# TRANSPORT LAYER

## RESPONSIBILITIES / SERVICES

1. Process to Process Delivery (Port to Port)
2. Flow Control
3. Error Control
4. Congestion Control
5. Connection Control
6. Segmentation and Reassembly

## ADDRESSING

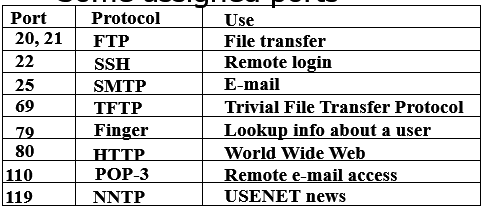
### PORT NO

To distinguish between multiple processes on a device.

**16** BIT { **0 to 65535** }

**TYPES OF PORTS**

* **0 – 1023** : Well-Defined
* **1024 – 49151** : Reserved by Microsoft, Oracle
* **49152 – 65535** : Locally available



### SOCKET ADDRESS

Socket Address = IP Address + Port Number

Example: 192.168.1.10:8080

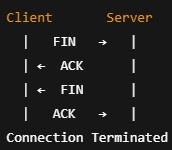
## CONNECTION ESTABLISHMENT & RELEASE

### Connection Establishment (3-Way Handshake)

* **Step 1: SYN**
* **Step 2: SYN-ACK**
* **Step 3: ACK**

### Connection Release (4-Way Handshake)

* **Step 1**: Client sends **FIN** (Finish) to server.
* **Step 2**: Server acknowledges **ACK**.
* **Step 3**: Server sends its own **FIN**.
* **Step 4**: Client acknowledges **ACK**.

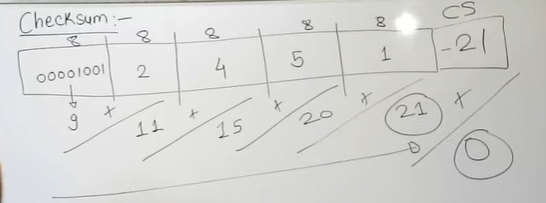


## ERROR CONTROL

 **Purpose**: Ensure reliable data transmission.

 **Methods**:

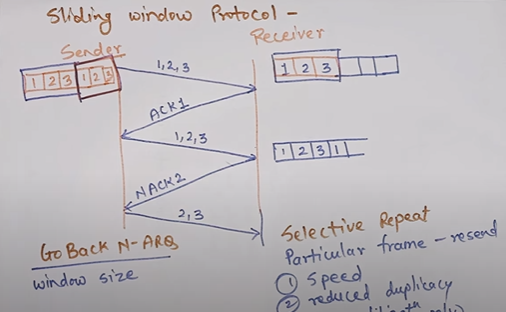
* **Checksum**: Detects errors.



* **Acknowledgment (ACK)**: Confirms successful receipt.
* **Negative Acknowledgment (NAK/NACK)**: Requests retransmission if error detected.
* **Retransmission Timers**: Resend segments not acknowledged in time.

 **Automatic Repeat Request (ARQ)** protocols:

* Stop-and-Wait ARQ
* Go-Back-N ARQ
* Selective Repeat ARQ



## FLOW CONTROL

 **Purpose**: Prevent overwhelming the receiver.

 **Technique**: **Sliding Window Protocol** (TCP uses it).

 **Mechanism**:

* Sender maintains a window of frames it can send without receiving an acknowledgment.
* Receiver advertises window size based on its buffer capacity (called **advertised window**).

## ATM LAYERS

Asynchronous Transfer Mode

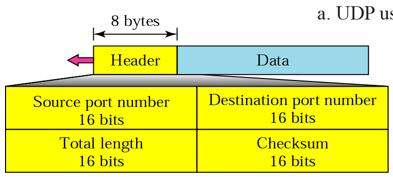
ATM is a high-speed, cell-based switching technique.

## UDP : USER DATAGRAM PROTOCOL ✅

Connectionless, unreliable protocol that has   
no flow and error control,   
no congestion control,   
no retransmission upon receipt of a bad segment.

### UDP HEADER

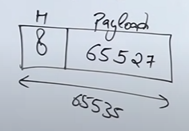
UDP packets, called user datagrams, have a fixed-size header of 8 bytes.



**TOTAL LENGTH -**

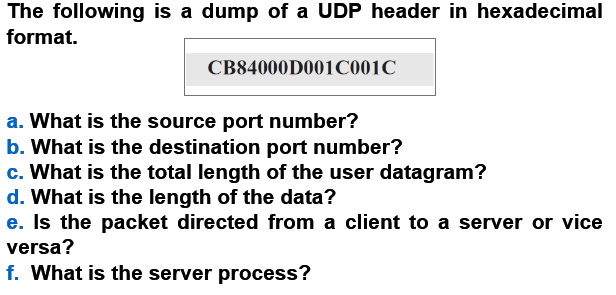
16 => 216 = 65535 {0 to 65534}

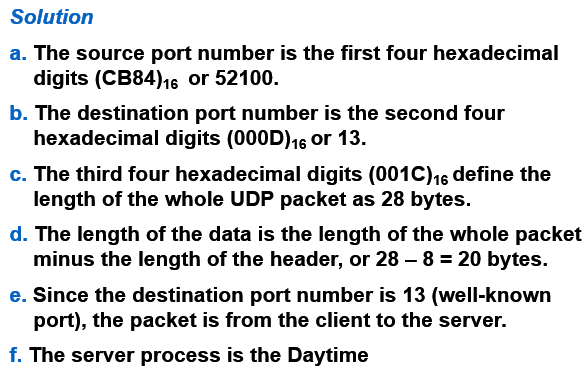
If header length fix = 8 => Payload = 65535-8



**CHECK SUM(optional)**

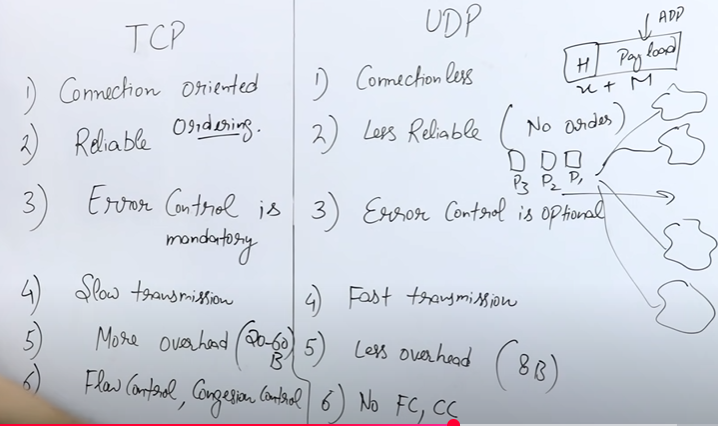
Calculated using   
UDP header + UDP data + Pseudo Header  
If Hash values == for both then no error





### UDP APPLICATIONS

1. Query response protocol
2. Speed
3. Broadcast / Multicast
4. Streaming



## TCP

