LAB 02

Họ và tên: Nguyễn Văn Hồng

MSSV: 20173146

Lớp: CNTT-11 K62

1. Các file NED có tác dụng:
   * Cho phép người dùng khai báo các module đơn, các kết nối và cách các module ghép lại tạo thành module phức tạp.
   * Các khai báo giúp người dùng đặc tả được cấu trúc tĩnh của một mạng.
   * Ở ví dụ tic toc: txc1 là một loại module đơn, Tictoc1 là một mạng gồm 2 module con: tic và toc. Cả tic và toc đều có loại module là txc1. tic và toc kết nối với nhau nhờ các connection, có độ trễ lan truyền là 100ms.
2. Tác dụng của file omnetpp.int: khai báo mạng mà người dùng muốn mô phỏng (các mạng được định nghĩa trong file NED).
3. Ý nghĩa của đoạn mã: 

Cổng ra (out) của tic kết nối với cổng vào (in) của toc nhờ một kết nối có độ trễ lan truyền 100ms.

1. Giải thích đoạn mã nguồn

if (strcmp("tic", getName()) == 0) {

[cMessage](https://omnetpp.org/doc/omnetpp/api/classomnetpp_1_1cMessage.html) \*msg = new [cMessage](https://omnetpp.org/doc/omnetpp/api/classomnetpp_1_1cMessage.html)("tictocMsg");

send(msg, "out");

}

Nếu getName() trả về “tic” thì hàm strcmp(“tic”, getName()) trả về giá trị 0

Khi đó một đối tượng cMessage được khởi tạo với giá trị truyền vào “tictocMsg” sau đó thực hiện hàm send với các tham số: đối tượng cMessage và cổng ra “out”.

1. Các module tic và toc được khai báo trong submodules của file NED nên chúng tách biệt nhau.
2. Vai trò của các biến counter và limit.
   * Counter: được gán với một giá trị nhất định khi khởi tạo dùng để đếm số lần trao đổi dữ liệu, bị giảm đi một đơn vị sau mỗi trao đổi. Ý nghĩa: dùng để kết thúc mô phỏng khi thực hiện được một số lần hữu hạn các cuộc trao đổi dữ liệu.
   * Limit: được khai báo tại parameters của NED file và được gán cho một giá trị mặc định. Ý nghĩa: counter có thể đọc và lấy giá trị mặc định trong quá trình khởi tạo initialize().

**void** **Txc1::handleMessage**(cMessage \*msg)

{

// The handleMessage() method is called whenever a message arrives

// at the module. Here, we just send it to the other module, through

// gate `out'. Because both `tic' and `toc' does the same, the message

// will bounce between the two.

**if**(**strcmp**("toc", getName()) == 0){

msg = **new** cMessage("Hello, I am Toc. How are you. Tic!");

send(msg, "out");

}**else**{

msg = **new** cMessage("Hello, I am Tic. How are you. Toc!");

send(msg, "out");

}

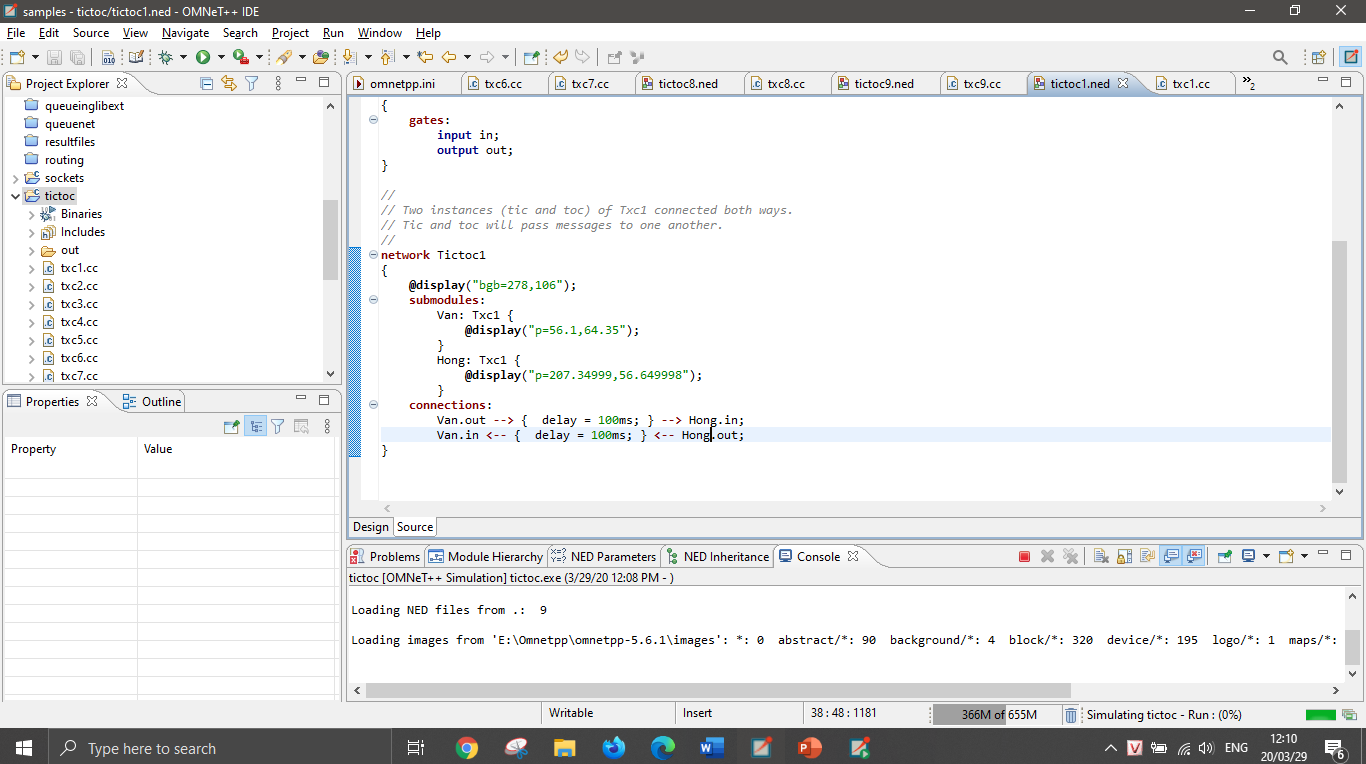
}

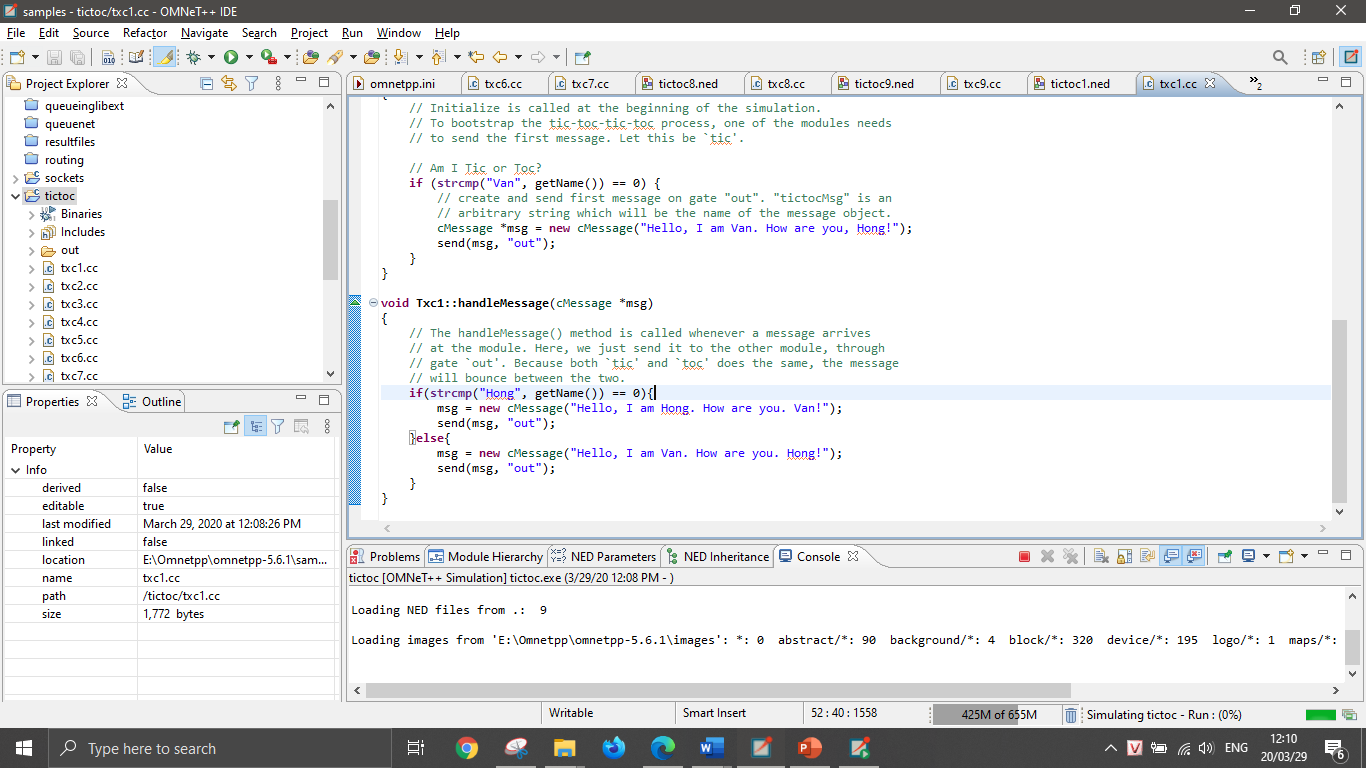
Trong handleMessage em check xem nếu module gửi là toc thì gửi đi với message

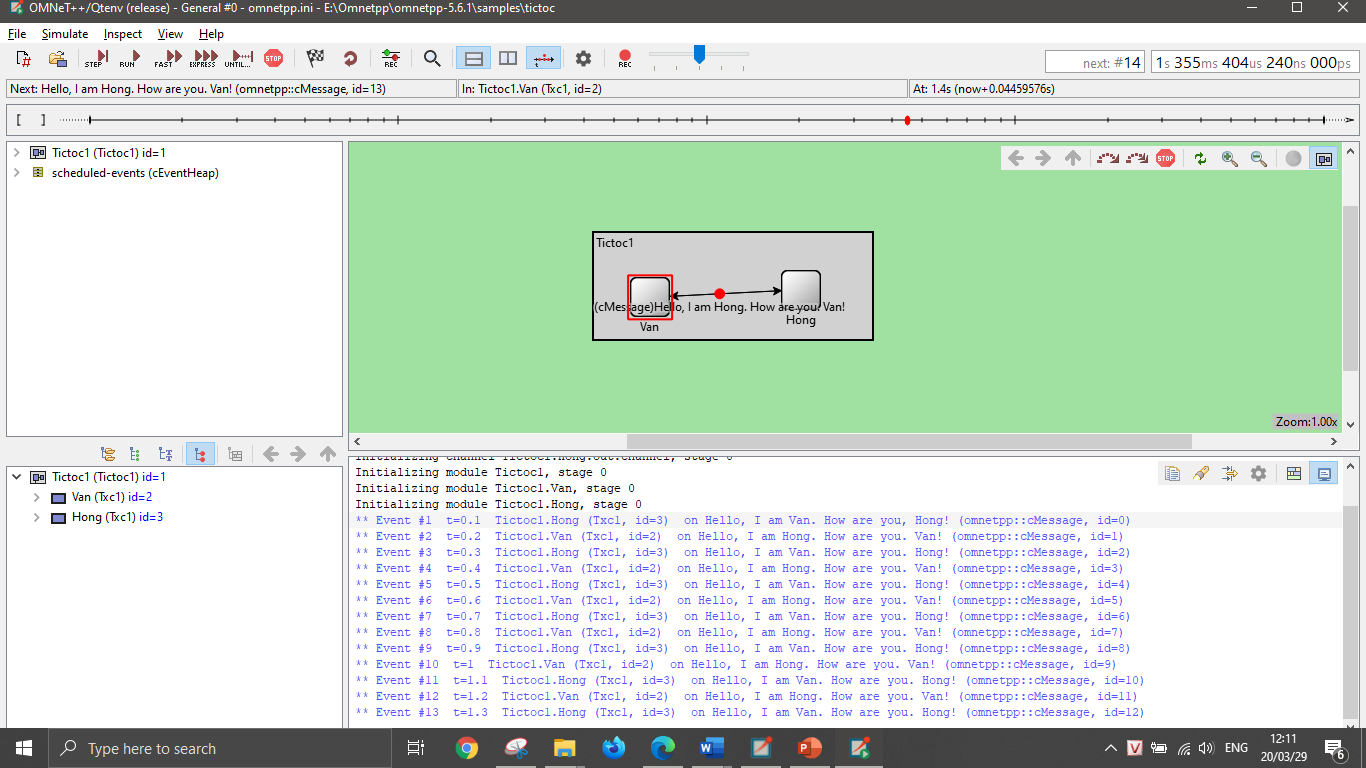
“Hello, I am Toc. How are you. Tic!”

Ngược lại module gửi là tic thì gửi đi với message

“Hello, I am Tic. How are you. Toc!”

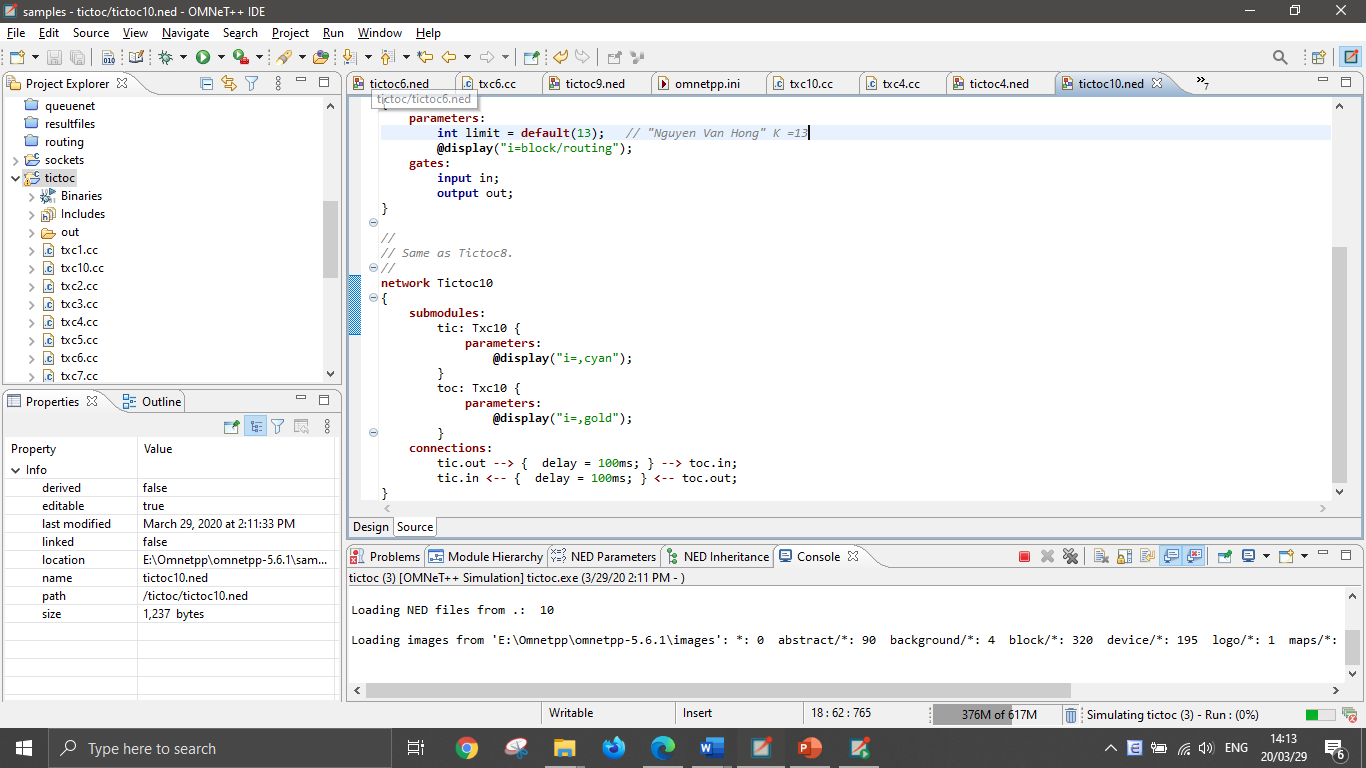
1. 

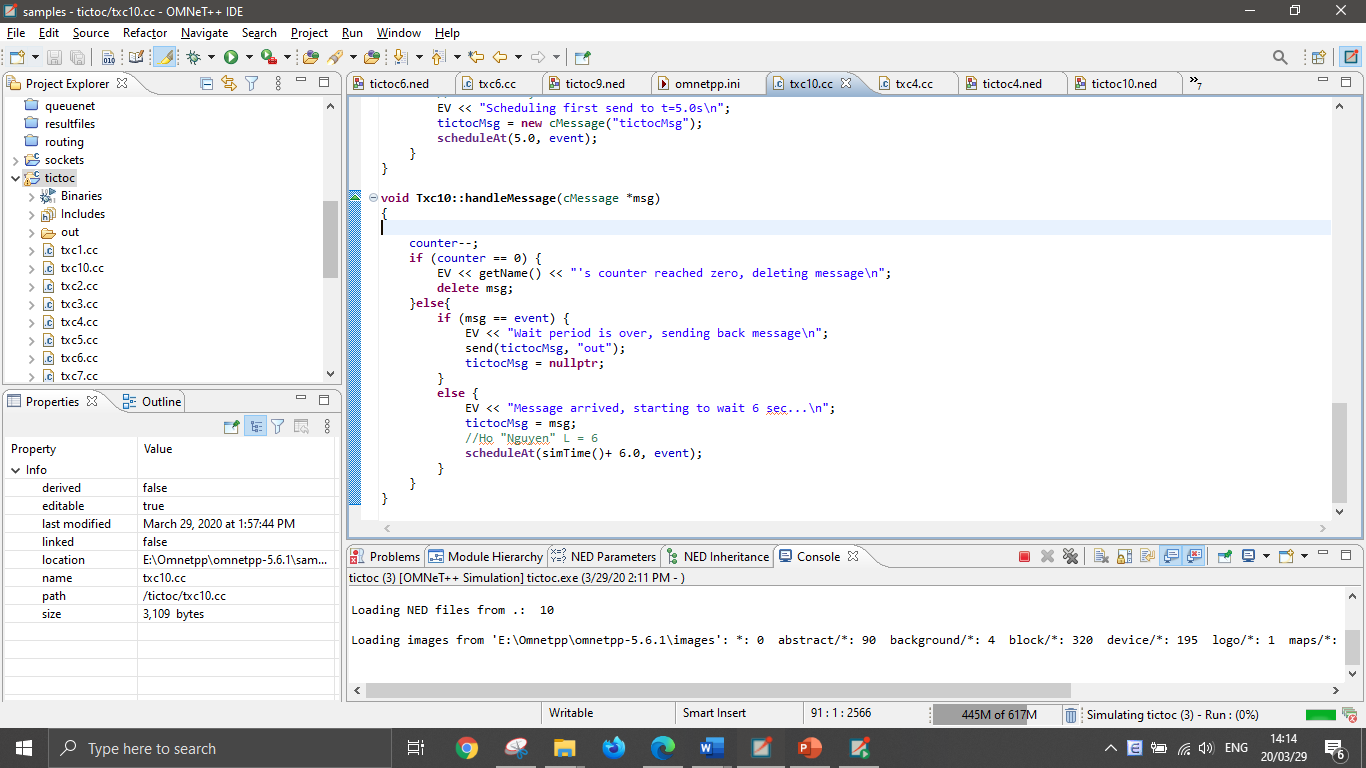


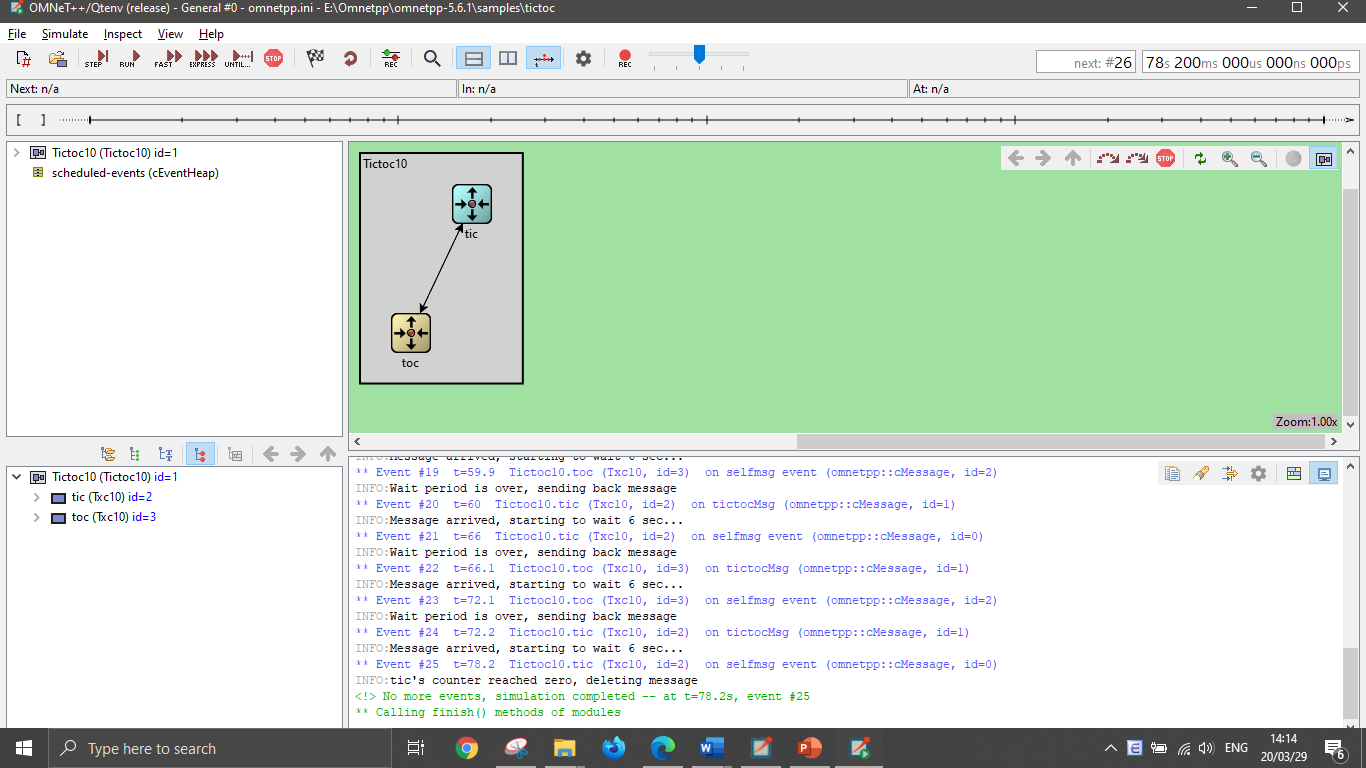


1. Ý nghĩa câu lệnh : scheduleAt([simTime](https://omnetpp.org/doc/omnetpp/api/group__SimCore.html#ga546abe6e6fd7a646f74d81cc944a3f08)()+timeout,timeoutEvent):

Thực hiện timeoutEvent sau khi dừng 1 thời gian = timeout

1. 





Họ tên: Nguyễn Văn Hồng

K=13, L =6