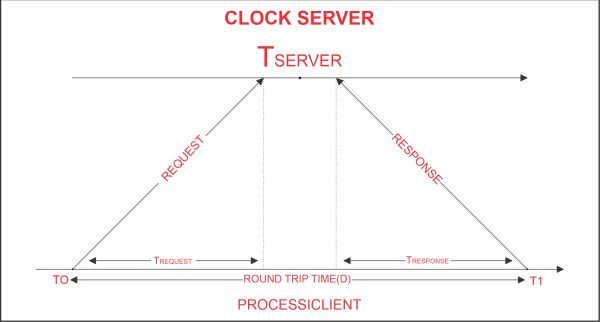
**Aim:** Implement Cristian Algorithm for clock synchronization in distributed systems

**Theory:**

Client processes synchronise time with a time server using Cristian's Algorithm, a clock synchronisation algorithm. While redundancy-prone distributed systems and applications do not work well with this algorithm, low-latency networks where round trip time is short relative to accuracy do. The time interval between the beginning of a Request and the conclusion of the corresponding Response is referred to as Round Trip Time in this context.

An example mimicking the operation of Cristian's algorithm is provided below:



Algorithm:

1. The process on the client machine sends the clock server a request at time T 0 for the clock time (time at the server).
2. In response to the client process's request, the clock server listens and responds with clock server time.
3. The client process retrieves the response from the clock server at time T1 and uses the formula below to determine the synchronised client clock time.

**TCLIENT = TSERVER plus (T1 - T0)/2.**

where TCLIENT denotes the synchronised clock time, TSERVER denotes the clock time returned by the server, T0 denotes the time at which the client process sent the request, and T1 denotes the time at which the client process received the response