

# Real Time Group



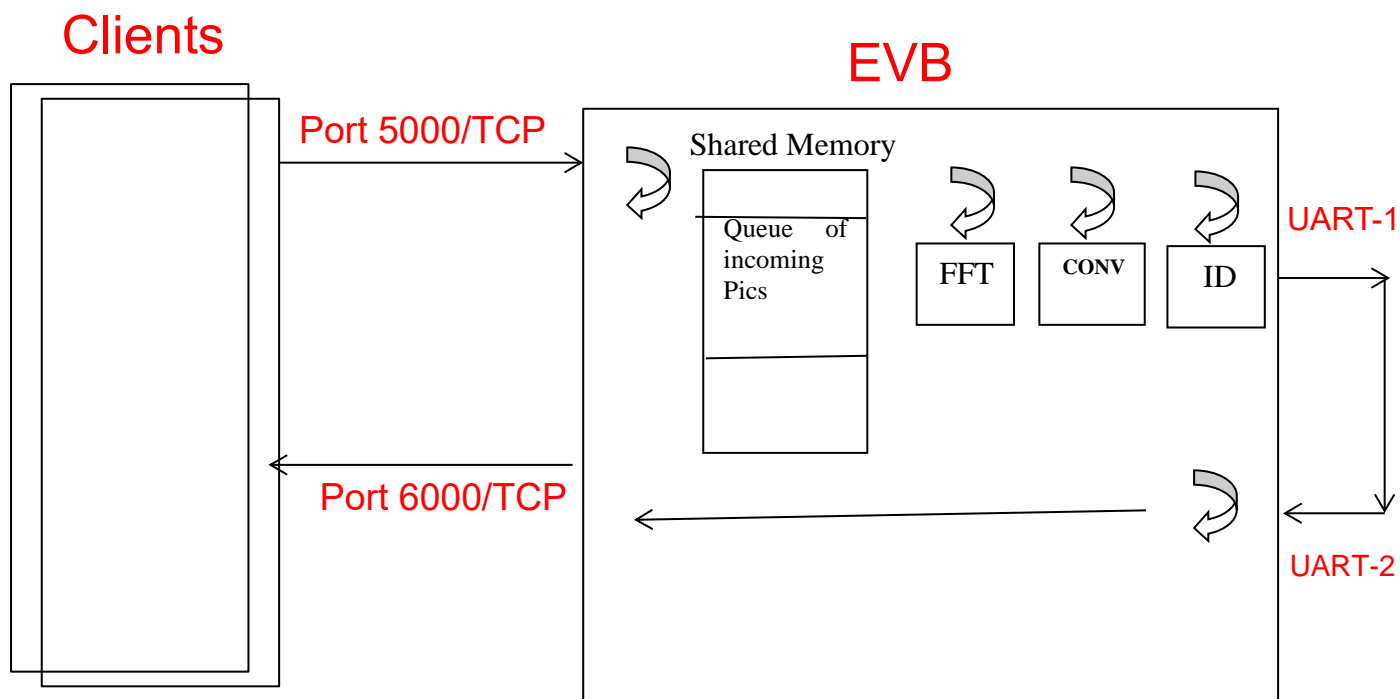
## RT Embedded Linux Solutions

### Embedded Linux Final Project

#### Project Overview

This project's goal is to implement an inference (identifying picture objects) by software and demonstrate the new approach for processing requests on EVB.

1. Clients (P.C.s or Mobile phones) will send pics (as 1Kbyte packets) to the EVB using TCP port 5000.
2. A process will receive all incoming packets and enqueue them in Shared Memory.
3. Each pic will go through a pipeline made of the following processes:
  - a. **FFT** (Fast Furrier Transform)  
The process will include 1 thread.  
Please multiply each byte in the payload by 2 instead of a true FFT Implementation.
  - b. **Convolution**  
The process will include 1 thread.  
Please add 1 to each byte in the payload instead of a true Convolution Implementation.
  - c. **Identification** – Identifying the object as Cat \ Mouse or Dog.  
The process will include 1 thread.  
Please use the Random function between 1-3:
    - i. 1 for identifying a Cat
    - ii. 2 for identifying a Mouse
    - iii. 3 for Dog).
4. The identification ( 1 or 2 or 3 ) is then sent through UART1 to UART2.
5. UART2 will receive the identification and send it back to the Client using TCP port 6000.



# Real Time Group

## RT Embedded Linux Solutions



6. The client will receive the identification and print it on the terminal.
7. Please choose Shared Memory size LONG enough for 50 Pics running concurrently.
8. Pipe line means that all processes should be running concurrently
9. Please implement the best database needed for this project.
10. Rate limiting - The system should be able to service up to 50 pics waiting in database, if clients keep on sending pics their requests should fail until space on Shared memory will be available.
11. Keep track (log file) of amount of pics entered:
  - a. Client's IP Address.
  - b. The time at which Pic was received.
  - c. The inference (1 , 2 or 3).
12. If there are no more Pics enqueued in database the system (threads) should go to sleep and wake up when new pics will arrive.
13. In order to test your application, run 100 clients with a delay of 200msec between them.

Good Luck