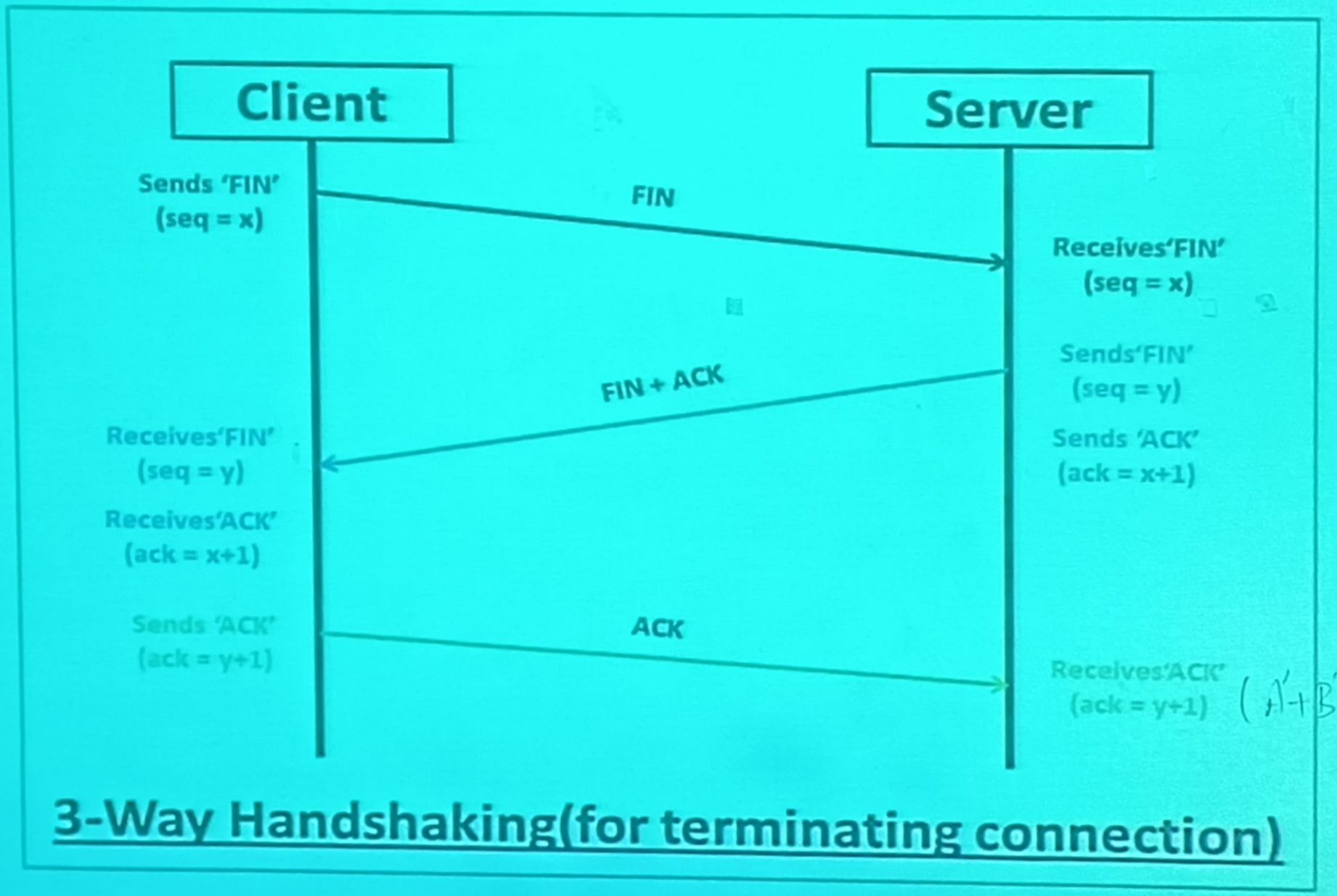
TCP Connection Management: Establishing a new Connection

- **Step 3.** Upon receiving the SYNACK segment, the client also allocates buffers and variables to the connection. The client host then sends the server yet another segment; this last segment acknowledges the server's connection-granted segment. The client does so by putting the value ***server_isn+1*** in the acknowledgment field of the TCP segment header.
- The SYN bit is set to zero, since the connection is established. This third stage of the three-way handshake may carry client-to server data in the segment payload.



• TCP three-way handshake: segment exchange

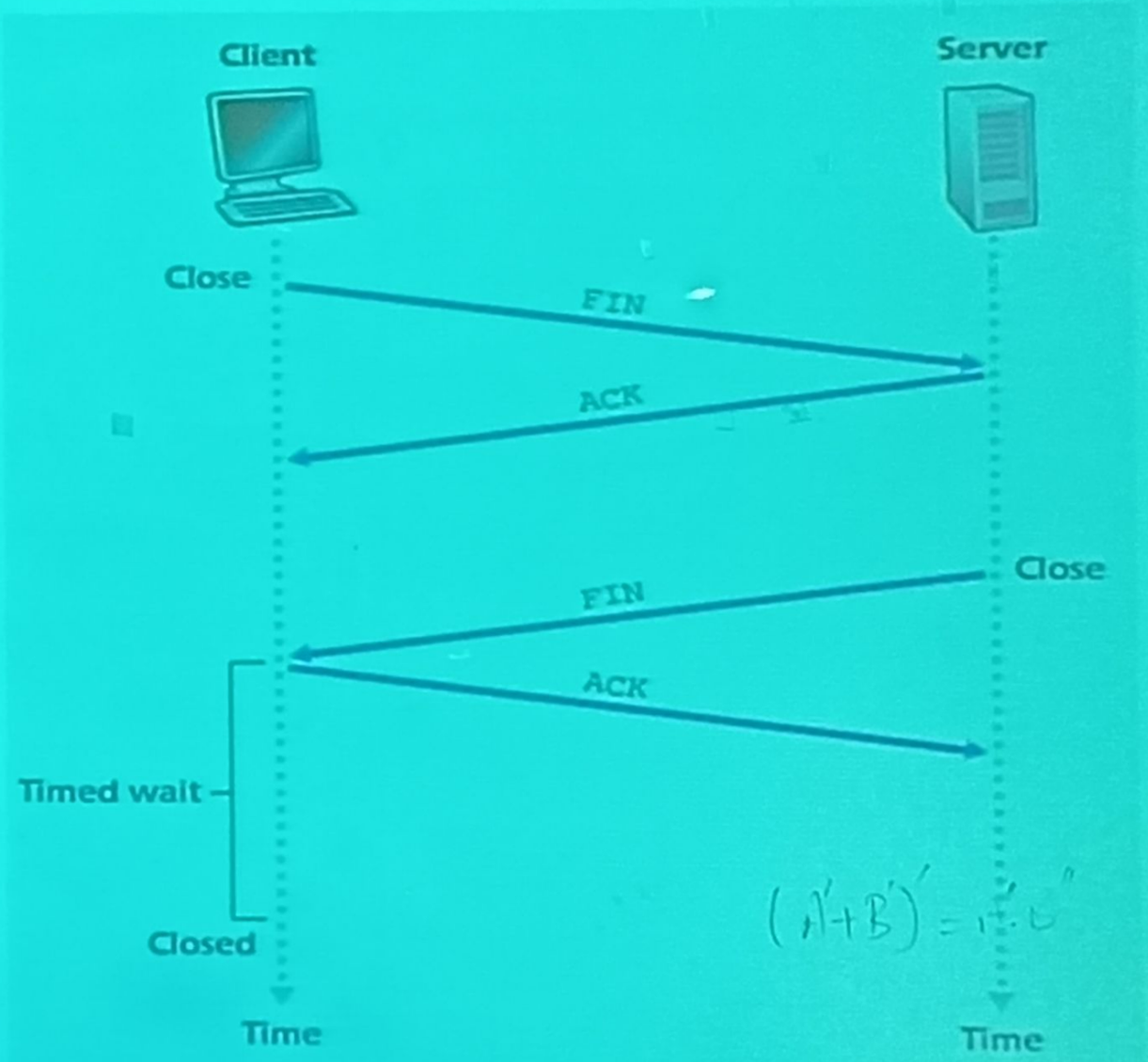
TCP Connection Management: Connection Termination



3-Way Handshaking(for terminating connection)

TCP Connection Management: Connection Termination

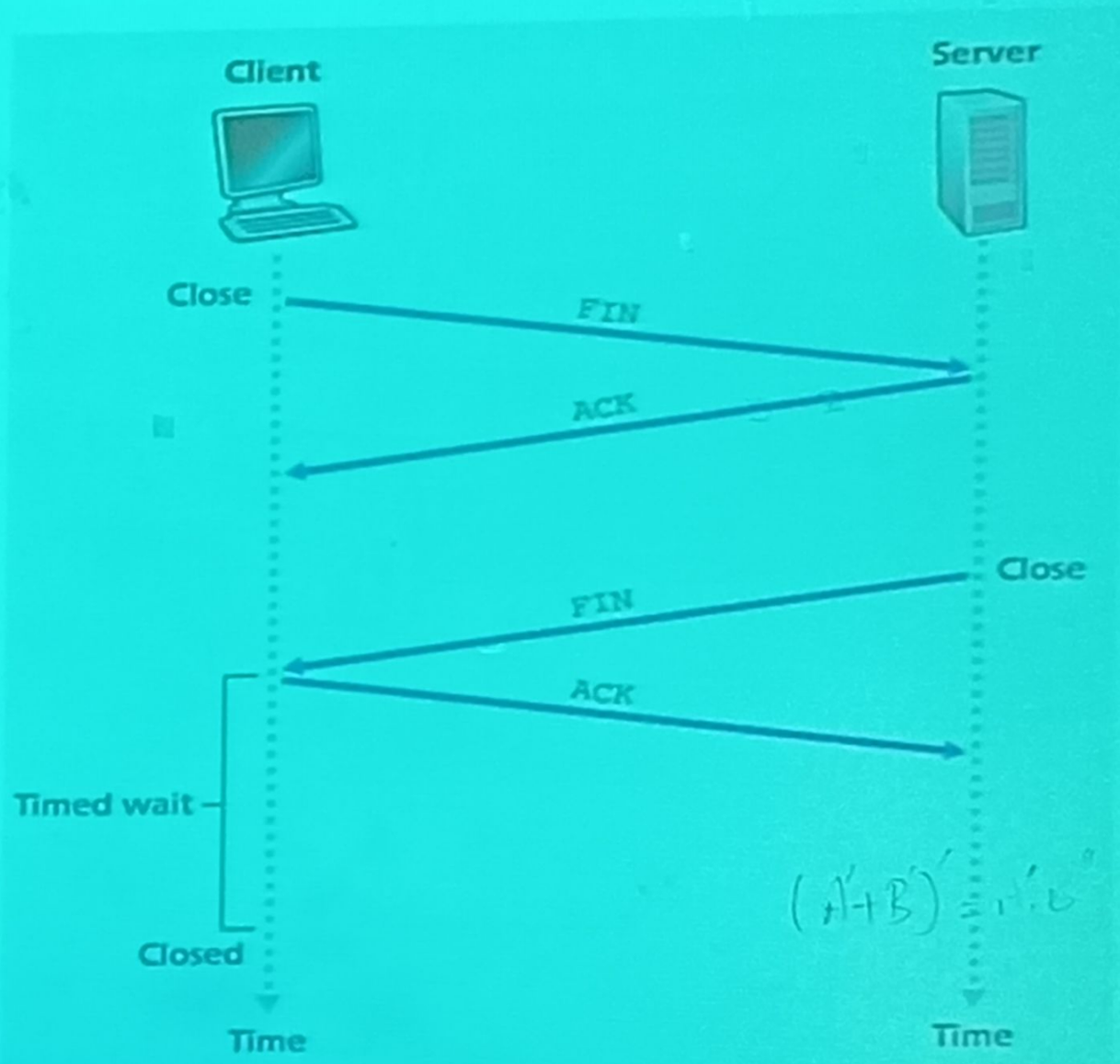
- Either of the two processes participating in a TCP connection can end the connection.
- When a connection ends, the "resources" (that is, the buffers and variables) in the hosts are deallocated.
- As an example, suppose the client decides to close the connection, as shown in Figure. The client application process issues a close command. **This causes the client TCP to send a special TCP segment to the server process.** This special segment has FIN flag bit set to 1 in segment's header.



♦ Closing a TCP connection

TCP Connection Management: Connection Termination

- When the server receives this segment, it sends the client an acknowledgment segment in return.
- The server then sends its own shutdown segment, which has the FIN flag bit set to 1. Finally, the client acknowledges the server's shutdown segment. At this point, all the resources in the two hosts are now deallocated.



♦ Closing a TCP connection