

## CHƯƠNG I: TỔNG QUAN VỀ MẠNG TRUYỀN SỐ LIỆU

### 1 Sơ đồ tổng quan hệ thống thông tin

- Sơ đồ khối tổng quát của một hệ thống thông tin ở dạng đơn giản sau:



- Trong đó:

1. Thông tin vào  $m$ .
2. Dữ liệu hay tín hiệu  $g(t)$  đầu vào.
3. Tín hiệu phát  $s(t)$ .
4. Tín hiệu thu  $r(t)$ .
5. Dữ liệu  $g'$  hay tín hiệu  $g'(t)$  thu được.
6. Thông tin ra  $m'$ .

- Chức năng các khối:

- Thiết bị vào: là thiết bị tạo ra thông tin để truyền đi.

Ví dụ: máy tính, điện thoại...

- Thiết bị phát: chuyển đổi và mã hóa thông tin thành tín hiệu điện từ có thể truyền được qua hệ thống truyền dẫn.

Ví dụ: modem (modulation/ demodulation)

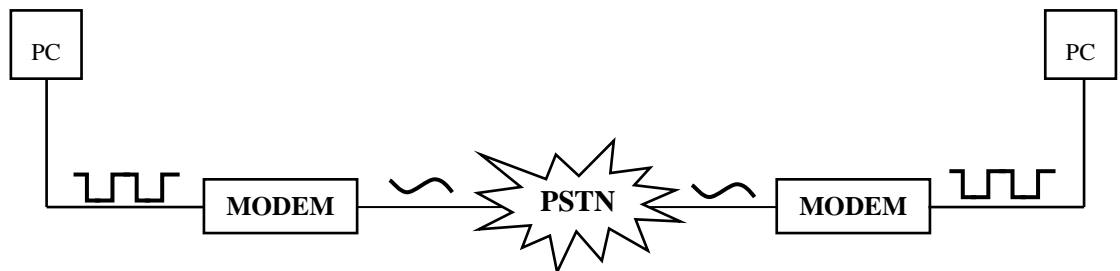
- Môi trường truyền dẫn: là môi trường truyền đơn hoặc một mạng liên hợp được kết nối tới các hệ thống nguồn và đích.

Ví dụ: PSTN (Public Switching Telephone Network\_Mạng điện thoại chuyển mạch công cộng)

- Thiết bị ra: nhận tín hiệu từ thiết bị thu và biến đổi nó thành thông tin đầu ra.

Ví dụ: máy tính, máy điện thoại

- Ví dụ minh họa hệ thống thông tin.

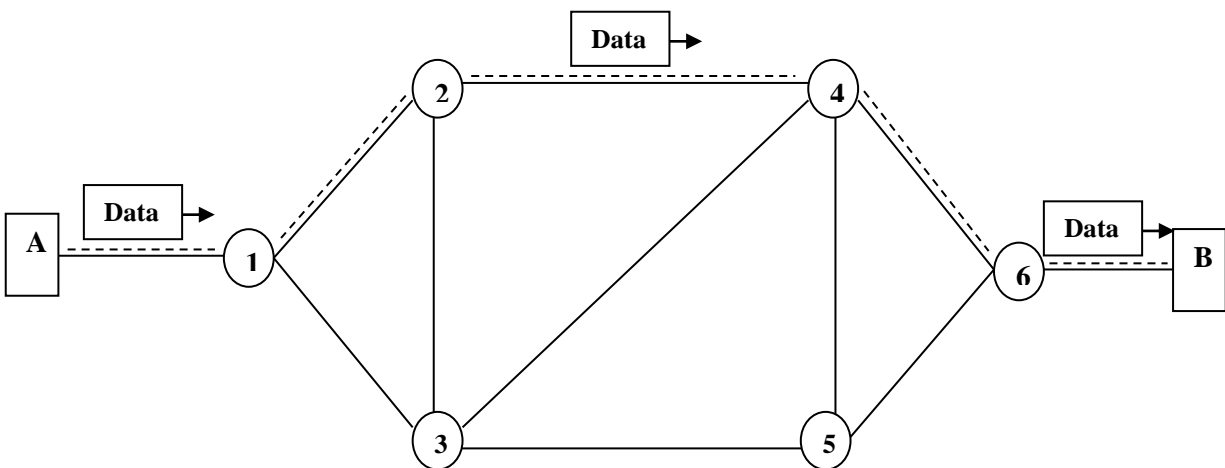


- Bên phát: thông tin nhập vào từ PC qua PC thông tin có dạng tín hiệu số, tín hiệu này được đưa tới MODEM. MODEM biến đổi tín hiệu số thành tín hiệu tương tự truyền trong mạng tương tự PSTN
- Bên thu: MODEM biến đổi tín hiệu tương tự nhận được từ môi trường truyền dẫn PSTN thành tín hiệu số để PC có thể hiểu được.

## 2 Mạng truyền thông chuyển mạch

### 2.1 Mạng chuyển mạch kênh (Circuit Switched Network)

Trong mạng chuyển mạch kênh khi trạm cần trao đổi thông tin với nhau thì giữa chúng sẽ được thiết lập 1 kênh (Circuit) cố định và duy trì kênh vật lý đó cho đến khi 1 trong 2 trạm ngắt liên lạc.



Ví dụ điển hình của mạng chuyển mạch kênh là chuyển mạch của điện thoại.

Giả sử A cần trao đổi thông tin với B qua mạng điện thoại công cộng. Khi đó quá trình trao đổi thông tin phải thực hiện qua các bước sau:

- Bước 1: thiết lập 1 kênh truyền vật lý từ A đến B (đường nét đứt)
- Bước 2: Kiểm tra xem bên B đã sẵn sàng nhận dữ liệu hay chưa ? (B có rỗi hay không? ). Nếu B chưa sẵn sàng nhận dữ liệu (B đang bận) thì bên gửi A cúp máy (phá bỏ đường truyền đã được thiết lập ngay trước đó), sau đó bên gửi A đợi một thời gian rồi quay lại bước 1. Nếu B sẵn sàng nhận dữ liệu thì hai bên sẽ tiến hành truyền/ nhận dữ liệu.
- Bước 3: Sau khi kết thúc việc truyền nhận dữ liệu thì phải giải phóng đường truyền.
- Ưu điểm:
  - Kênh truyền được dành riêng trong suốt quá trình giao tiếp do đó tốc độ dữ liệu luôn ổn định điều này đặc biệt quan trọng trong truyền Audio, Video.
  - Không có trễ truy nhập do kênh truyền luôn sẵn sàng vì vậy việc truyền lại là không cần thiết.
- Nhược điểm:

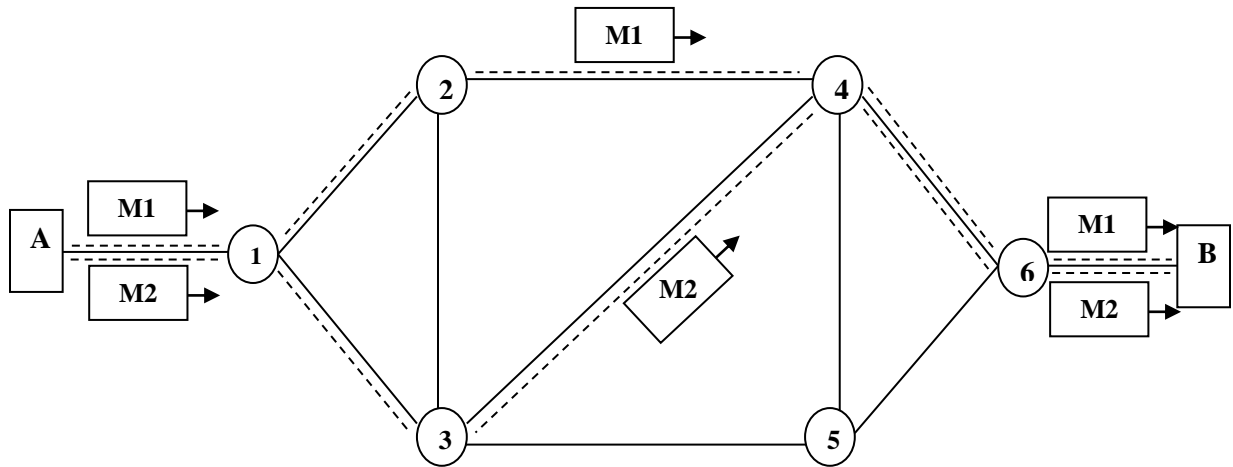
Phương pháp chuyển mạch kênh có 2 nhược điểm chính là:

- Tiêu tốn thời gian thiết lập đường truyền (kênh) cố định giữa 2 trạm.
- Hiệu suất sử dụng kênh truyền không cao vì có lúc kênh bị bỏ không do cả 2 bên đều hết thông tin để truyền trong khi các trạm khác không được phép sử dụng kênh truyền này.

### **2.2 Mạng chuyển mạch thông báo (Message Switched Network)**

Thông báo (Message) là một đơn vị thông tin của người sử dụng, có khuôn dạng quy định trước. Mỗi thông báo đều có chứa vùng thông tin điều khiển, trong đó chỉ rõ đích của thông báo. Căn cứ vào thông tin này mà mỗi nút trung gian có thể chuyển thông báo tới nút kế tiếp theo đường dẫn tới đích của nó. Như vậy, ở tại

mỗi nút mạng cần lưu trữ tạm thời để đọc thông tin điều khiển trên thông báo rồi sau đó chuyển thông báo đi. Tùy thuộc vào điều kiện của mạng, các thông báo khác nhau có thể được gửi đi trên nhiều đường khác nhau.



- Ưu điểm:

Phương pháp chuyển mạch thông báo có nhiều ưu điểm hơn so chuyển mạch kênh:

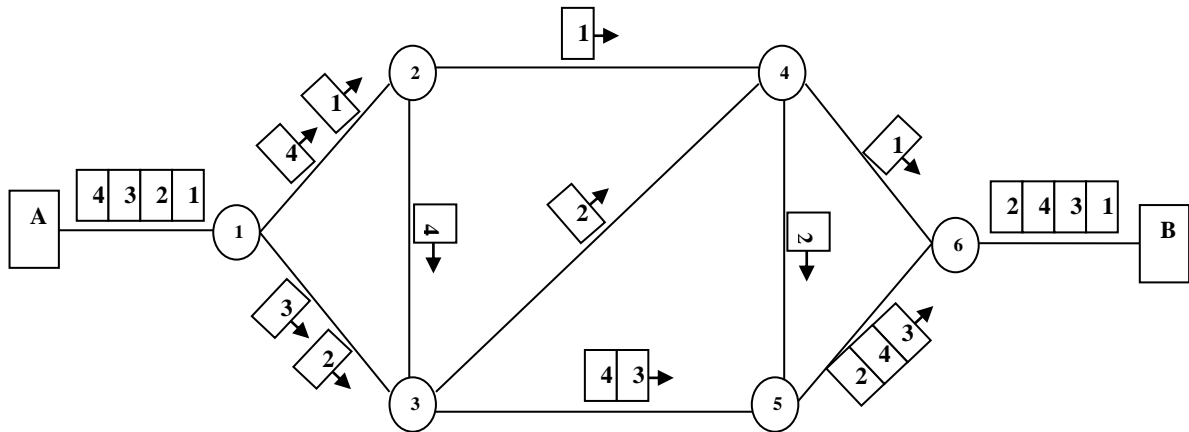
- Hiệu suất sử dụng đường truyền cao hơn vì không bị chiếm dụng độc quyền mà được phân chia giữa nhiều trạm.
- Ở tại mỗi nút mạng có thể lưu trữ thông báo cho tới khi kênh truyền rỗi rồi mới gửi thông báo đi, do đó giảm được tình trạng tắc nghẽn mạng.
- Có thể tăng hiệu suất sử dụng giải thông của mạng bằng cách gán địa chỉ quảng bá (Broadcast Addressing) để gửi thông báo đồng thời tới nhiều đích khác nhau

- Nhược điểm:

- Không hạn chế kích thước của thông báo, có thể dẫn tới phí tổn lưu trữ tạm thời cao và ảnh hưởng tới thời gian phức tạp và chất lượng truyền đi (không đáp ứng được tính thời gian thực).
- Thích hợp với các dịch vụ thông tin không đòi hỏi tính thời gian thực (Real time) như dịch vụ Email (Electric mail).

### 2.3 Mạng chuyển mạch gói (Packet Switched Network)

- Với phương thức chuyển mạch gói, mỗi thông báo được chia thành nhiều đơn vị nhỏ hơn gọi là gói tin (Packet) có khuôn dạng định trước. Mỗi gói tin chứa các thông tin điều khiển, trong đó có địa chỉ nguồn (người gửi) địa chỉ đích (Người nhận) của gói tin. Các gói tin thuộc một thông báo có thể gửi qua mạng tới đích bằng nhiều đường khác nhau.



- Trong mạng chuyển mạch gói, các gói tin được giới hạn kích thước tối đa MTU (Maximum Transfer Unit) sao cho các nút mạng có thể xử lý toàn bộ gói tin trong bộ nhớ mà không lưu trữ tạm thời trên đĩa hoặc bộ nhớ phụ (bộ đệm). Bởi thế nên mạng chuyển mạch gói truyền các gói tin qua mạng nhanh hơn và hiệu quả hơn mạng chuyển mạch thông báo. Giới hạn MTU tùy thuộc vào từng công nghệ mạng.

Ví dụ:

Giao thức X25 có MTU = 128 byte

Frame Relay có MTU = 4096 byte

ATM có MTU = 53 byte

Trong chuyển mạch gói có 2 phương pháp khác nhau được sử dụng, đó là: Datagram và chuyển mạch ảo (Virtual Circuit) hay còn gọi là kênh logic (logic channel).

- Datagram

Trong phương pháp này, mỗi gói tin tới nút thì nó được nhân bản lên rồi đẩy đi hướng đích tới các nút kế tiếp nhưng được định hướng độc lập với nhau. Ở đây mỗi nút chuyển mạch sẽ xác định phần mạng nào sẽ được sử dụng ở các bước tiếp theo của tuyến. Nó có khả năng chuyển mạch tới các nút bên để đạt được tốc độ truyền tin gói nhanh.

Với phương pháp này, vì các gói được nhân bản ở mỗi nút trên trao đổi 1 lượng thông tin lớn đến 1 lúc nào đó mạng sẽ bị nghẽn. Chính vì vậy phương pháp này chỉ được ứng dụng đối với những mạng trao đổi ít.

- Chuyển mạch ảo (Virtual Circuit)

Với phương pháp này, nó hoạt động giữa trên sự thiết lập giữa 2 thiết bị đầu cuối. Khi bắt đầu một phiên kết nối, hai thiết bị đầu cuối sẽ thỏa thuận kích cỡ của gói tin, đường truyền... từ đó nó tạo ra một kênh ảo. Kênh ảo (không phải là kênh vật lý) là đường truyền quy định trước qua mạng.

Tại mỗi nút mạng có nhiều kênh ảo đến và đi khác nhau trong khi phương pháp chuyển mạch kênh chỉ có 1 đường đi duy nhất.

Có 2 loại mạch ảo được sử dụng đó là: VC (Virtual Circuit) và PVC (Permanence Virtual Circuit).

- Kênh ảo (VC): Trước khi trao đổi thông tin thì thiết lập kênh ảo giữa nguồn và đích. Sau khi hai bên không có nhu cầu trao đổi thông tin thì hủy bỏ.

- Kênh ảo thường trực (PVC): Thiết lập 1 kênh ảo vĩnh cửu giữa 2 thiết bị đầu cuối có thể truyền nhận dữ liệu bất kỳ lúc nào mà không cần thiết lập kênh ảo và giải phóng nó.

- Ưu điểm:

- Sử dụng đường truyền hiệu quả hơn so với phương pháp chuyển mạch kênh vì nhiều gói tin có thể dùng chung một đường truyền.
- Tốc độ dữ liệu được giải quyết vì các trạm kết nối tới nút ở 1 tốc độ dữ liệu thích hợp
  - Không xảy ra hiện tượng tắc nghẽn như phương pháp chuyển mạch thông báo. Khi mạng bị quá tải, các gói tin vẫn được nhận nhưng thời gian trễ tăng lên
  - Các gói tin có thể được cung cấp một số thuộc tính như là: Các gói tin có mức ưu tiên cao hơn sẽ có độ trễ nhỏ hơn những gói tin có mức ưu tiên thấp hơn.
- Nhược điểm:
  - Do việc chia thông báo thành nhiều gói tin nhỏ hơn, nên hiệu suất truyền tin giảm.
  - Cần có cơ chế sắp xếp lại các gói tin đúng số thứ tự ở bên thu vì các gói tin có thể đến bên thu không đúng thứ tự.

### **3 Chuẩn hóa và mô hình tham chiếu OSI**

#### ***3.1 Kiến trúc phân tầng.***

*Ý nghĩa của việc phân tầng.*

Đơn giản cho việc phân tích thiết kế.

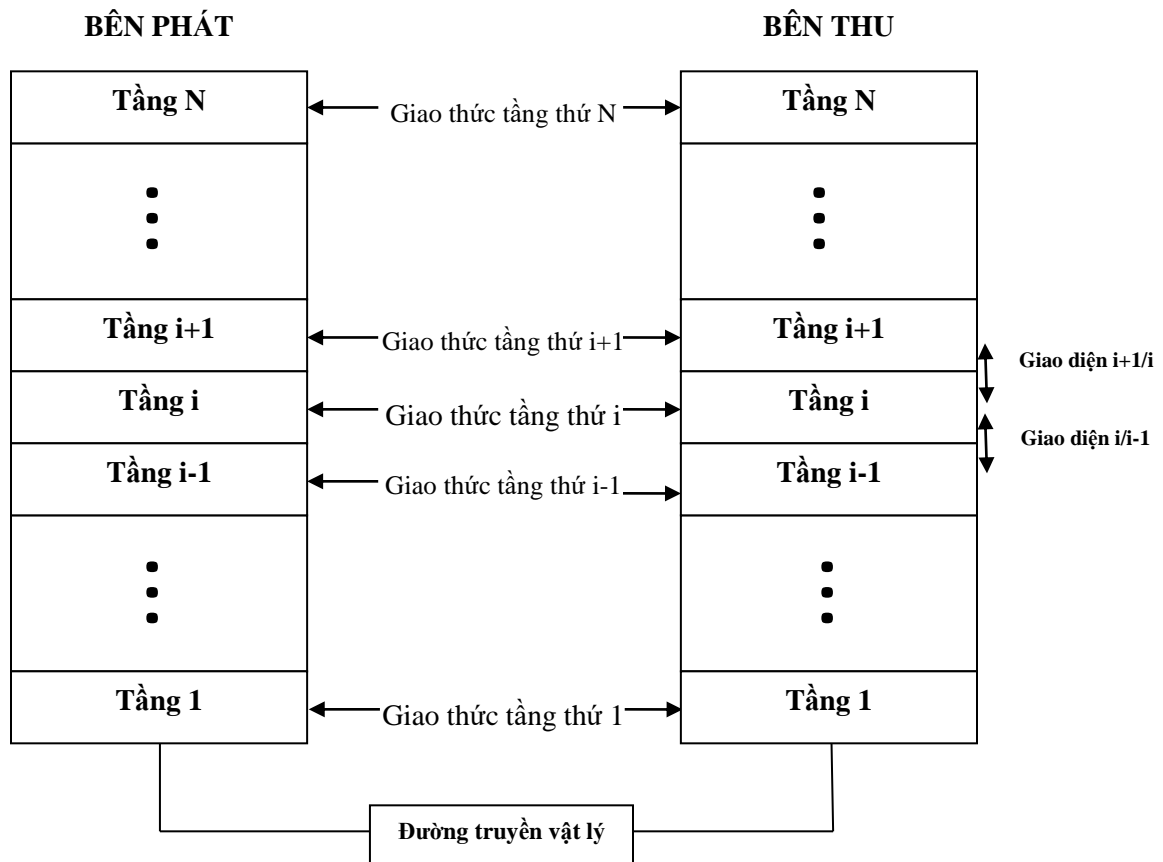
Tạo khả năng modul hóa cao.

Dễ dàng cho việc tiêu chuẩn hóa giao diện.

Đảm bảo khả năng làm việc giữa các công nghệ khác nhau.

Gia tốc cho những hướng phát triển mới.

- Nguyên tắc phân tầng
- Số lượng, chức năng các tầng trong hệ thống phải như nhau .



- Xác định mối quan hệ giữa các đồng tầng (gọi là giao thức tầng).

+ Trước khi liên lạc các tầng phải đảm bảo bắt tay hiểu nhau và cùng thỏa thuận các tham số sử dụng.

+ Trong tầng có các thực thể hoạt động: Thực thể thường là các thủ tục hoặc chương trình con

- Xác định mối quan hệ giao diện giữa 2 tầng kề nhau, tầng thứ N phải biết:

+ Phải cung cấp những dịch vụ gì cho các hoạt động của thực thể tầng N+1.

+ Được sử dụng dịch vụ gì của tầng N-1.



- Dữ liệu không được truyền trực tiếp từ tầng thứ  $i$  của hệ thống này sang tầng thứ  $i$  của hệ thống khác (trừ tầng thấp nhất).

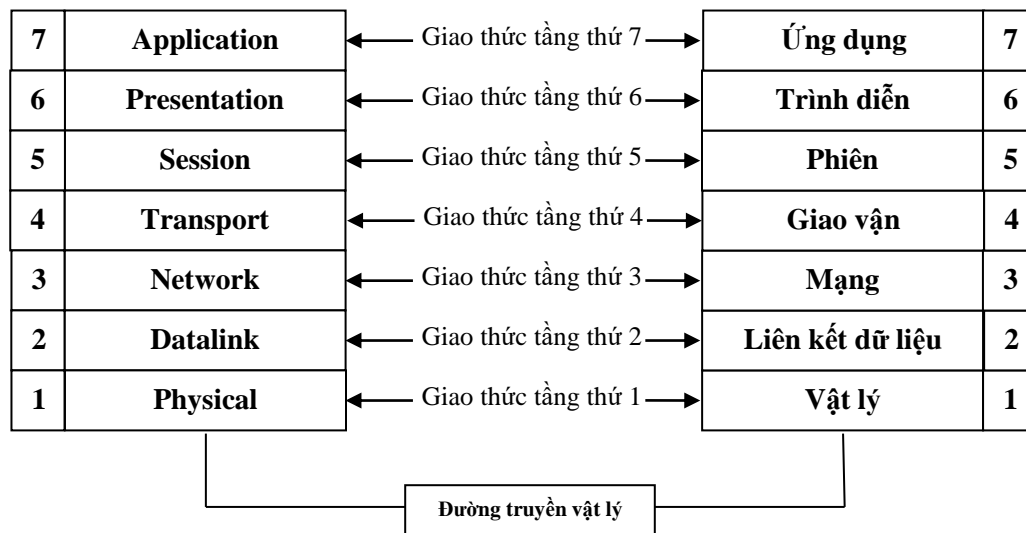
+ Bên phát: Dữ liệu truyền từ tầng cao nhất xuống tầng thấp nhất qua đường truyền vật lý sang bên thu.

+ Bên thu: Sau khi nhận được thông tin (chuỗi 0,1) sau đó truyền từ tầng thấp nhất lên tầng cao nhất.

### 3.2 Mô hình OSI (Open Systems Interconnection)

*Mô hình tham chiếu kết nối các hệ thống mở OSI*

Mô hình OSI gồm có 7 tầng với tên gọi và chức năng như sau:



- Chức năng các tầng trong mô hình OSI.

- Tầng 1: Tầng vật lý (Physical layer).

Truyền chuỗi bit không cấu trúc qua đường truyền vật lý.

Nhận các khung tin (Frame) dữ liệu từ tầng liên kết dữ liệu (data link) và chuyển dòng bit thành các tín hiệu trên môi trường truyền dẫn vật lý.

Liên quan đến việc đưa ra các kiểu dây dẫn được sử dụng, các kiểu thiết bị kết nối được sử dụng để kết nối mọi thiết bị với môi trường truyền dẫn và sơ đồ tín hiệu.

- Tầng 2: tầng liên kết dữ liệu (Data link layer).

Truyền dữ liệu giữa các đầu cuối của đường truyền vật lý.

Cung cấp cơ chế phát hiện lỗi, tạo khung và điều khiển luồng dữ liệu.

Giải quyết về các vấn đề hỏng khung, mất khung hoặc trùng khung.

Định dạng thông báo như là các Frame hơn nữa là các Packet.

- Tầng 3: Tầng mạng (Network).

Định tuyến end-to-end hoặc chuyển dữ liệu để thiết lập một kết nối dùng cho việc chuyển dữ liệu 1 cách dễ dàng.

Đánh địa chỉ và giải quyết các vấn đề có liên quan tới việc truyền dữ liệu giữa các mạng không đồng nhất.

Sử dụng tầng giao vận ở trên và tầng liên kết ở dưới.

Định dạng thông báo là Packet.

- Tầng 4: Tầng giao vận (Transport layer).

Truyền dữ liệu không lỗi. Nhận dữ liệu từ tầng phiên, phân dữ liệu thành các gói tin có kích thước nhỏ hơn nếu cần thiết, chuyển các gói tin tới tầng mạng và đảm bảo các gói tin tới đích là không sai và toàn vẹn.

- Tầng 5: Tầng phiên (Session layer).

Cung cấp các khả năng phối hợp giữa các tiến trình truyền thông giữa các nút. Ép buộc các bên phải tuân thủ theo quy tắc hội thoại (cho phép truyền song công hay bán song công), đồng bộ hóa luồng dữ liệu, và thiết lập lại một kết nối trong trường hợp có sự cố lỗi xảy ra.

Sử dụng tầng trình diễn ở trên và tầng giao vận ở dưới.

- Tầng 6: Tầng trình diễn (Presentation layer).

Cung cấp các phương pháp định dạng dữ liệu, chuyển đổi, và sự chuyển đổi mã. Liên quan đến cú pháp và ngữ nghĩa của dữ liệu được truyền đi. Nén và mã hóa dữ liệu. Nhận dạng thông báo từ tầng ứng dụng, định dạng nó và chuyển tới tầng phiên.

- Tầng 7: Tầng ứng dụng (Application layer).

Bao gồm các giao thức định nghĩa chính xác, rõ ràng các chương trình ứng dụng hướng người sử dụng như là Email, truyền file và đầu cuối ảo. Cung cấp các phương tiện (giao diện tương tác) để cho người dùng truy xuất vào hệ thống.

Cung cấp các dịch vụ thông tin phân tán. Cung cấp các giao diện lập trình ứng dụng.

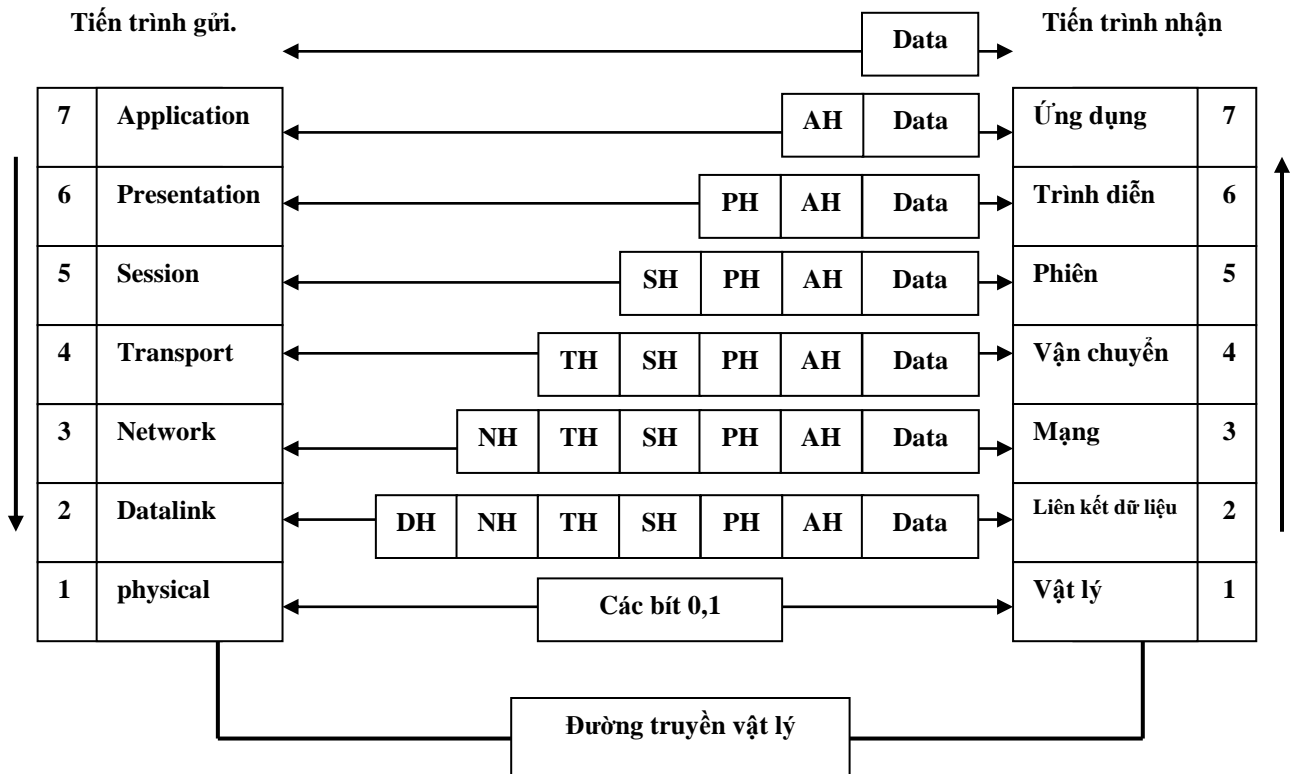
*Nguyên tắc hoạt động của mô hình OSI.*

Nguyên tắc hoạt động của mô hình OSI dựa trên cơ chế đóng gói dữ liệu.

Tiến trình gửi: Dữ liệu được gửi từ người sử dụng được đưa vào tầng trên cùng (tầng ứng dụng), tầng này xử lý tùy theo yêu cầu tùy thuộc vào giao thức được sử dụng, sau đó gán Header của tầng ứng dụng là AH vào đầu dữ liệu và chuyển xuống tầng 6 (Tầng trình diễn). Tại tầng trình diễn, dữ liệu lại được gán Header của tầng trình diễn là PH vào đầu dữ liệu (lúc này dữ liệu tầng 7 và Header

AH được coi là dữ liệu tầng 6) sau đó chuyển xuống tầng 5 (tầng phiên) tiến trình được lặp lại cho đến khi dữ liệu chuyển tới tầng vật lý. Tại tầng vật lý (tầng 1) dữ liệu truyền tới máy nhận qua kênh truyền dẫn dưới dạng các bit 0,1.

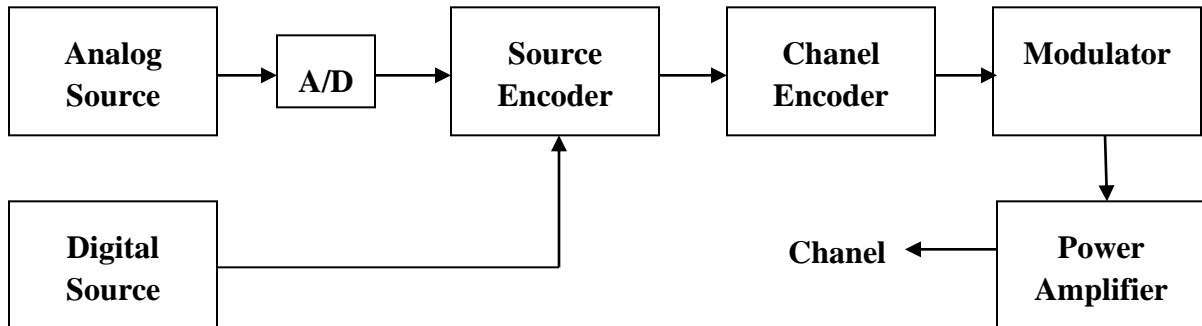
Tiến trình nhận: Thực hiện ngược lại với tiến trình gửi. Dữ liệu từ tầng dưới đi lên sẽ được tách bỏ 1 Header ở mỗi tầng trước khi gửi dữ liệu lên tầng trên nó cho đến cuối tiến trình nhận.



## CHƯƠNG III. BIẾN ĐỔI DỮ LIỆU THÀNH TÍN HIỆU

### 1 Mô hình hệ thống thông tin số (dạng đơn giản)

Mô hình hệ thống thông tin (số) được thể hiện như hình 3.1 sau:



- Chức năng các khối:

- Nguồn tương tự (Analog Source): nguồn tin cần truyền thể hiện dạng tương tự.

Ví dụ: Tín hiệu thoại.

- Nguồn tin số (Digital Source): nguồn tin cần truyền thể hiện dưới dạng số.

- A/D (Analog/Digital): chuyển đổi tín hiệu tương tự thành tín hiệu số.

- Bộ mã hóa nguồn (Source Encoder): Thực hiện nén nhằm giảm phổ chiếm của tín hiệu để giảm tốc độ bit, giảm băng thông yêu cầu.

Ví dụ: Các bộ mã hóa nguồn ASCII, Shanon-Fano, Huffman...

- Bộ mã hóa kênh (Chanel Encoder): Nhằm chống nhiễu và các tác động xấu khác của đường truyền. Mã hóa kênh làm tăng tốc độ dư thừa của thông tin, do đó làm tăng tốc độ bit làm tăng băng thông yêu cầu trên đường truyền. Tăng độ dư thừa của thông tin nhằm phát hiện và sửa lỗi.

Ví dụ: Các mã chống nhiễu (các mã thực hiện chức năng mã hóa kênh): mã Cyclic, mã Hamming, mã BCH

- Bộ điều chế (Modulator): Định dạng các kí hiệu số thành dạng sóng phù hợp cho việc truyền dẫn.
- Bộ khuếch đại công suất (Applifier): Khuếch đại tín hiệu sau khi điều chế để đưa vào kênh truyền.
- Phía thu: Thực hiện hoàn toàn ngược lại.

## 2 Biến đổi dữ liệu số thành tín hiệu tương tự

### 2.1 Kỹ thuật điều chế ASK (*Amplitude Shift Keying*).

### 2.2 Kỹ thuật điều chế FSK (*Frequency Shift Keying*).

#### 2.2.1 Kỹ thuật điều chế BFSK.

Dùng 2 tần số khác nhau để biểu diễn trạng thái của các bit 0,1. Tín hiệu trên đường truyền có dạng:

$$\begin{cases} s_1(t) = A \cos(2\pi f_1 t + \Phi_1); & KT \leq t \leq (K+1)T: \text{biểu diễn bit 1} \\ s_2(t) = A \cos(2\pi f_2 t + \Phi_2); & KT \leq t \leq (K+1)T: \text{biểu diễn bit 0} \end{cases}$$

Trong đó:

$T$  là độ rộng bit dữ liệu.

$\Phi_1, \Phi_2$  là các pha ban đầu.

Trường hợp  $\Phi_1 = \Phi_2$  ta có phương pháp điều chế Coherent BFSK

Trường hợp  $\Phi_1 \neq \Phi_2$  ta có phương pháp điều chế Noncoherent BFSK

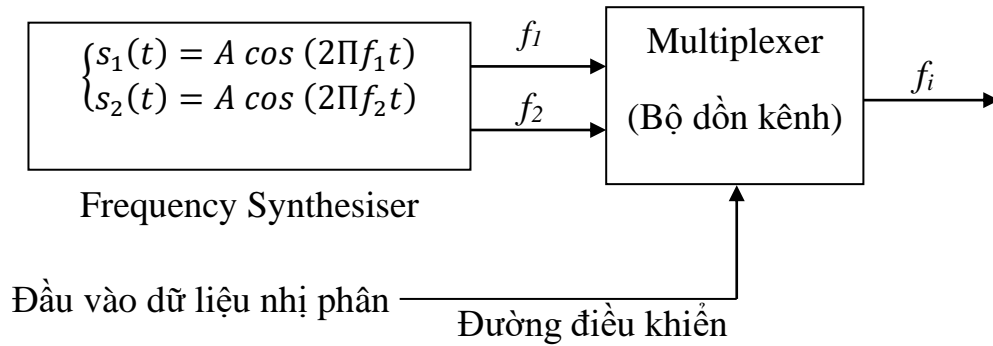
#### a. Kỹ thuật điều chế Coherent BFSK

Với kiểu điều chế này 2 tín hiệu có pha ban đầu là  $\Phi$  tại thời điểm  $t=0$

$$\begin{cases} s_1(t) = A \cos(2\pi f_1 t + \Phi) & \text{Biểu diễn bit 1} \\ s_2(t) = A \cos(2\pi f_2 t + \Phi) & \text{Biểu diễn bit 0} \end{cases}$$

- Để cho đơn giản, giả sử  $\Phi = 0$

- Bộ điều chế:

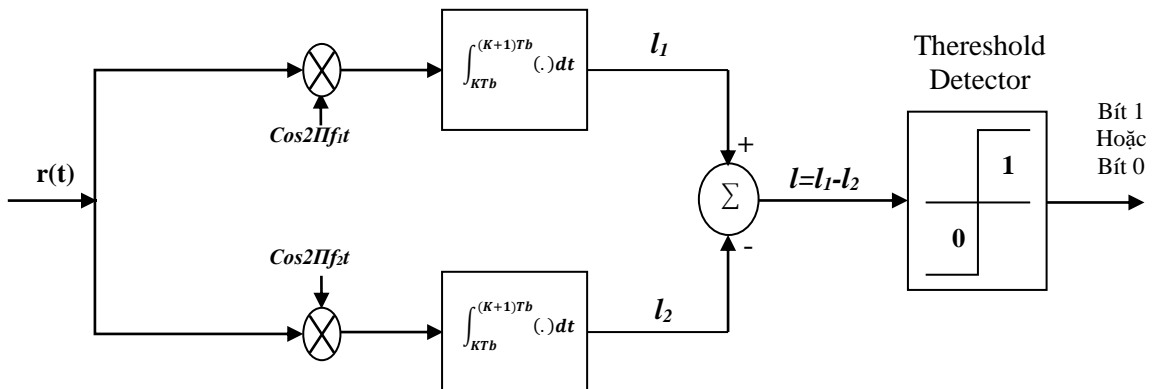


Hai tín hiệu  $s_1(t)$ ,  $s_2(t)$  được chọn sao cho trực giao với nhau. Tức là:

$$\int_{KT}^{(K+1)T} s_1(t)s_2(t)dt = 0$$

- Bộ giải điều chế:

Giả sử tín hiệu đầu vào của bộ giải điều chế là:  $r(t) = s_1(t) = A \cos 2\pi f_1 t$

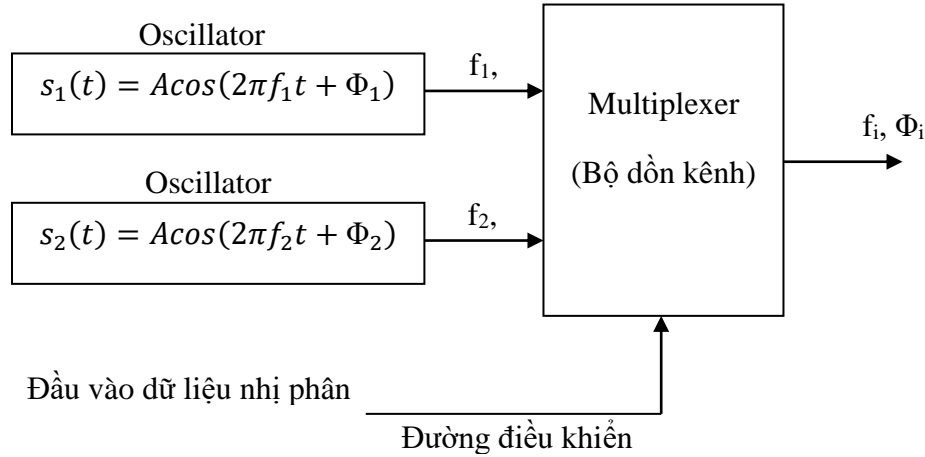


$$l = \begin{cases} l_1 - l_2 > 0: \text{chọn bit 1} \\ l_1 - l_2 < 0: \text{chọn bit 0} \end{cases}$$

b. Kỹ thuật điều chế Noncoherent BFSK.

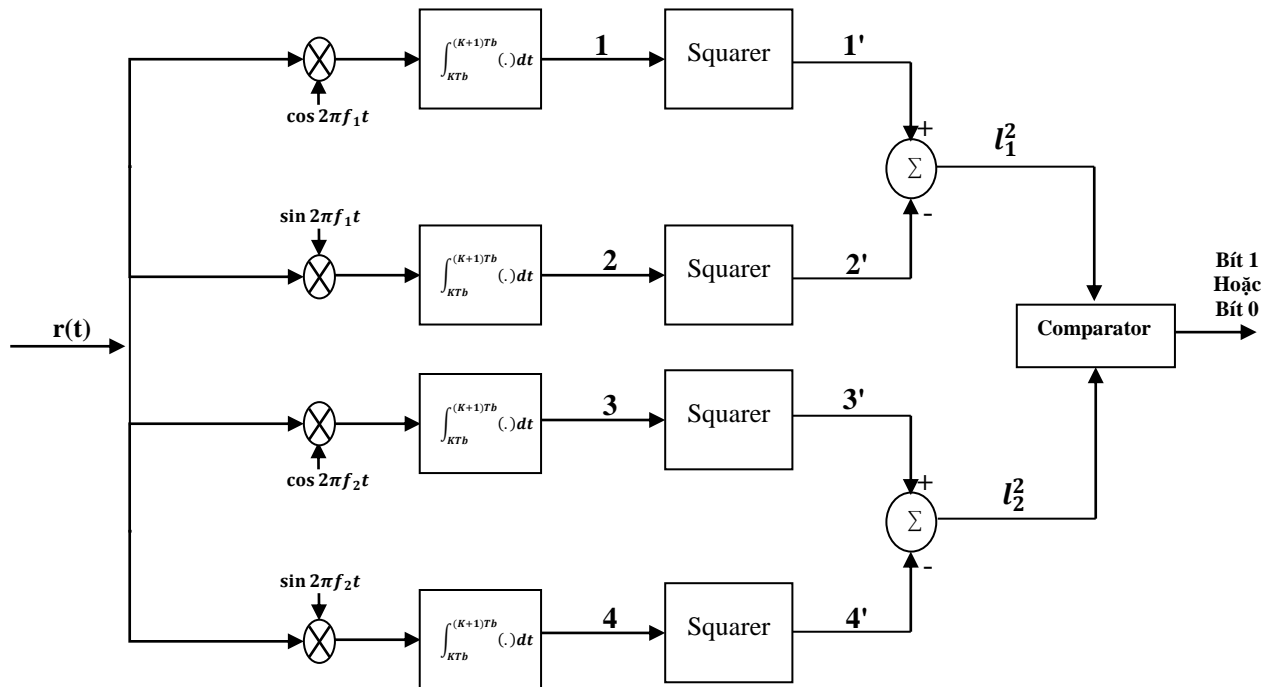
Tập tín hiệu:  $\begin{cases} s_1(t) = A\cos(2\pi f_1 t + \Phi_1). \text{ biểu diễn bit 1} \\ s_2(t) = A\cos(2\pi f_2 t + \Phi_2). \text{ biểu diễn bit 0} \end{cases}$

• Bộ điều chế.



• Bộ giải điều chế:

Giả sử tín hiệu thu được là:  $r(t) = s_i(f_i, \Phi_i) = A\cos(f_i t + \Phi_i)$  với  $\begin{cases} i = 1, 2 \\ \Phi_i: \text{Pha bất kỳ} \end{cases}$





### 2.2.2 Kỹ thuật điều chế $M$ -FSK ( $M$ -ary FSK).

- Dòng dữ liệu nhị phân đầu vào được chia thành tổ hợp bit. Hay còn gọi là symbol. Mỗi symbol có  $n = \log_2 M$  (bit) ( $M$  là số trạng thái của tín hiệu trên đường truyền).

- Nếu  $M$  có dạng lũy thừa của 2 thì:

$$n = \lceil \log_2 M \rceil + 1. \text{ Lấy số nguyên lớn hơn gần nhất.}$$

- Dòng  $M$  tín hiệu với các tần số khác nhau để biểu diễn các symbol.

→ Tốc độ truyền tăng, tỷ lệ lỗi bit tăng.

Trong thực tế lấy  $M = 2^n$ .

- Tín hiệu thứ  $i$  có thể biểu diễn là:

$$\begin{cases} s_i(t) = A \cos 2\pi f_i t + \Phi_i \\ i = \overline{1, M} \end{cases} \quad KT_s \leq t \leq (K+1)T_s. \text{ Biểu diễn symbol thứ } i.$$

Trong đó:

$M$  là số trạng thái tín hiệu trên đường truyền.

$T_s$  là độ rộng của symbol.  $T_s = nT_b$ .

$\Phi_i$  là các góc pha ban đầu.

+ Nếu  $\Phi_i \neq \Phi_j, \forall i \neq j$ : Kỹ thuật điều chế coherent MFSK

+ Nếu  $\Phi_i = \Phi_j, \forall i \neq j$ : Kỹ thuật điều chế Non coherent MFSK

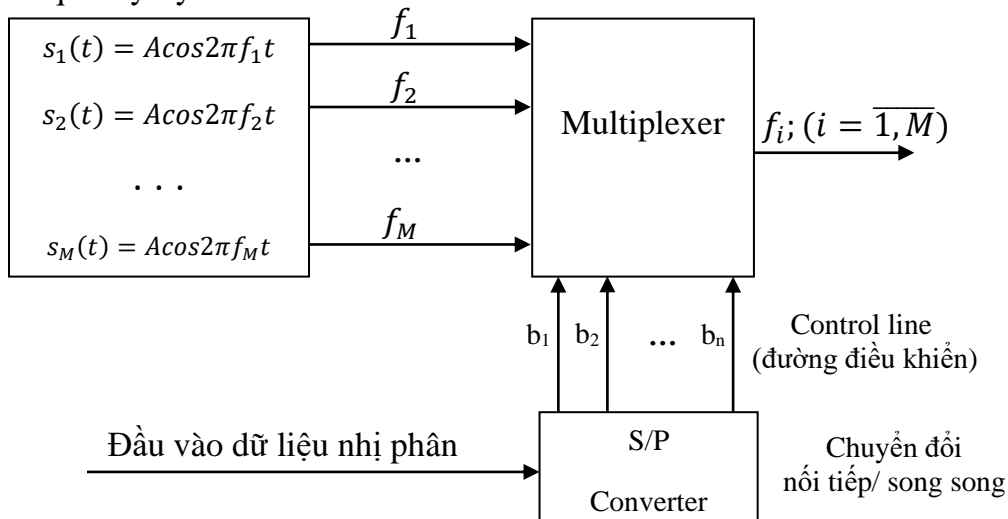
#### a. Kỹ thuật điều chế Coherent MFSK.

Để đơn giản, giả sử  $\Phi_i = 0$ . Lúc này tín hiệu thứ  $i$  có thể biểu diễn được như

sau: 
$$\begin{cases} s_i(t) = A \cos 2\pi f_i t. \text{ Biểu diễn bit } 1 \\ i = \overline{(1, M)} \end{cases}$$

- Bộ điều chế:

Frequency Synthesiser

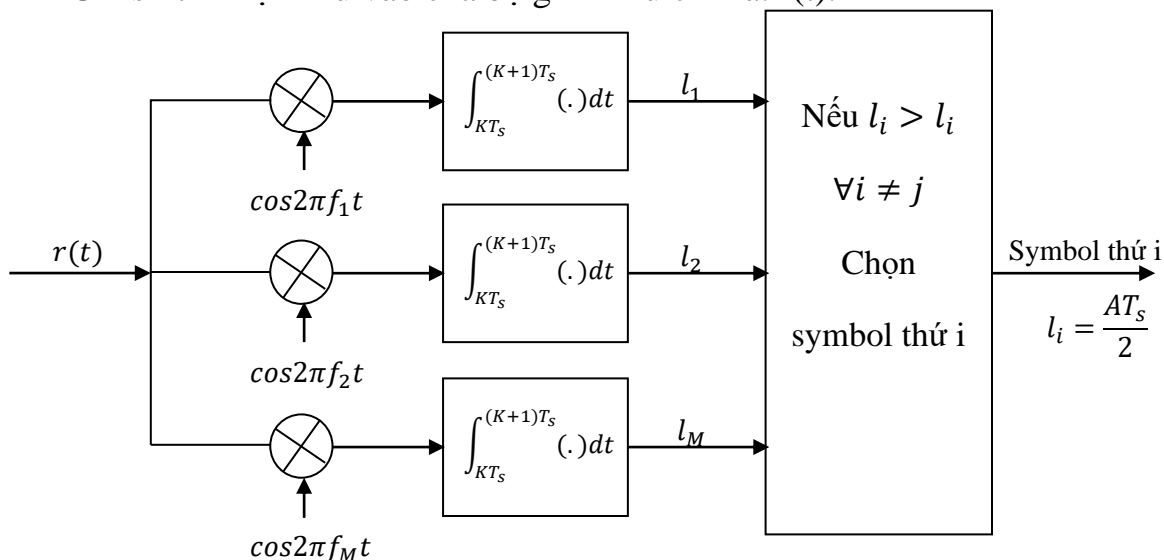


\* Chú ý: Các tín hiệu  $s_i(t)$  được chọn sao cho trực giao với nhau:

$$\int s_i(t)s_j(t)dt = 0; \forall i \neq j.$$

- Bộ giải điều chế:

Giả sử tín hiệu đầu vào của bộ giải điều chế là:  $r(t)$ .

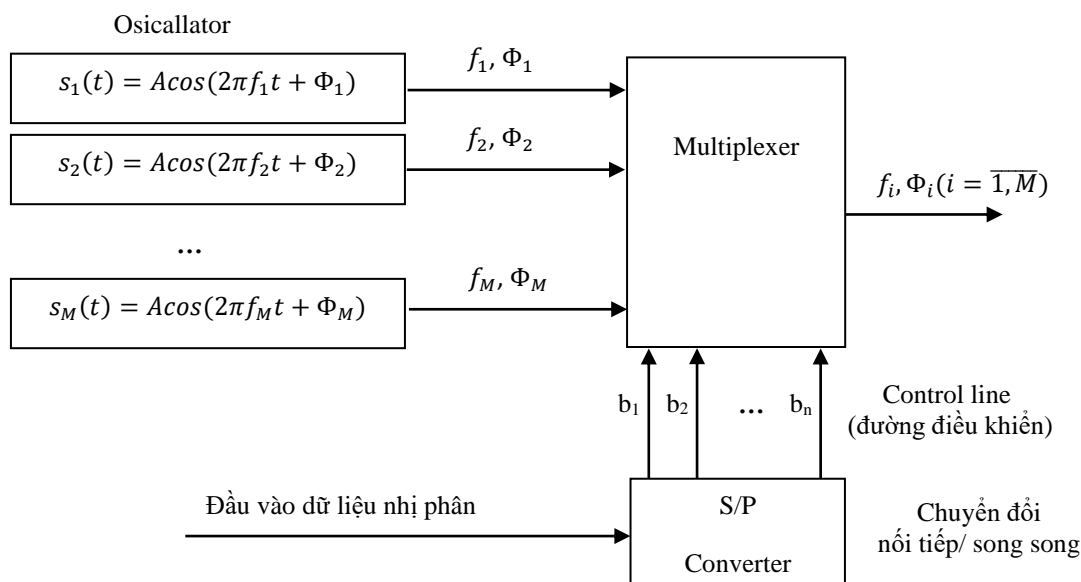


b. Kỹ thuật điều chế Non Coherent MFSK.

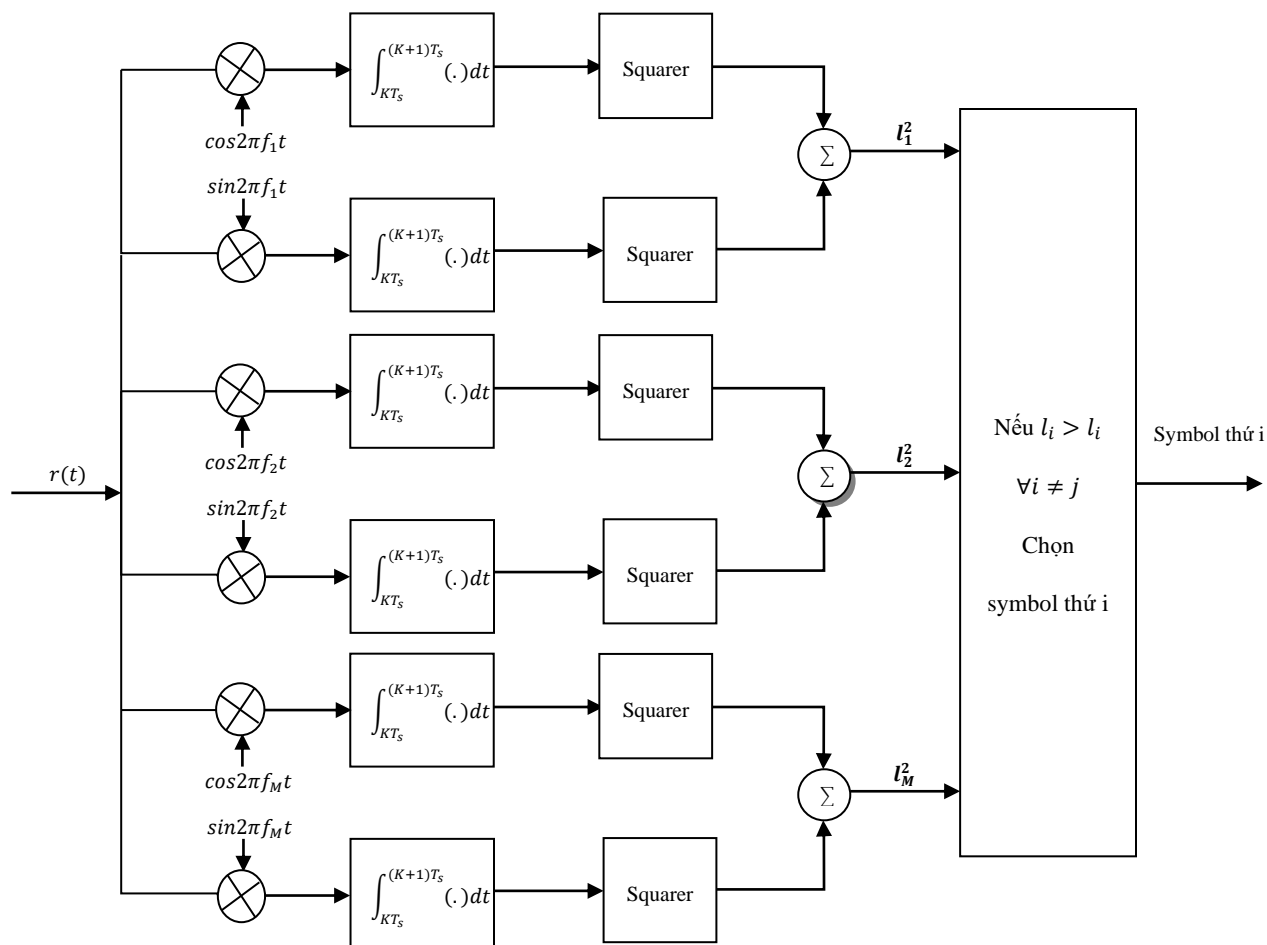
Tín hiệu trên đường truyền.

$$\begin{cases} s_i(t) = A \cos(2\pi f_i t + \Phi_i) \\ i = \overline{1, M} \end{cases} \quad KT_s \leq t \leq (K+1)T_s. \text{ Biểu diễn symbol thứ } i$$

- Bộ điều chế:



- Bộ giải điều chế: Giả sử tín hiệu đầu vào của bộ giải điều chế là:  $r(t)$ .



## 2.3 Kỹ thuật điều chế PSK (Phase Shift Keying)

### 2.3.1 Kỹ thuật điều chế BPSK (Binary PSK).

Dữ liệu nhị phân được biểu diễn bằng 2 tín hiệu có pha khác nhau.

$$\begin{cases} s_1(t) = A\cos(2\pi f_c t + \Phi_1); 0 \leq t \leq T. \text{ biểu diễn bit 1} \\ s_2(t) = A\cos(2\pi f_c t + \Phi_2); 0 \leq t \leq T. \text{ biểu diễn bit 0} \end{cases}$$

Trong đó:

T: độ rộng bit.

$\Phi_1, \Phi_2$  là 2 góc pha ban đầu.

$$\text{Thực tế lấy } \Phi_1 = 0, \Phi_2 = \pi. \Rightarrow \begin{cases} s_1(t) = A\cos 2\pi f_c t \\ s_2(t) = -A\cos 2\pi f_c t \end{cases}$$

Trường hợp tổng quát:

$$\begin{aligned} s_i(t) &= A\cos(2\pi f_c t + \Phi_i); i = (1, 2) \\ &= A\cos 2\pi f_c t \cdot \cos \Phi_i - A\sin 2\pi f_c t \cdot \sin \Phi_i \\ &= s_{i1}\Phi_1(t) + s_{i2}\Phi_2(t) \end{aligned}$$

Trong đó:

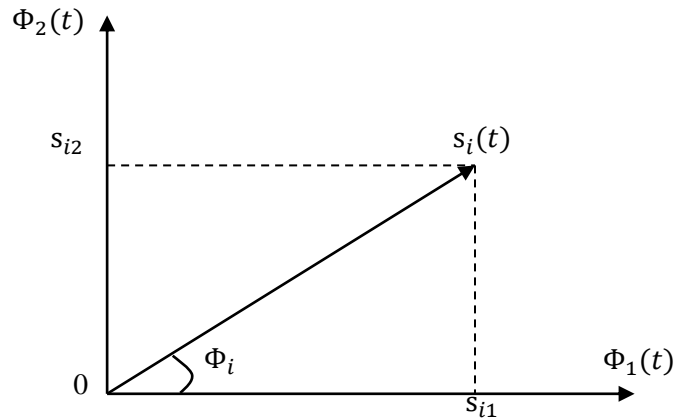
$$s_{i1} = \sqrt{\frac{T_b}{2}} A\cos \Phi_i = \sqrt{E}\cos \Phi_i$$

$$s_{i2} = \sqrt{\frac{T_b}{2}} A\sin \Phi_i = \sqrt{E}\sin \Phi_i$$

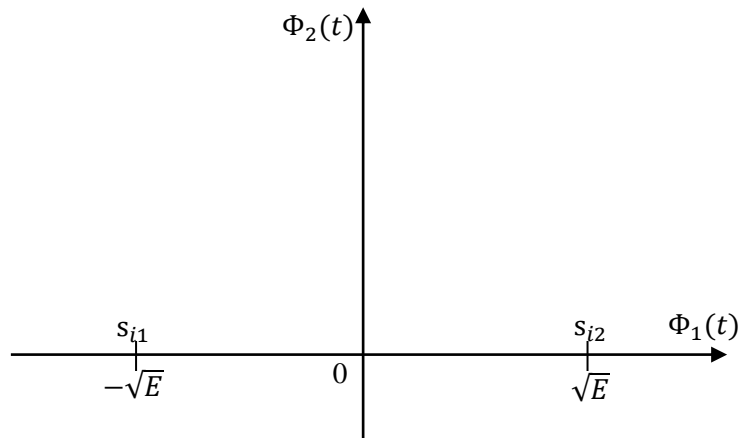
$$\Rightarrow E = \frac{A^2 T_b}{2}. \text{ Năng lượng của tín hiệu.}$$

$$\Phi_1(t) = \sqrt{\frac{2}{T_b}} \cos 2\pi f_c t; 0 \leq t \leq T_b; \quad \Phi_2(t) = -\sqrt{\frac{2}{T_b}} \sin 2\pi f_c t; 0 \leq t \leq T_b$$

Từ đây có thể biểu diễn  $s_i(t)$  trên hệ trục tọa độ như sau:

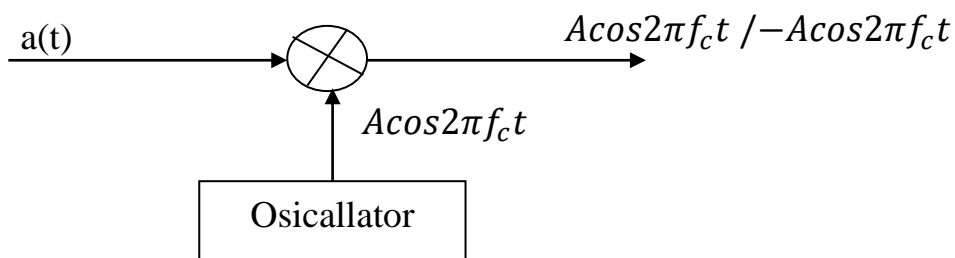


Trường hợp,  $\Phi_1(t)=0$ ;  $\Phi_2(t)=\pi$  ta có:

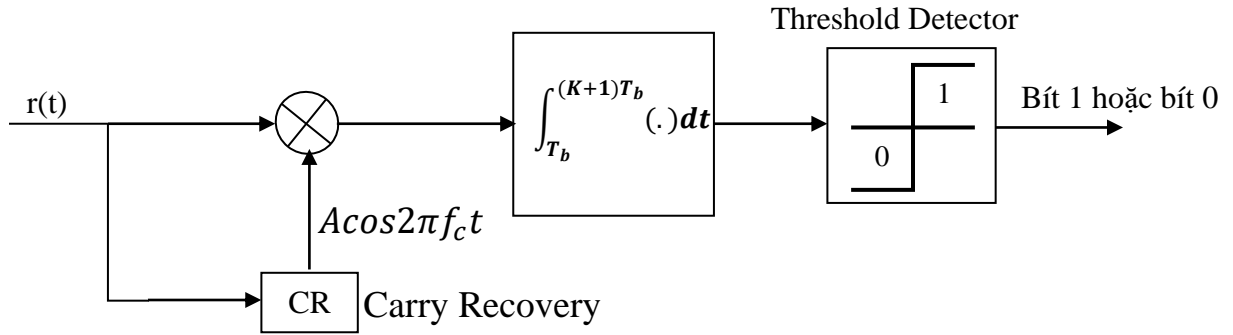


Từ đây ta có:  $s_i(t) = \sqrt{E} \cos \Phi_i \Phi_1(t) + \sqrt{E} \sin \Phi_i \Phi_2(t)$

- Bộ điều chế:



- Bộ giải điều chế BPSK.



$$\int_{KT_b}^{(K+1)T_b} Aa(t)\cos 2\pi f_c t \cdot \cos 2\pi f_c t \cdot dt = \begin{cases} A \text{ khi } a(t) = 1 \\ -A \text{ khi } a(t) = -1 \end{cases}$$

### 3.3.2 Kỹ thuật điều chế M-PSK (M-arry PSK)

Trong M-PSK dòng dữ liệu được chia thành các Symbol, mỗi symbol có  $n = \log_2 M$  (bít)  $\Rightarrow$  hiệu quả băng thông được tăng lên  $n$  lần.

Tập tín hiệu MPSK được biểu diễn như sau:

$$s_i(t) = A \cos(2\pi f_c t + \Phi_i) \text{ với } 0 \leq t \leq KT_s; i = \overline{1, M}.$$

Trong đó:

$f_c$ : tần số sóng mang.

$T_s$ : độ rộng của symbol.

$\Phi_i$ : góc pha ban đầu  $\Phi_i = \frac{2i-1}{M} \cdot \pi$

Từ phương trình trên ta có:

$$\begin{aligned} s_i(t) &= A \cos(2\pi f_c t + \Phi_i) \\ &= A \cos 2\pi f_c t \cos \Phi_i - A \sin 2\pi f_c t \sin \Phi_i \end{aligned}$$

$$= s_{i1}\Phi_1(t) + s_{i2}\Phi_2(t)$$

Với  $\Phi_1(t)$  và  $\Phi_2(t)$  là các hàm trực giao và:

$$s_{i1} = \int_0^{T_s} s_i(t)\Phi_1(t).dt$$

$$s_{i2} = \int_0^{T_s} s_i(t)\Phi_2(t).dt$$

\* Tính  $s_{i1}$ :

$$s_{i1} = \int_0^{T_s} s_i(t)\Phi_1(t).dt = \sqrt{E}. \cos\Phi_i$$

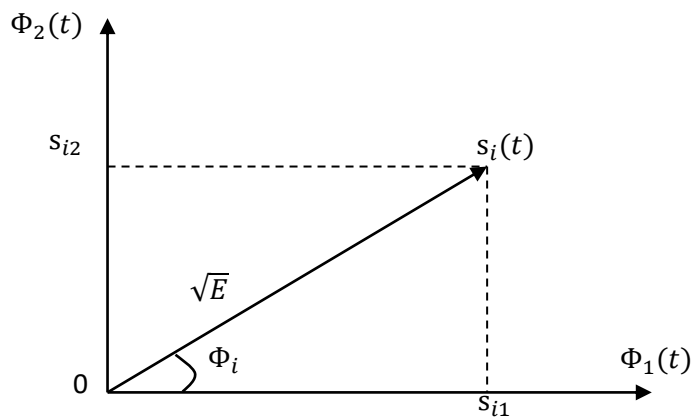
Với  $E = \frac{A^2 T_s}{2}$ . Năng lượng của tín hiệu.

Tương tự ta có:  $s_{i2} = \sqrt{E}\sin\Phi_i$

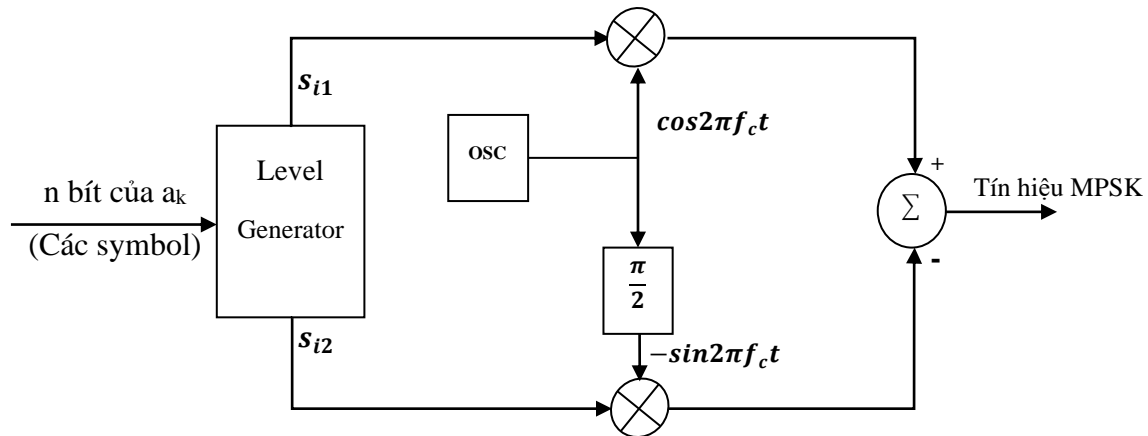
Nên:  $s_i(t) = \sqrt{E}\cos\Phi_i\Phi_1(t) + \sqrt{E}\sin\Phi_i\Phi_2(t)$

Với:  $\Phi_i = \arctg \frac{s_{i1}}{s_{i2}}$

- Ta có thể biểu diễn  $s_i(t)$  trên hệ trục tọa độ.

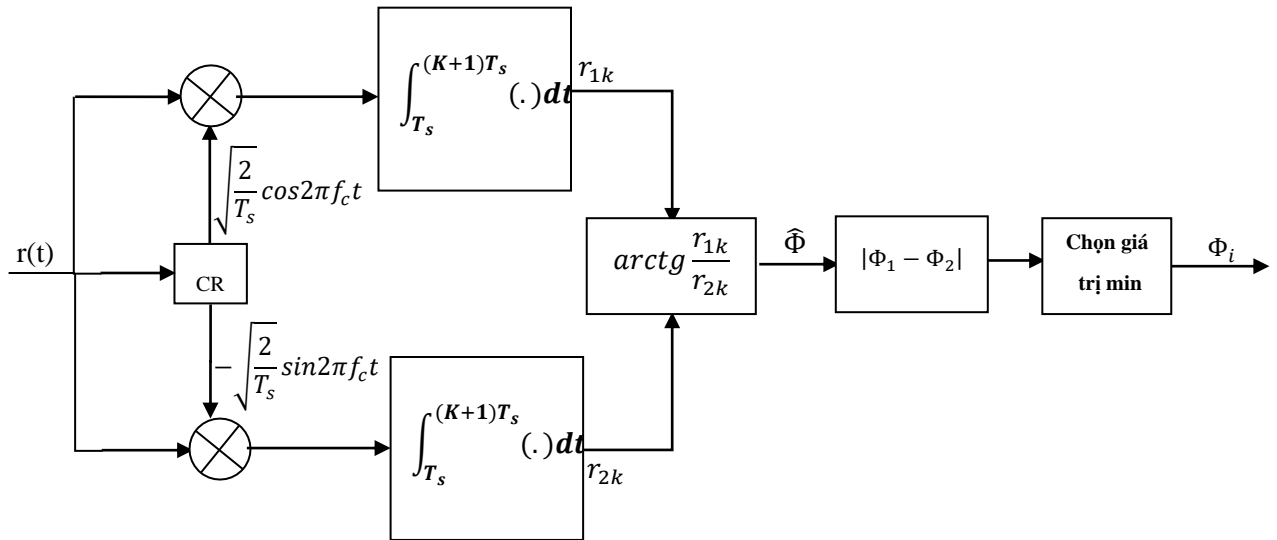


- Bộ điều chế MPSK.



- Bộ giải điều chế:

Giả sử  $r(t)$  là tín hiệu thu được, ta có bộ giải điều chế MPSK





## CHƯƠNG IV GIAO TIẾP KẾT NỐI SỐ LIỆU

### 1. Truyền nối tiếp bất đồng bộ

#### 1.1 Nguyên tắc truyền nối tiếp bất đồng bộ

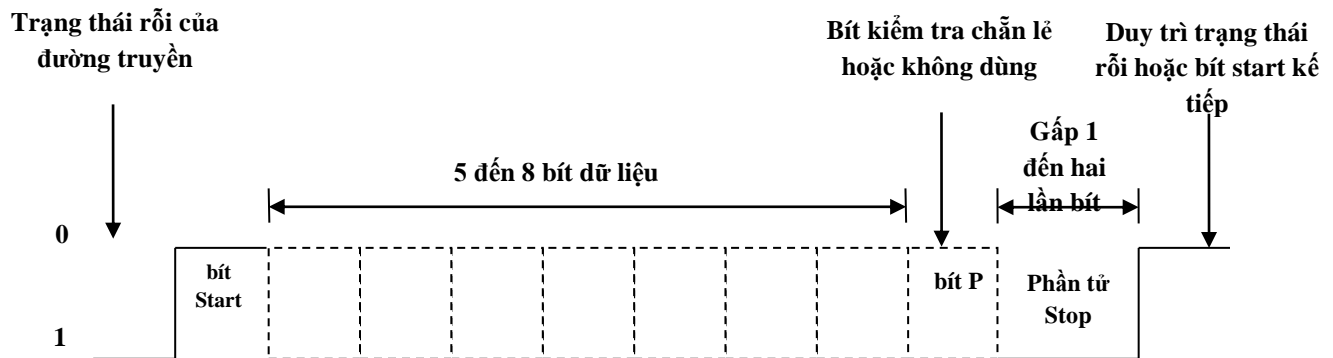
- Để tạo sự đồng bộ giữa máy phát và máy thu cần có quy định cụ thể về mẫu tín hiệu trong hệ thống bất đồng bộ như sau:

+ Mỗi ký tự gồm một số bit gọi là ký tự dữ liệu, có thể từ 5÷8 bit (tùy thuộc vào loại mã được sử dụng để mã hóa thông tin). Ngoài ra, để tạo sự đồng bộ, kèm theo các bit mã ký tự còn có:

+ Bit START ở đầu mỗi ký tự (Bit 0): số lượng 1 bit.

+ Bit STOP ở cuối mỗi ký tự (Bit 1): số lượng 1; 1,5; 2 bit.

+ Nếu có thêm bit chẵn lẻ PARITY BIT thì bit này nằm trước bit STOP.



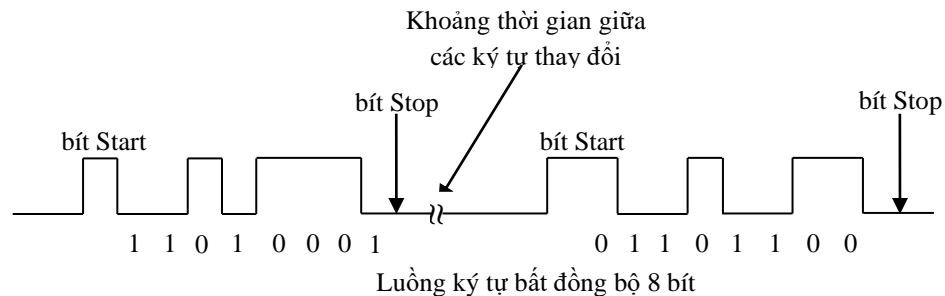
+ Khoảng thời gian giữa hai ký tự liên tiếp là ngẫu nhiên, trong khoảng thời gian này các thiết bị ở trạng thái nghỉ không truyền dữ liệu, và trạng thái này còn được gọi là trạng thái nghỉ. Ở trạng thái nghỉ (Idle state) máy phát luôn phát đi bit 1 gọi là bit nghỉ (Idle bit)

+ Máy thu dò ra bit START khi có sự chuyển đổi trạng thái 1→0, sau đó lấy mẫu từ 5÷8 khoảng kế tiếp (tùy theo số bit trong ký tự).

+ Kiểm tra trạng thái 1 ở cuối khối ký tự (bit STOP). Chờ việc chuyển trạng thái từ 1→0 đồng bộ cho ký tự tiếp theo. Đối với dòng dữ liệu đều, khoảng cách giữa các ký tự là đồng nhất nên:

+ Chỉ cần giữ đồng bộ trong một ký tự.

+ Tái đồng bộ cho các ký tự tiếp theo.



- Ưu điểm: Đơn giản, Rẻ tiền

- Nhược điểm: Hiệu suất thấp

- Ứng dụng

Dùng trong truyền ký tự giữa bàn phím và máy tính. Ngoài ra, truyền nối tiếp bất đồng bộ được sử dụng để truyền khối ký tự giữa hai máy tính.

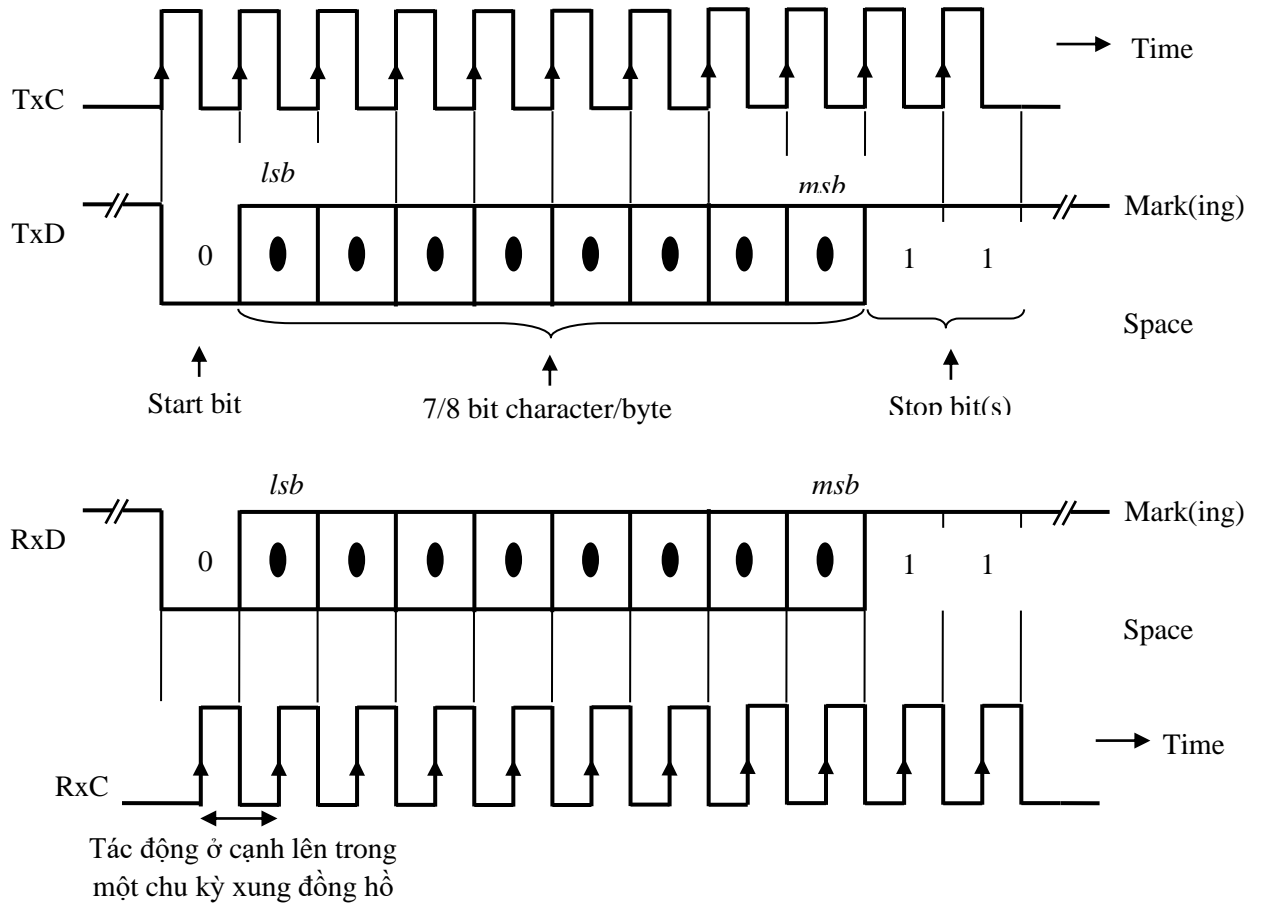
## 1.2 Nguyên tắc đồng bộ bit

- Chuyển đổi byte (ký tự) thông tin thành/từ chuỗi bit được thực hiện bởi thanh ghi PISO/SIPO.

- Trong truyền bất đồng bộ, đồng hồ của thiết bị thu chạy không đồng bộ với tín hiệu thu.

- Để xử lý thu hiệu quả, cần dùng đồng hồ thu để lấy mẫu tín hiệu đến ngay điểm giữa thời của bit dữ liệu (điểm giữa của thời gian).

- Để đạt được điều này, tín hiệu đồng hồ thu phải nhanh gấp N lần đồng hồ phát vì mỗi bit được dịch vào thanh ghi SIPO sau N chu kỳ xung đồng hồ.



Sự chuyển trạng thái từ 1 xuống 0 là dấu hiệu của bit start, có nghĩa là điểm bắt đầu của một ký tự và chúng được dùng để khởi động bộ đếm xung clock ở máy thu. Mỗi bit bao gồm cả bit start, được lấy mẫu tại khoảng giữa của thời bit ngay sau khi phát hiện.

Bit start được lấy mẫu sau N/2 chu kỳ xung clock (giữa sườn xuống của xung), tiếp tục lấy mẫu sau mỗi N chu kỳ xung clock tiếp theo cho mỗi bit trong ký tự (sườn xuống của xung tiếp theo).

Cần lưu ý rằng, đồng hồ thu chạy bất đồng bộ với tín hiệu đến, do đó các vị trí tương đối của hai tín hiệu (tín hiệu start và bit ký tự) có thể ở bất kỳ vị trí nào

trong một chu kỳ của xung đồng hồ thu (vị trí bất kỳ của sườn xuống), với N càng lớn thì vị trí lấy mẫu có khuynh hướng gần giữa thời bit hơn. Do vậy chế độ truyền tin này có tốc độ truyền không thể cao được.

## 2. Truyền nối tiếp đồng bộ

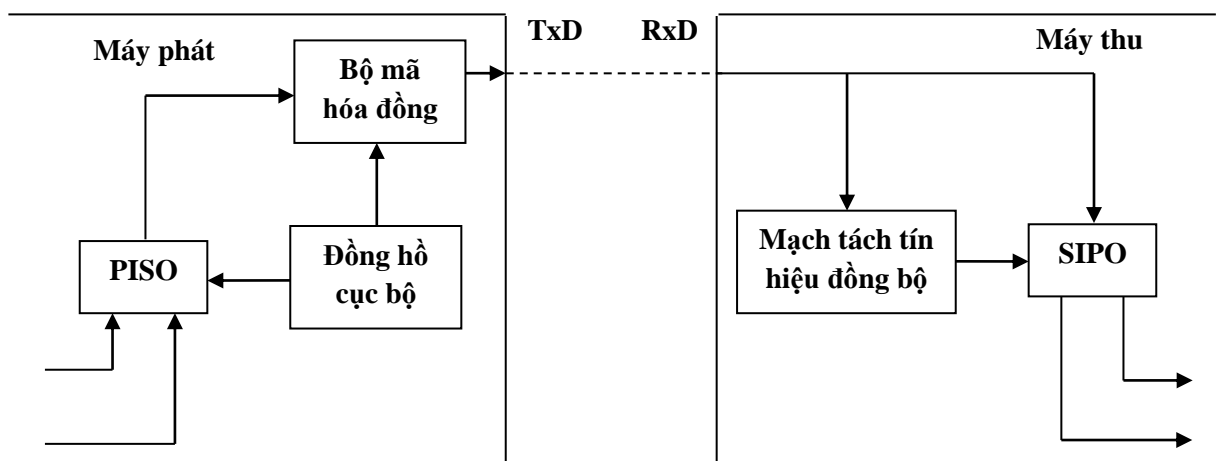
### 2.1 Nguyên tắc đồng bộ bit.

- Sự khác nhau cơ bản của truyền bất đồng bộ và đồng bộ là đối với truyền bất đồng bộ đồng hồ thu chạy bất đồng bộ với tín hiệu đến, còn truyền đồng bộ thì đồng hồ thu chạy đồng bộ với tín hiệu đến.

- Trong đó truyền đồng bộ, các bit START, STOP không được dùng thay vào đó, mỗi khung tin được truyền như dòng liên tục các ký tự số nhị phân.

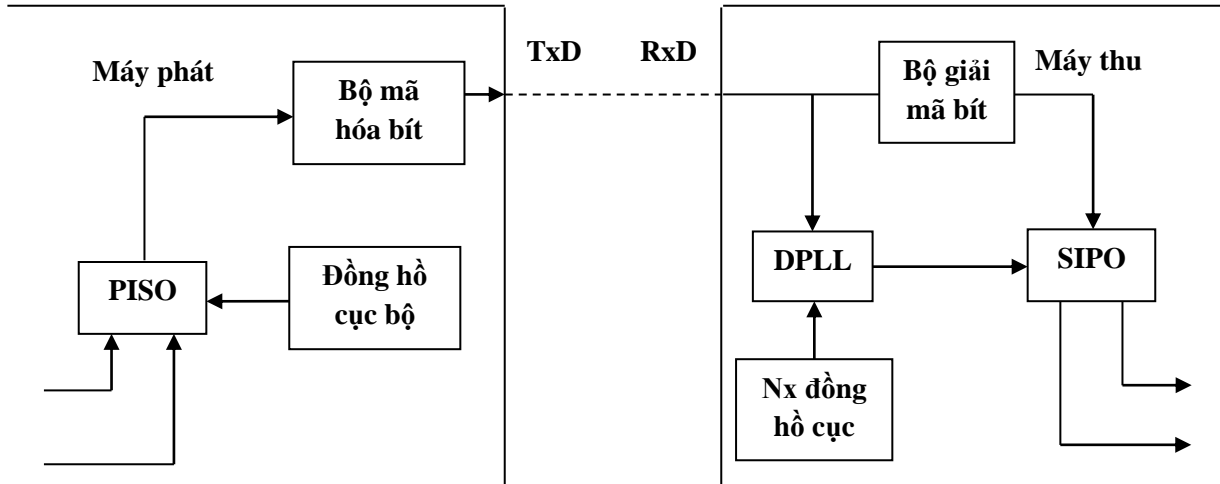
- Máy thu đồng bộ bit trong hai cách:

+ Cách thứ nhất là thông tin định thời được nhúng vào trong tín hiệu truyền đi và sau đó được tách ra bởi máy thu.



+ Cách thứ hai là máy thu có một đồng hồ cục bộ được giữ đồng bộ với tín hiệu thu nhờ thiết bị vòng khóa pha số (DPLL\_Digital Phase Lock Loop)

- Ta thấy rằng, DPLL lợi dụng sự chuyển trạng thái từ 1 sang 0 hay từ 0 sang 1 trong tín hiệu thu để duy trì sự đồng bộ qua một khoảng thời gian định kỳ nào đó.



## 2.2 Truyền đồng bộ hướng ký tự.

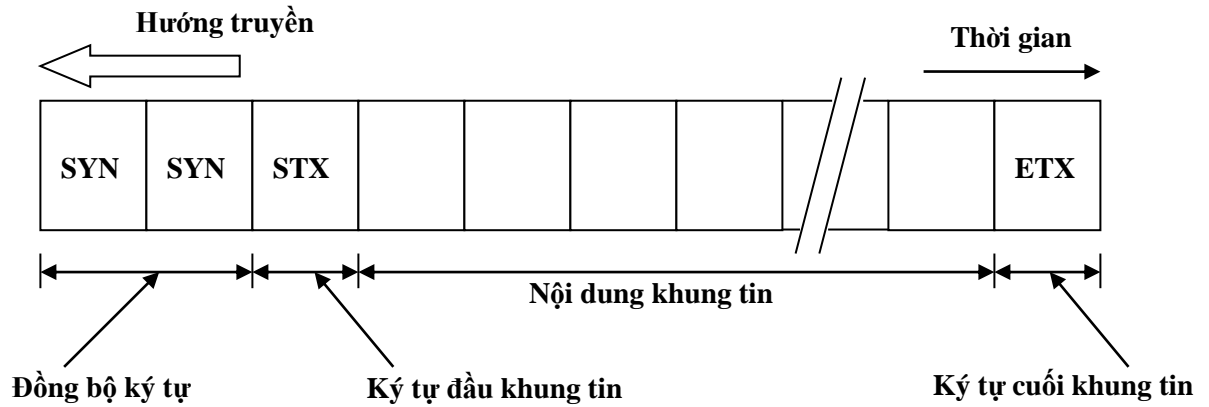
- Truyền đồng bộ hướng ký tự được dùng chủ yếu để truyền khối ký tự, như là các tập tin định dạng text. Vì không có bit START, STOP nên cần phải có cách thức để đồng bộ ký tự.

- Để thực hiện đồng bộ, máy phát thêm vào các ký tự điều khiển gọi là các ký tự điều khiển SYN, ngay trước các khối ký tự truyền.

- Các ký tự điều khiển này phải có hai chức năng:

+ Trước hết các ký tự điều khiển cho máy thu duy trì đồng bộ bit.

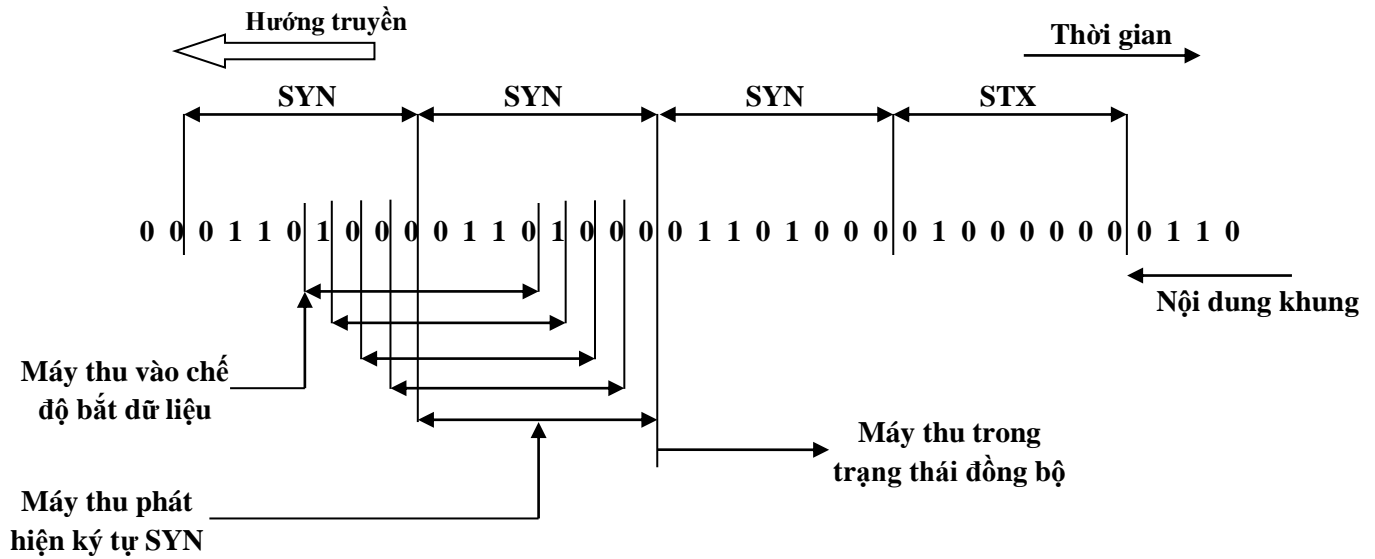
+ Thứ hai, khi điều khiển đã được thực hiện, các ký tự điều khiển cho phép máy thu bắt đầu biên dịch luồng bit chính xác theo các ranh giới ký tự - sự đồng bộ ký tự.



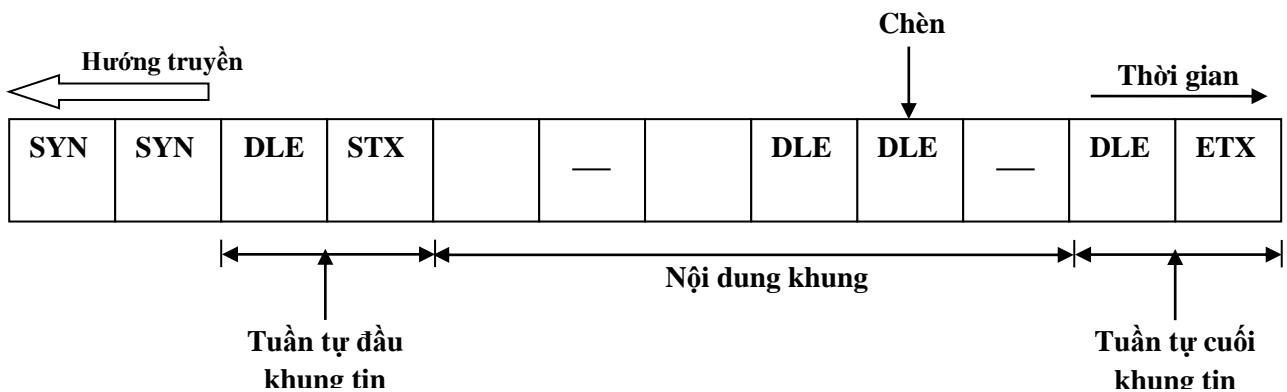
- Đồng bộ khung thực hiện giống như truyền bất đồng bộ bằng cách đóng gói khối ký tự giữa cặp ký tự điều khiển truyền STX, EXT. Các ký tự điều khiển SYN thường được dùng bởi bộ thu để đồng bộ ký tự thì đứng trước ký tự STX (start of Text).

- Khi máy thu được đồng bộ bit thì nó sẽ chuyển vào chế độ làm việc gọi là chế độ bắt số liệu. Khi bộ thu vào chế độ bắt số liệu, nó bắt đầu dịch dòng bit trong một cửa sổ 8 bit khi tiếp nhận một bit mới. Bằng cách này, khi nhận được mỗi bit, bộ thu kiểm tra xem 8 bit sau cùng có đúng bằng ký tự đồng bộ không. Nếu không bằng ký tự đồng bộ, bộ thu sẽ tiếp tục thu bit kế tiếp và lặp lại thao tác kiểm tra này. Trong trường hợp bằng với ký tự đồng bộ, các ký tự tiếp theo được đọc sau mỗi 8 bit thu được.

- Khi ở trạng thái đồng bộ ký tự, máy thu bắt đầu xử lý mỗi ký tự thu nối tiếp để dò ra ký tự STX đầu khung tin. Khi phát hiện ra một STX, máy thu xử lý nhận nội dung khung tin và chỉ kết thúc công việc này khi phát hiện ra ký tự EXT. Trên một liên kết điểm - điểm, thông thường máy thu sẽ quay trở lại truyền các ký tự SYN để máy thu duy trì cơ cấu đồng bộ. Các thủ tục trên được lặp lại mỗi khi truyền một khung tin mới.



- Khi dữ liệu nhị phân đang được truyền, sự trong suốt dữ liệu đạt được tương tự như phương pháp đồng bộ khung đã trình bày ở trên. Nghĩa là, dùng một ký tự DLE chèn vào trước STX và EXT đồng thời chèn một DLE vào bất cứ vị trí nào trong nội dung có chứa DLE. Trường hợp này, các ký tự SYN đứng trước ký tự DLE đầu tiên.



### 2.3 Truyền đồng bộ hướng bit

- Bắt đầu và kết thúc bằng một cờ “0111 1110”. Sở dĩ có thuật ngữ “hướng bit” là vì luồng thu dò từng bit.

- Do đó, về nguyên lý nội dung của khung tin nhất thiết phải là bội số của 8.

- Để cho máy thu có thể tiếp cận và duy trì cơ cấu đồng bộ بیت, máy phát phải gửi một chuỗi các byte rồi 0111 1111 đứng trước cờ bắt đầu khung.

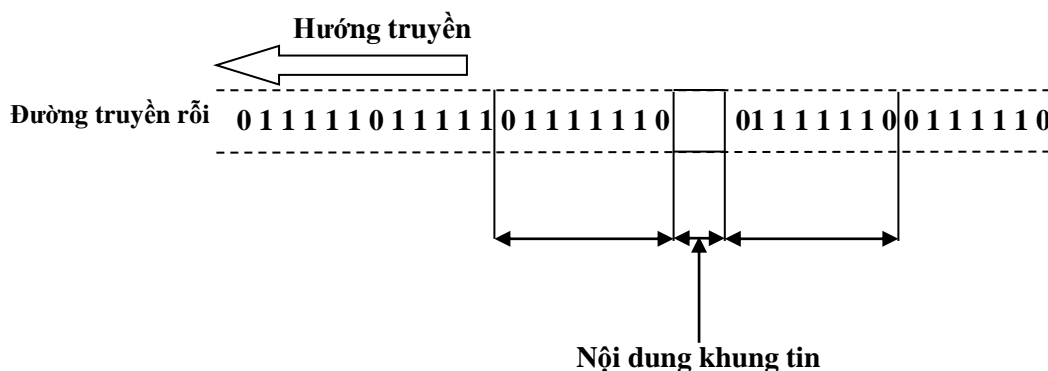
- Dùng bộ mã NRZI mã hóa بیت 0 trong byte rồi cho phép DPLL tại máy thu tiếp cận và duy trì sự đồng bộ đồng hồ. Khi nhận được cờ khởi đầu, khung tin, nội dung của khung tin được đọc và dịch theo các khoảng 8 بیت cho đến khi gặp cờ kết thúc khung tin.

- Để đạt được tính trong suốt dữ liệu, cần đảm bảo rằng cờ không bị nhận dạng nhầm với nội dung của khung tin.

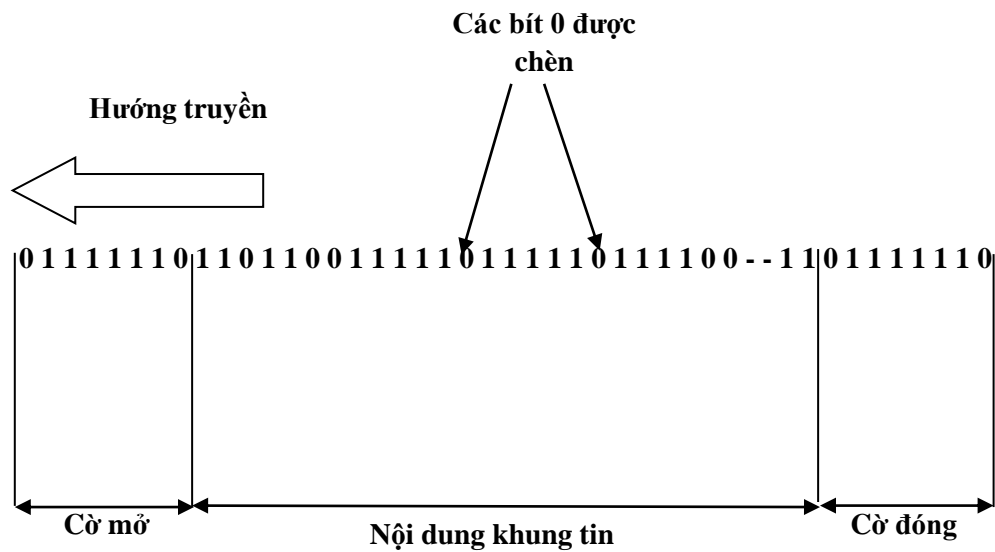
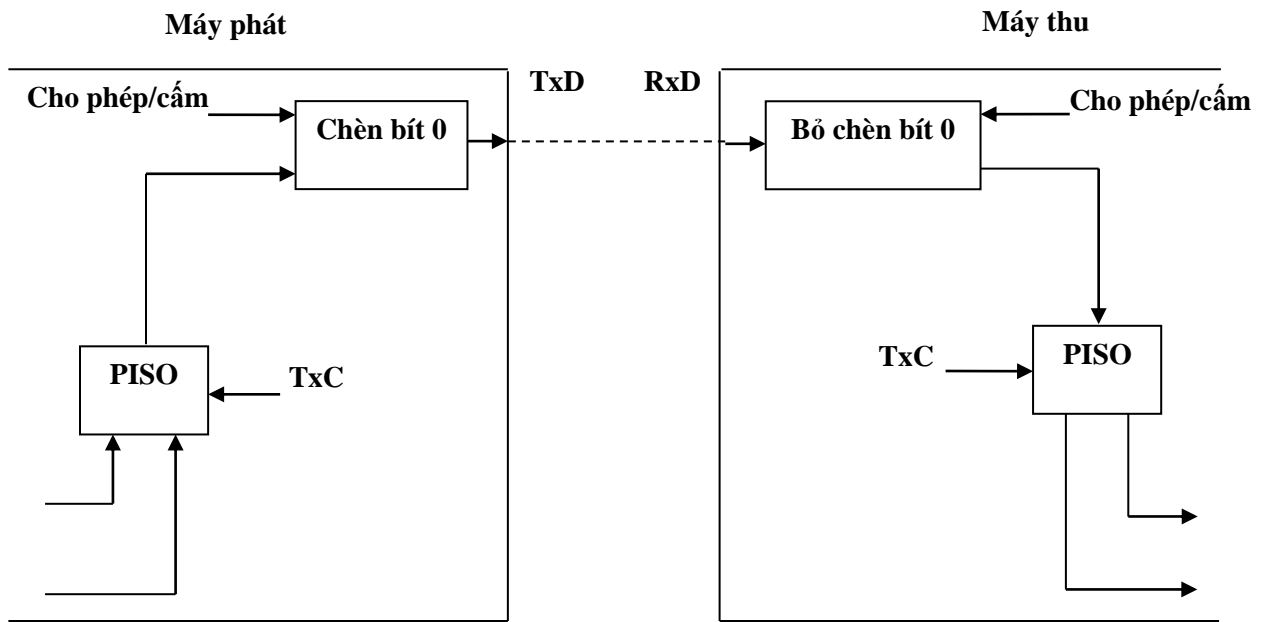
- Để giải quyết vấn đề này người ta sử dụng kỹ thuật tạo khung dùng بیت độn. mạch thực hiện chức năng này được đặt tại đầu ra của thanh ghi PISO. Mạch này chỉ hoạt động trong quá trình truyền nội dung của khung tin.

- Khi phát hiện thấy có 5 بیت 1 liên tiếp, nó sẽ tự động chèn vào 1 بیت 0. Bằng cách này, sẽ không bao giờ có cờ trong phần nội dung truyền đi.

- Một mạch tương tự tại máy thu nằm ngay trước đầu vào thanh ghi PISO để thực hiện chức năng gỡ bỏ بیت 0.



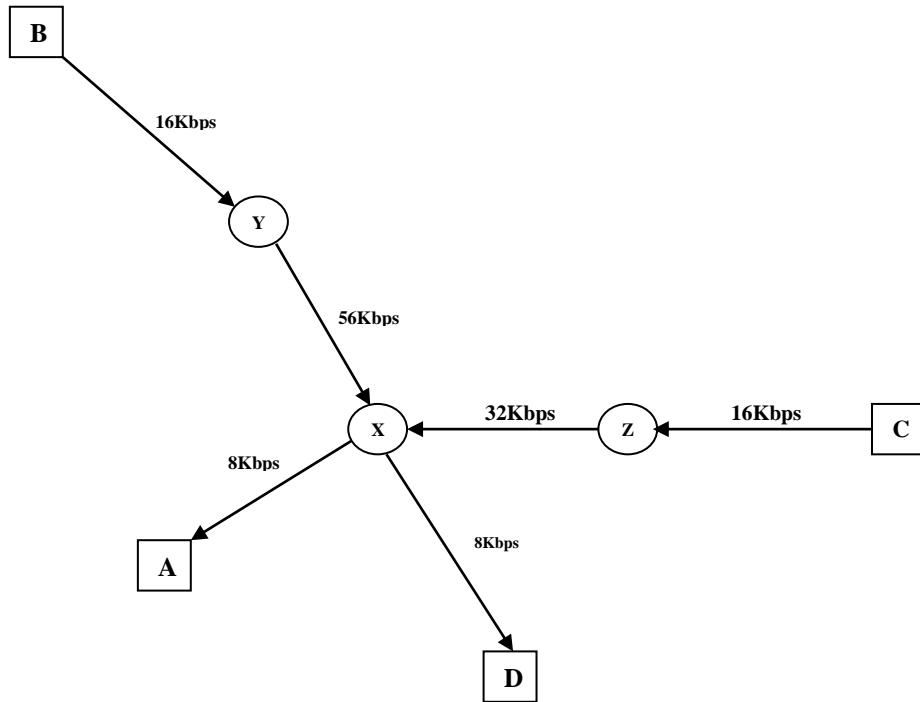




## CHƯƠNG V. ĐIỀU KHIỂN LƯỒNG VÀ KIỂM SOÁT LỖI.

### 1. Phân tích các trường hợp trong mạng truyền số liệu

Xét mô hình mạng truyền thông số liệu sau:



- Mạng 2 kết nối:
  - $B \rightarrow A$  ( $B \rightarrow Y \rightarrow X \rightarrow A$ ).
  - $C \rightarrow D$  ( $C \rightarrow Z \rightarrow X \rightarrow D$ ).

Gọi  $\lambda_{BA}$  là tốc độ truyền từ  $B \rightarrow A$

Gọi  $\lambda_{CD}$  là tốc độ truyền từ  $C \rightarrow D$

Giả sử mạng không kiểm soát, các gói tin điều khiển có thể truy nhập vào mạng và bộ đệm tại các nút X, Y, Z đều có thể được sử dụng bởi bất kỳ gói tin nào.

Giả sử môi trường truyền không có lỗi, lúc này các gói tin không bị sai nhưng có thể phải truyền lại nếu gói tin bị nút mạng hủy do tràn bộ nhớ đệm (do không còn dung lượng để lưu gói tin tạm thời trước khi xử lý).

Giả thiết rằng khi gói tin bị mất vì không được lưu trữ tạm thời phía phát phải thực hiện phát lại nhằm đảm bảo việc truyền tin tin cậy.

Để minh họa việc truyền tin trong mạng ta tìm hiểu các trường hợp sau:

- Trường hợp 1:  $\lambda_{BA} = 7$  Kbps và  $\lambda_{CD} = 0$

Trong trường hợp này, không xảy ra tắc nghẽn vì lưu lượng từ B→A sẽ được mạng trung chuyển hết. Tốc độ thông tin từ trạm A chính bằng tốc độ thông tin mà nút B đưa vào mạng, các đường B→Y, Y→X, X→A đều có tốc độ 7 Kbps.

- Trường hợp 2:  $\lambda_{BA} = 8 + \delta$  Kbps, ( $\delta > 0$ ) và  $\lambda_{CD} = 0$

Trong trường hợp này, tốc độ thông tin từ B→A lớn hơn tốc độ hoạt động từ X→A. Lượng thông tin dư thừa sẽ được lưu vào bộ đệm của X. Bộ đệm của X sẽ dần đầy và bị tràn dẫn đến các gói tin từ Y đến sẽ không được lưu và bị hủy. Vì bộ đệm của Y sẽ lưu lại các gói tin chưa được báo nhận (dùng để truyền lại) nên bộ đệm Y sẽ dần đầy và bị tràn. Nút X có thể chuyển 8 Kbps khi lưu lượng đầu vào của nó là  $8 + \delta$  Kbps ( X hủy  $\delta$  Kbps ). Lúc này đường Y→X sẽ có tốc độ là  $8 + 2\delta$  Kbps ( $8 + \delta$  Kbps là thông tin từ B đến,  $\delta$  Kbps là thông tin phát lại ). Nhưng nút X chỉ có thể truyền 8 Kbps nên nó sẽ hủy  $2\delta$  Kbps và Y lại phải truyền lại lượng thông tin này. Quá trình này cứ tiếp diễn và cuối cùng đường nối từ Y→X sẽ hoạt động với tốc độ 56 Kbps. Tương tự như vậy đường B→Y cũng sẽ hoạt động với tốc độ 16 Kbps.

Để giải quyết vấn đề này có 2 cách thực hiện:

- Xây dựng hệ thống mạng có khả năng đáp ứng được tốc độ thông tin từ  $X \rightarrow A$  ( $8 + \delta$  Kbps) nhằm đáp ứng với yêu cầu tốc độ của B. Giải pháp này chỉ khả thi và hiệu quả khi tốc độ phát tin của B là ổn định trong 1 thời gian dài, nếu không thì hiệu quả sử dụng tài nguyên là rất thấp, vì nếu xây dựng hệ thống mạng có khả năng đáp ứng lưu lượng lớn nhưng chỉ hoạt động với yêu cầu trao đổi lưu lượng nhỏ.

- Giới hạn tốc độ truyền tin của B xuống còn 8 Kbps. Phương án này là khả thi, khi yêu cầu truyền tin của B trong phần lớn thời gian là nhỏ 8 Kbps và tốc độ  $8 + \delta$  Kbps chỉ diễn ra trong thời gian ngắn.

Trong 2 phương án này, thực tế người ta sử dụng phương án 2 với sự hỗ trợ của các giao thức mạng.

- Trường hợp 3:  $\lambda_{BA} = 7$  Kbps và  $\lambda_{CD} = 7$  Kbps.

Tương tự như trường hợp 1, trường hợp 3 không xảy ra tắc nghẽn trong mạng thông tin được truyền dẫn đến A và D với tốc độ 7 Kbps cho mỗi nút. Mỗi 1 liên kết trong mạng sẽ hoạt động với tốc độ 7 Kbps.

- Trường hợp 4:  $\lambda_{BA} = 8 + \delta$  Kbps và  $\lambda_{CD} = 7$  Kbps.

Trong trường hợp này, đường đi từ  $C \rightarrow D$  có đủ dung lượng (tốc độ) để đáp ứng yêu cầu cho kết nối C-D, tuy nhiên, yêu cầu đường truyền thông tin từ  $B \rightarrow A$  vượt quá khả năng xử lý của tuyến truyền này. Trong trường hợp này, 2 liên kết  $B \rightarrow A$ ,  $C \rightarrow D$ , chia sẻ bộ đệm của nút X. Như ta đã xét ở trường hợp 2, lưu lượng thông tin  $B \rightarrow A$  sẽ làm tràn bộ đệm của X, điều này dẫn đến thông tin từ B đến A và C đến D đều bị hủy. Hiện tượng này xảy ra đối với tất cả các gói tin ( cả B và C) cho dù nguyên nhân gây ra do B. Hệ quả này là nút Y và nút Z cũng bị tràn bộ đệm và tất cả các liên kết sẽ hoạt động với tốc độ cực đại của chúng.

Do trước khi gửi gói tin từ B và C đến A và D tương ứng, nút X phải lưu các gói tin này vào bộ đệm để xử lý, nên trong trường hợp bộ đệm của X bị tràn, X phải

hủy các gói tin này. Do đó tốc độ thông tin từ  $Y \rightarrow X$  gấp đôi tốc độ  $Z \rightarrow X$  (khi các liên kết này hoạt động ở tốc độ đỉnh) nên số lượng các gói tin từ  $Y \rightarrow X$  gấp đôi  $Z \rightarrow X$ . Nói cách khác X sẽ hủy (hay chấp nhận) các gói từ Y và Z đến theo tỉ lệ 2:1.

Lúc này thông tin từ  $B \rightarrow A$  hoạt động với tốc độ 8 Kbps trong khi thông tin từ  $C \rightarrow D$  hoạt động với tốc độ 4 Kbps. So với trường hợp 3 ta thấy:

- Thông lượng tổng cộng của mạng giảm từ 14 Kbps xuống còn 12 Kbps.
- Nút C bị đối xử không công bằng, vì tốc độ truyền của nó đến D bị giảm từ 7 Kbps xuống còn 4 Kbps trong khi nút B không ảnh hưởng nhiều (giảm từ  $8+\delta$  Kbps xuống 8 Kbps) ngoài ra nguyên nhân gây ra tắc nghẽn là do nút B.

Để giải quyết vấn đề này, người ta có thể dành 1 phần dung lượng bộ nhớ đệm tại nút X cho các gói tin đến từ C. Việc dành trước tài nguyên này dường như trái với nguyên tắc chuyển mạch gói khi tài nguyên mạng được chia sẻ tới tất cả các nút và người dùng. Tuy nhiên, trong thực tế người ta có thể đánh đổi điều này để đảm bảo tính công bằng trong mạng.

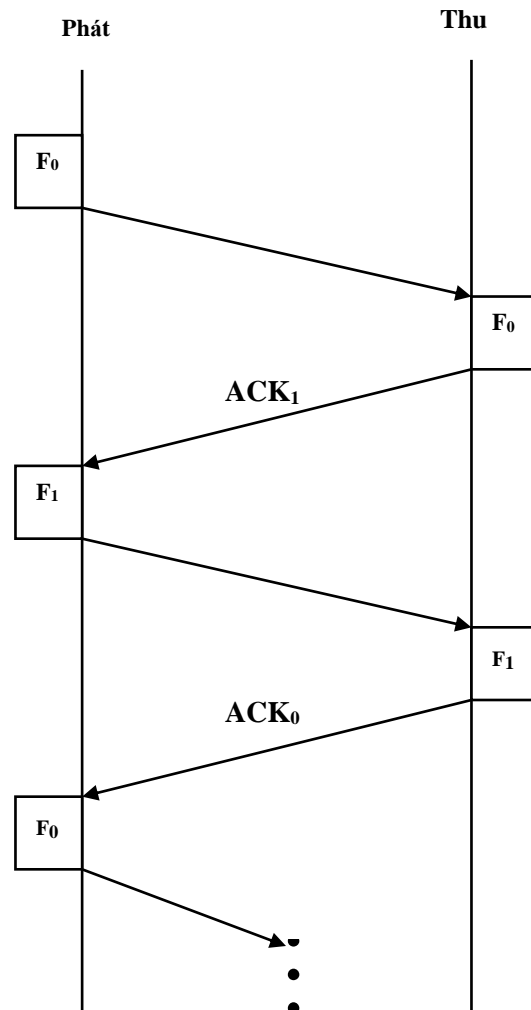
## 2. Điều khiển luồng.

### 2.1 Kỹ thuật điều khiển luồng theo kiểu dừng và đợi (*Stop and wait*).

#### a. Hoạt động.

- Phía phát, phát 1 khung tin sau đó dừng lại và đợi báo nhận từ phía thu (báo nhận từ phía thu là báo nhận ACK\_acknowledgement/ môi trường truyền không có lỗi).
- Khi phía thu nhận được 1 khung tin thì phía thu gửi lại cho phía phát 1 báo nhận ACK
- Khi phía phát nhận ACK, phía phát phát khung tin tiếp theo sau đó dừng lại và đợi báo nhận từ phía thu.

- Quá trình truyền được diễn ra tương tự cho đến khi phía phát phát hết khung tin.

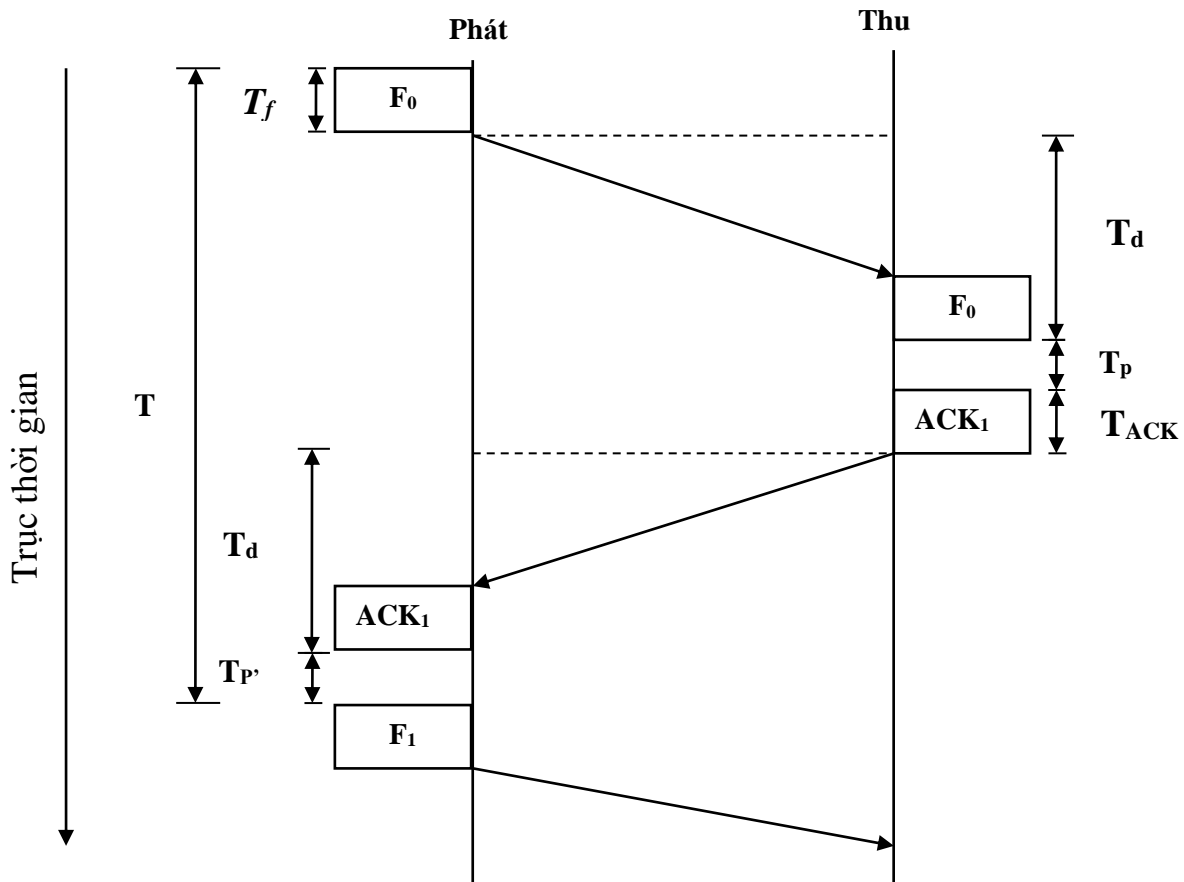


b. Hiệu suất:  $\eta_{saw}$

- Định nghĩa hiệu suất.

Hiệu suất truyền tin giữa bên phát và bên thu là tỉ lệ thời gian phía phát phát xong 1 lượng thông tin trên tổng thời gian để truyền hết lượng tin đó.

- Tính hiệu suất:  $\eta_{saw}$



$T_f$  là thời gian phát 1 khung tin.

$T_d$  là thời gian trễ truyền dẫn từ phía phát tới phía thu

$T_p$  là thời gian xử lý 1 khung tin

$T_{p'}$  là thời gian xử lý 1 ACK

$T_{ACK}$  là thời gian phát 1 ACK

$T$  là tổng thời gian để truyền xong 1 khung tin được tính từ lúc bắt đầu phát khung tin  $F_0$  đến khi bắt đầu phát khung tin  $F_1$ . Ta có:

$$T = T_f + T_d + T_p + T_{ACK} + T_d + T_{p'}$$

$$= T_f + 2T_d + T_p + T_{ACK} + T_{p'}$$

Theo định nghĩa hiệu suất ta có:

$$\eta_{saw} = \frac{T_f}{T} = \frac{T_f}{T_f + 2T_d + T_p + T_{ACK} + T_{p'}}$$

Do:  $T_f \gg T_p, T_{ACK}, T_{p'}$  nên:

$$\eta_{saw} = \frac{T_f}{T_f + 2T_d} = \frac{1}{1 + 2a}$$

Với:

$$a = \frac{T_d}{T_f} \text{ Mà : } T_f = \frac{l}{R}; \quad T_d = \frac{d}{v} \Rightarrow a = \frac{dR}{vl}$$

Trong đó:

$l$  là độ dài khung tin (bit)

$R$  tốc độ truyền tin qua kênh (bps)

$d$  là cự ly truyền giữa 2 trạm (m)

$v$  là vận tốc truyền sóng điện từ (m/s).

## **2.2 Kỹ thuật điều khiển luồng theo kiểu cửa sổ trượt (Sliding Window).**

### **a. Hoạt động.**

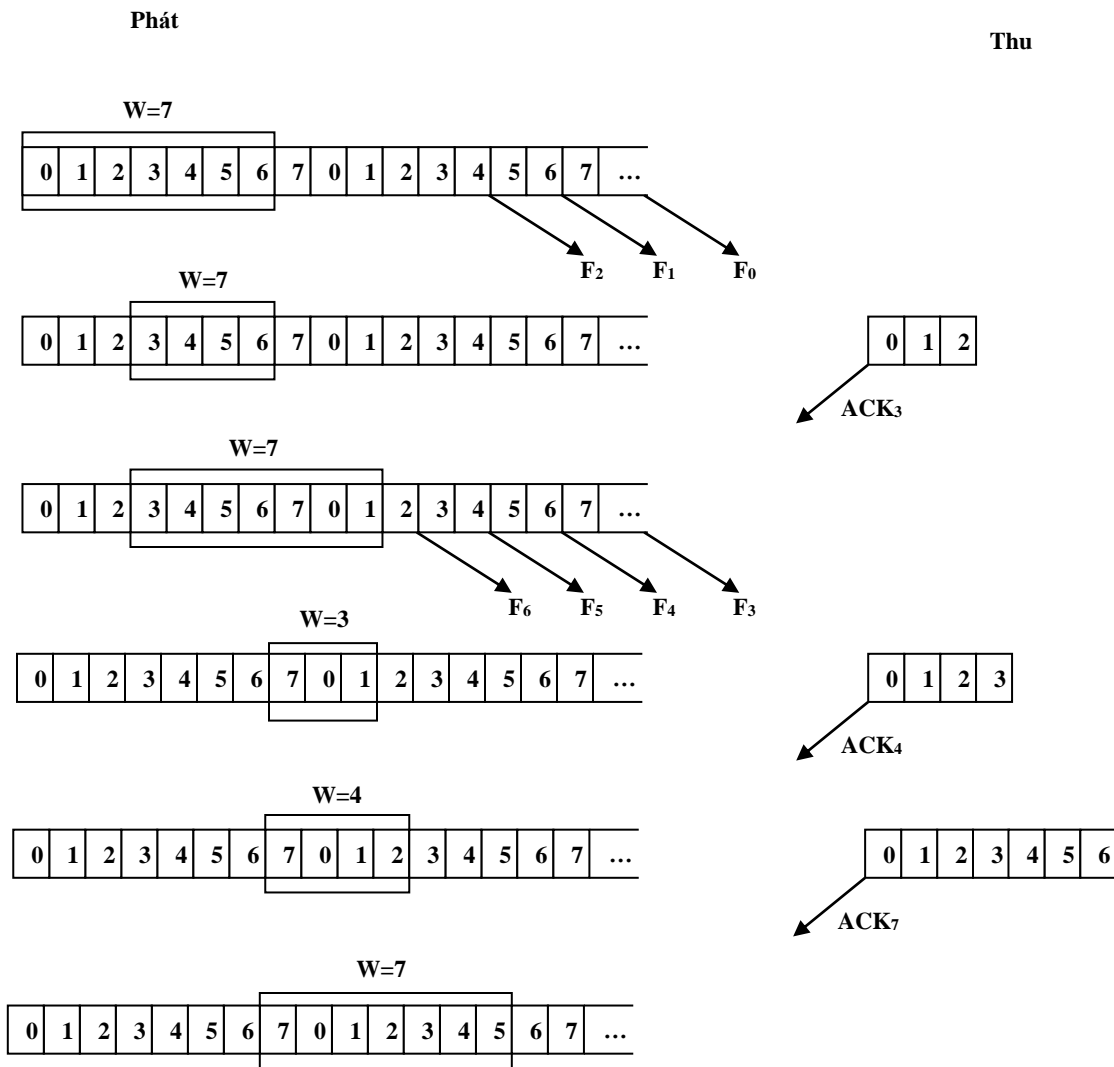
- Bên phát phát liên tiếp  $W$  khung tin trước khi được nhận báo nhận ( $W$ . Kích thước cửa sổ).
- Mỗi khi phát, phát xong 1 khung tin thì kích thước cửa sổ giảm 1 ( $W-1$ )
- Mỗi khi bên phát nhận được báo nhận ACK của 1 khung tin thì tự động tăng kích thước cửa sổ lên 1 ( $W+1$ )
- Trong trường hợp  $W > 0$ . Bên phát vẫn được phát tin.



- Trường hợp  $W=0$ , phía phát sẽ dừng phát tin.
- Do phía phát được phép phát nhiều hơn 1 khung tin nên cần có cơ chế đánh số thứ tự cho các khung tin. Dùng  $k$  bit để đánh số thứ tự cho các khung tin thì:

$$0 \leq W \leq 2^k - 1$$

Ví dụ: dùng  $k = 3$  bit để đánh số thứ tự khung tin, lấy  $W = 7$ .



*b. Hiệu suất.  $\eta_{sw}$*

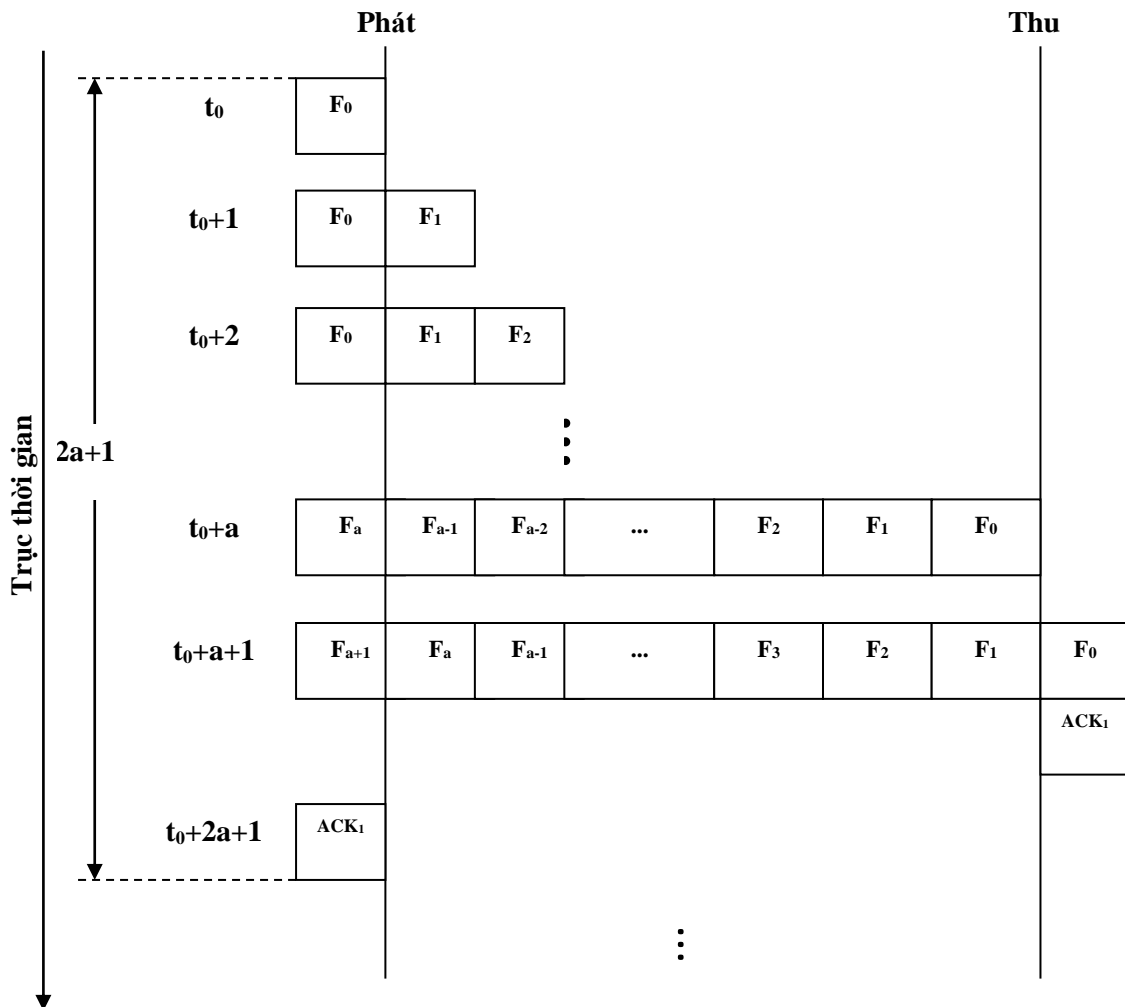
Theo kỹ thuật điều khiển luồng theo kiểu dừng và đợi ta có:

$$a = \frac{T_d}{T_f}$$

Để cho đơn giản chuẩn hóa thời gian  $T_f = 1$  đơn vị thời gian.

$\Rightarrow T_d = a$  ( đơn vị thời gian ).

Gọi  $T_0$  là thời điểm phía phát phát khung tin đầu tiên  $F_0$ .



- Nếu  $W < 2a + 1$

Bên phát đã phát hết  $W$  khung tin nhưng  $ACK_1$  vẫn chưa nhận được.

$$\eta_{sw} = \frac{W}{2a + 1}$$

- Nếu  $W \geq 2a + 1$

Bên phát chưa phát hết  $W$  khung tin nhưng đã nhận được ACK. Bên phát vẫn tiếp tục phát tin mà không dừng. Chu trình chỉ hoàn thành khi  $W=0$ .

Trường hợp này ta có:  $\eta_{sw} = 1$

Vậy

$$\eta_{sw} = \begin{cases} \frac{W}{2a + 1} & \text{Nếu } W < 2a + 1 \\ 1 & \text{Nếu } W \geq 2a + 1 \end{cases}$$

### 3. Kiểm soát lỗi.

#### 3.1 Nguyên tắc phát hiện lỗi.

Dùng phương pháp kiểm tra mã dư vòng CRC

- Nguyên tắc:

- Bên phát:

+ Bước 1: Chọn thông báo cần phát đi  $M(x)$

+ Bước 2: Chọn đa thức sinh  $G(x)$  có bậc  $n$

+ Bước 3: Tính  $\frac{M(x).x^n}{G(x)}$  lấy phần dư

$$\frac{M(x).x^n}{G(x)} = Q(x) + \frac{R(x)}{G(x)}$$

+ Bước 4: Tính  $T(x) = M(x).x^n + R(x)$

$T(x)$  chính là thông báo cần phát đi.

- Bên thu:

+ Bước 1: Tính

$$\frac{T(x)}{G(x)} = \frac{M(x).x^n}{G(x)} = \frac{M(x).x^n}{G(x)} + \frac{R(x)}{G(x)} = Q(x) + \frac{R(x)}{G(x)} + \frac{R(x)}{G(x)}$$

+ Bước 2: Tính  $R(x)$

Nếu  $R(x) = 0$  thì  $T(x)$  là không bị sai.

Nếu  $R(x) \neq 0$  thì  $T(x)$  nhận được là bị sai.

### ***3.2 Các kỹ thuật yêu cầu tự động phát lại.***

#### ***3.2.1 Kỹ thuật ARQ dừng và đợi.***

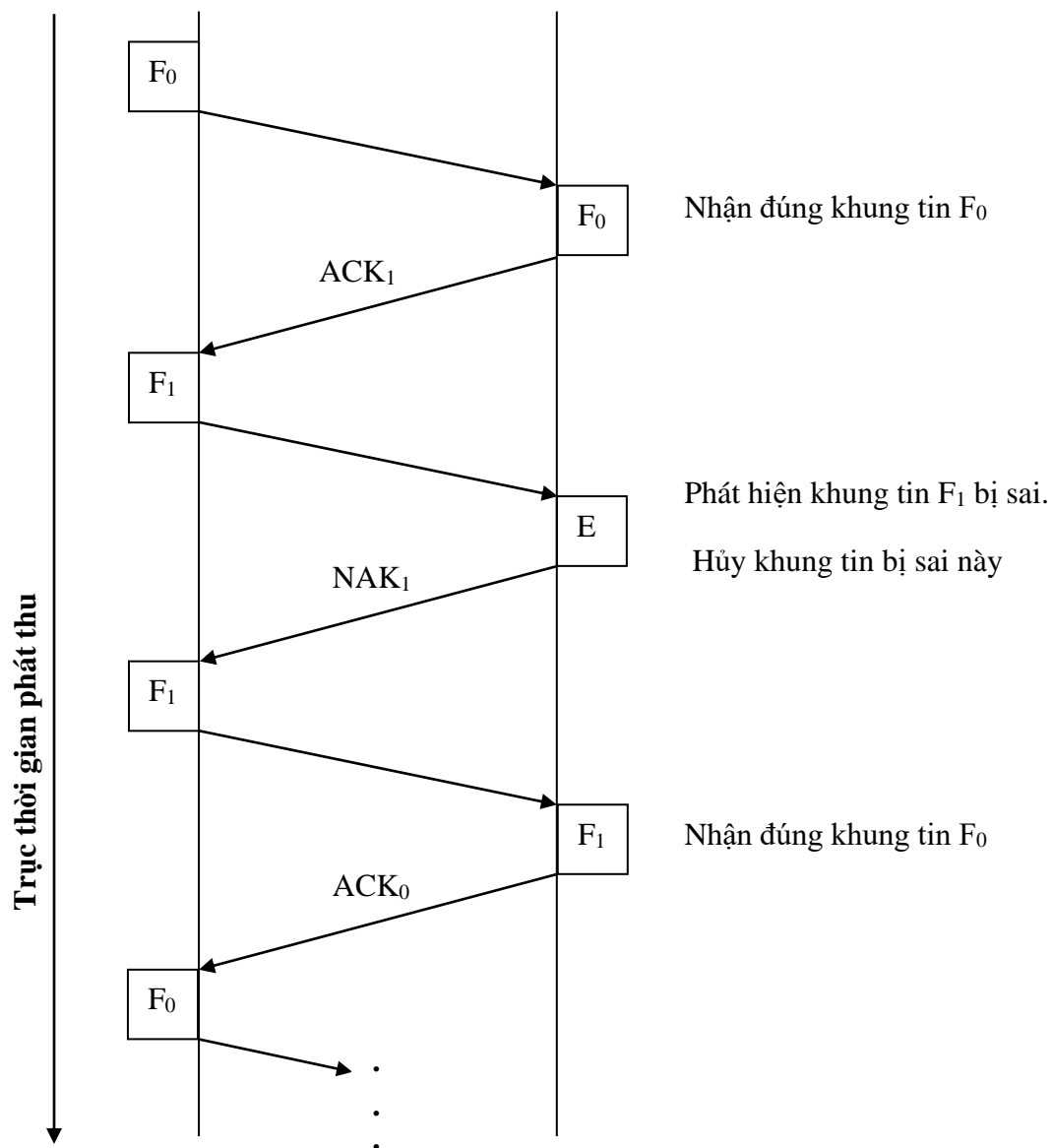
- Hoạt động:

Dựa trên nguyên lý của kỹ thuật điều khiển luồng theo kiểu dừng và đợi. Tức là:

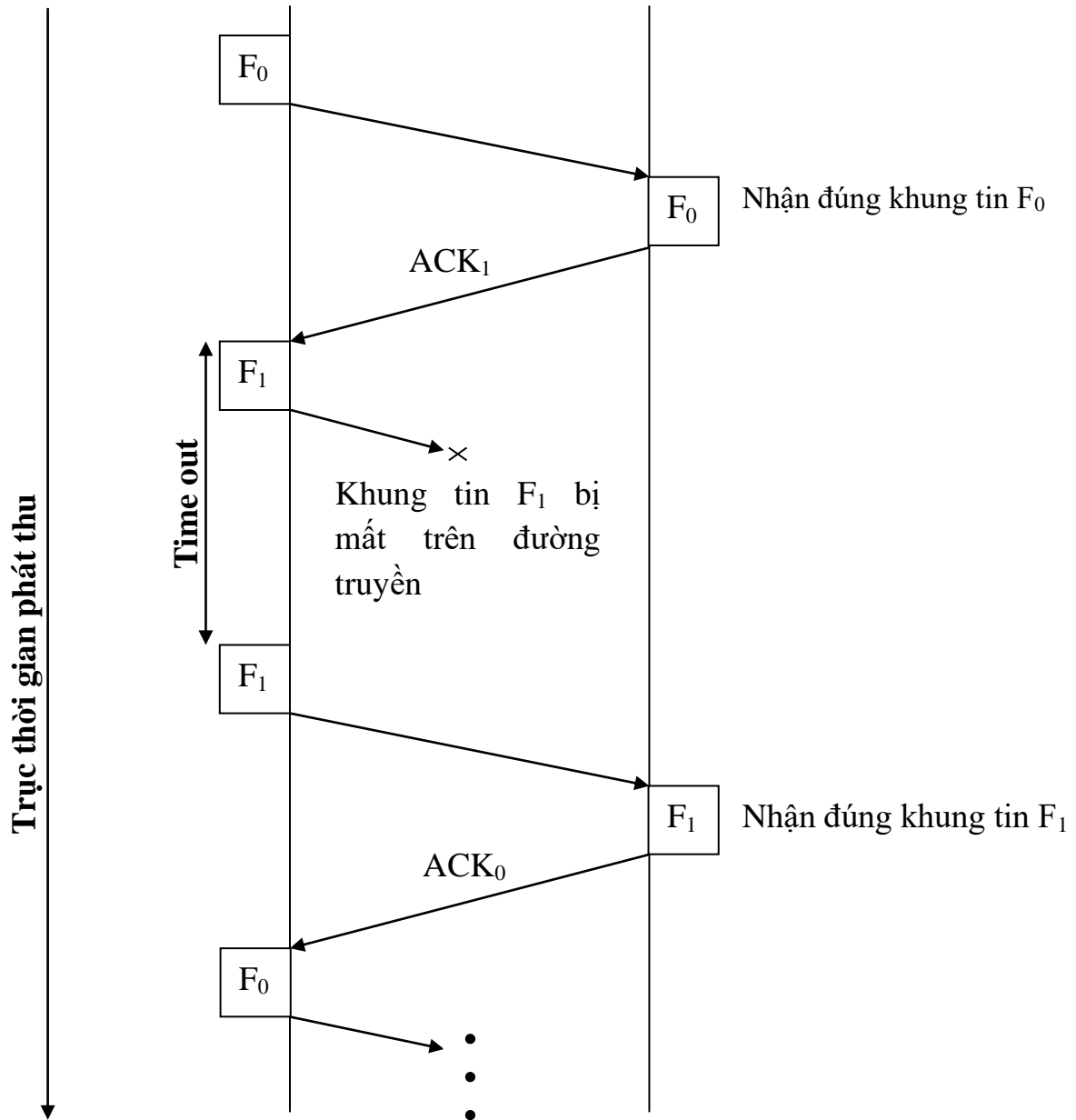
Khi không có lỗi xảy ra, phía thu gửi ACK 1 cách bình thường cho phía phát.

Khi phía thu nhận được 1 khung tin nào đó bị sai thì phía thu sẽ gửi cho phía phát 1 NAK có số hiệu tương ứng với số hiệu của khung tin bị sai đó đồng thời phía thu sẽ hủy khung tin bị sai vừa nhận được.

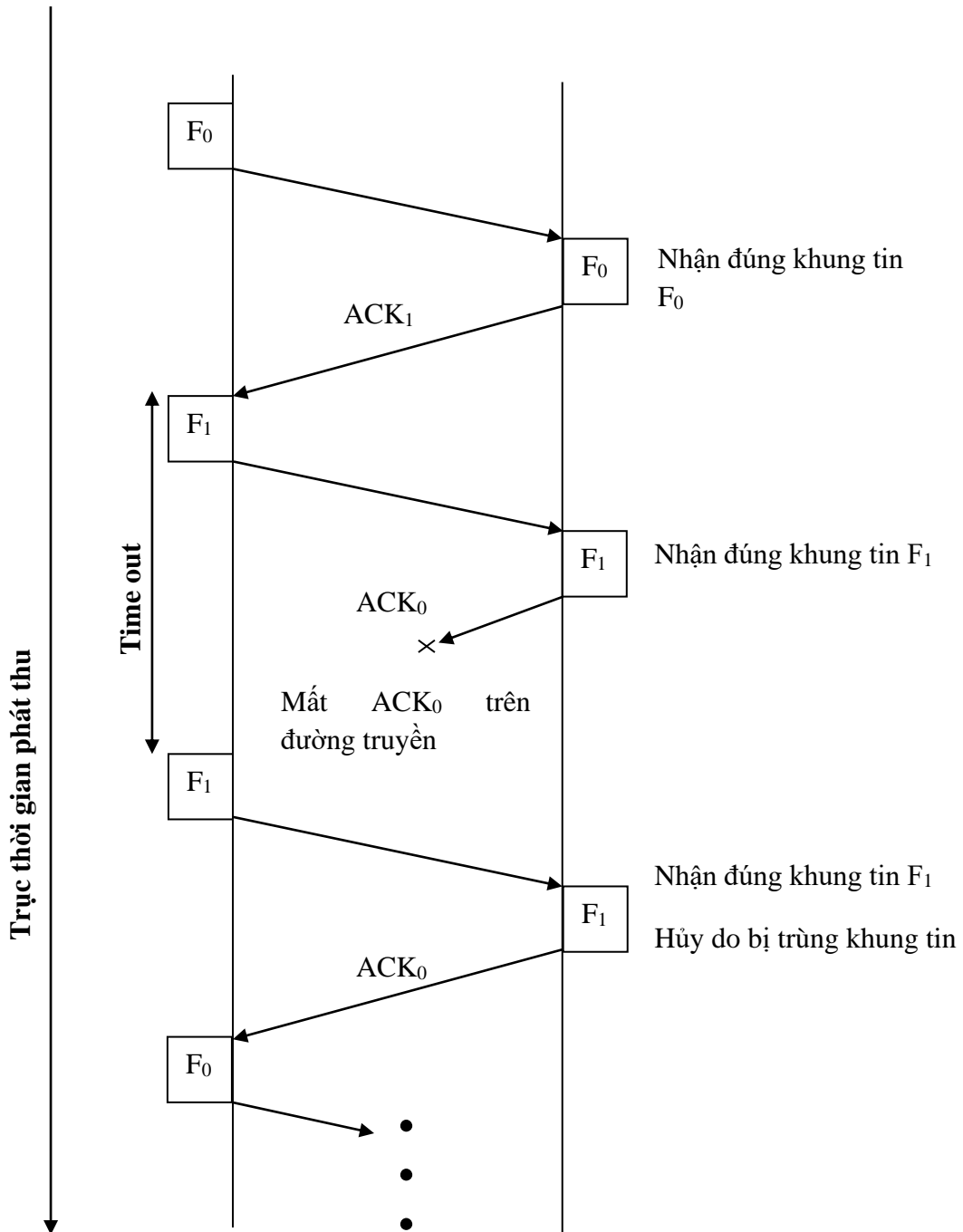
Khi phía phát nhận được NAK phía phát sẽ thực hiện phát lại khung tin đã phát trước đó. ( Khung tin có số hiệu tương ứng với số hiệu NAK vừa nhận được)



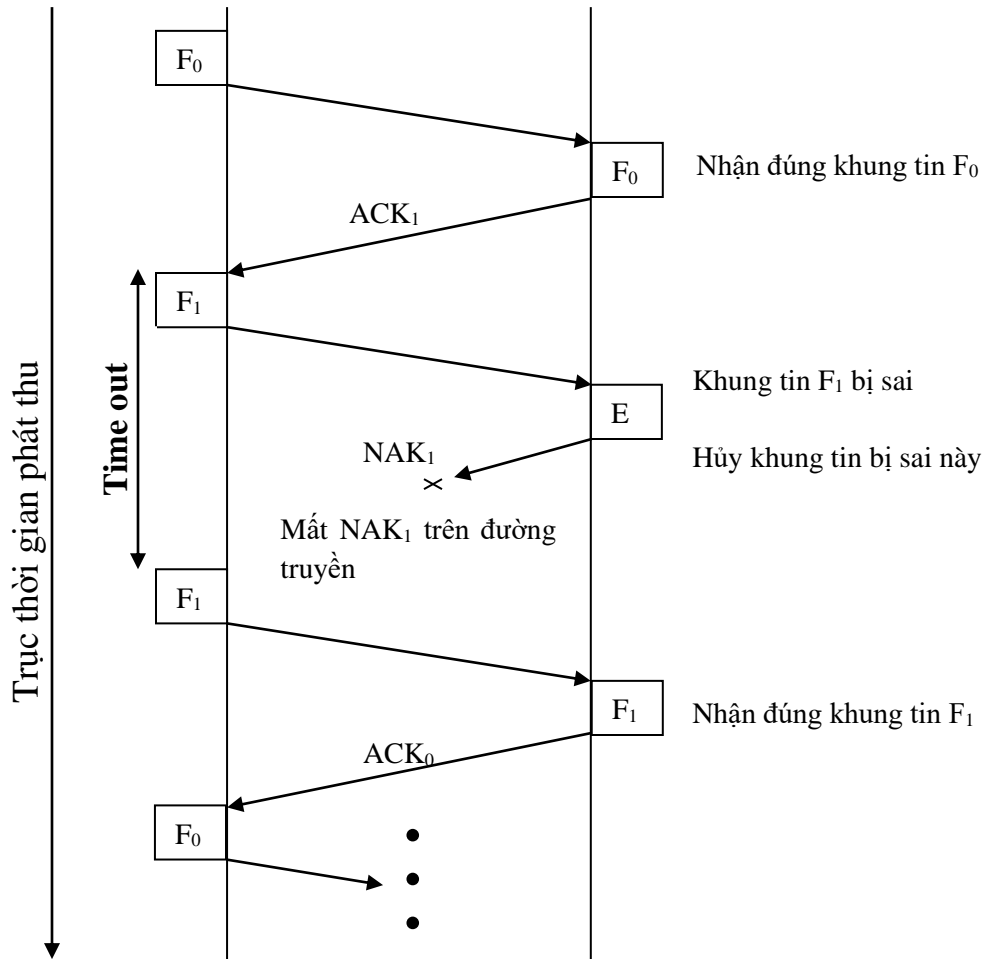
- Một số trường hợp lỗi khác:
  - + Khung tin bị mất trên đường truyền.



+ Mất ACK trên đường truyền.



Mất NAK trên đường truyền.



- Hiệu suất.  $\eta_{SAW-ARQ}$ 
  - Một số định nghĩa:
    - + Xác suất lỗi bit ( $P_b$ ): là xác suất phía thu nhận được bit 1 trong khi phía phát truyền bit 0 ( hoặc ngược lại), ( $0 \leq P_b \leq 1$ ).
    - + Xác suất lỗi khung tin  $P_f$ :  $P_f \approx l \cdot P_b$ . Với  $l$  là độ dài khung tin.
    - + Nếu gọi  $N_r$  ( $1 \leq N_r \leq \infty$ ) là số khung tin trung bình phải truyền cho đến khi truyền thành công.



- Hiệu suất thực tế:

$$\eta_{thực\,tế} = \frac{\eta_{li\,tương}}{N_r}$$

Do đó:

$$\eta_{SAWARQ} = \frac{\eta_{SAW}}{N_r} = \frac{1}{(1 + 2a) \cdot N_r}$$

- Tính  $N_r$ :

+ Giả sử ACK, NAK không bao giờ bị lỗi.

+ Gọi  $P_k$  là xác suất 1 khung tin bị lỗi.

+ Giả sử phía phát cần truyền khung tin lần thứ  $i$  mới thành công ( $1 \leq i \leq \infty$ )

$\Rightarrow$  Xác suất truyền đúng khung tin ở lần thứ  $i$ :

$$P(i) = P_k^{i-1} \cdot (1 - P_k)$$

Số khung tin phải truyền cho đến lần thứ  $i$  là  $f(i) = i$  (khung tin).

Vậy

$$N_r = \sum_{i=1}^{\infty} f(i) \cdot P(i) = \sum_{i=1}^{\infty} i \cdot P_k^{i-1} \cdot (1 - P_k) = \frac{1}{1 - P_f}$$

### 3.2.2 Kỹ thuật ARQ trở lại $N$ (go back $N$ -ARQ).

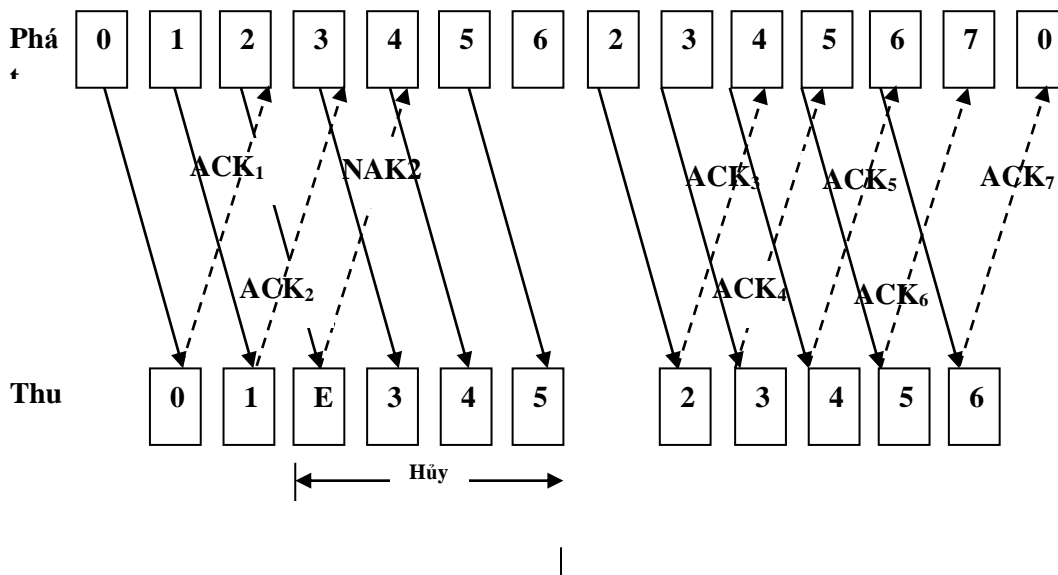
a. Hoạt động.

- Dựa trên nguyên lý hoạt động của kỹ thuật điều khiển luồng theo kiểu cửa sổ trượt
- Khi không có lỗi phía thu gửi ACK 1 cách bình thường cho phía phát.

- Khi phía thu phát hiện 1 khung tin nào đó bị sai thì phía thu sẽ gửi 1 NAK có số hiệu tương ứng với số hiệu khung tin bị sai, đồng thời phía thu sẽ hủy tất cả các khung tin tính từ khung tin bị sai trở đi.

- Khi phía phát nhận được NAK thì phía phát sẽ thực hiện phát lại tất cả các khung tin tính từ khung tin có số hiệu tương ứng với số hiệu NAK vừa nhận được.

Ví dụ: Dùng  $k=3$  bit để đánh số thứ tự cho các khung tin,  $W=6$



• Xét các trường hợp sau:

- Trường hợp 1: Lỗi khung tin: 3 khả năng xảy ra.

+ Khả năng 1: Giả sử khung tin thứ  $i$  bị lỗi và phía thu nhận đúng khung tin  $i-1$  trở về trước. Với trường hợp này phía thu sẽ hủy các khung tin nhận được từ khung tin thứ  $i$  gửi  $NAK_i$  để báo lỗi cho khung tin thứ  $i$ .

+ Khả năng 2:

Giả sử khung tin  $i$  bị mất trên đường truyền và khung tin  $i+1$  đã nhận được ở bên thu. Trong trường hợp này phía thu sẽ nhận thấy rằng khung tin  $i+1$  đến không đúng thứ tự và hiểu rằng khung tin  $i$  bị mất trên đường truyền do đó phía thu sẽ hủy

tất cả các khung tin nhận được kể cả khung tin thứ  $i+1$  và gửi  $NAK_i$  cho phía phát để báo lỗi cho khung tin  $i$ . Phía phát nhận được  $NAK_i$  sẽ thực hiện phát lại các khung tin tính từ khung tin  $i$  vừa phát trước đó.

+ Khả năng 3:

Giả sử khung tin  $i$  bị mất trên đường truyền và phía phát không phát thêm 1 khung tin nào nữa. Trường hợp này phía phát phải chờ hết time-out của khung tin thứ  $i$  rồi phát lại khung tin thứ  $i$ .

- Trường hợp 2: ACK bị mất trên đường truyền. Có 2 khả năng xảy ra:

+ Khả năng 1:

Giả sử phía thu nhận đúng khung tin thứ  $i$ , phía phát gửi  $ACK_{i+1}$  để báo nhận đúng cho khung tin  $i$  và  $ACK_{i+1}$  bị mất trên đường truyền. Nếu trước khi time-out của khung tin thứ  $i$  xảy ra mà phía phát nhận được  $ACK_{i+n}$  ( $n \geq 2$ ) thì phía phát hiểu rằng khung tin  $i$  nhận được đúng bởi phía thu.

+ Khả năng 2:

Sau khi time-out của khung tin xảy ra mà bên phát không nhận được bất cứ  $ACK_{i+n}$  ( $n \geq 2$ ) nào thì phía thu sẽ thực hiện phát lại khung tin  $i$  và các khung tin tiếp theo sau đó nếu đã được phát.

- Trường hợp 3: NAK bị mất trên đường truyền.

Điều này đồng nghĩa với việc khung tin  $i$  bị lỗi do đó phía thu sẽ không nhận các khung tin sau khung tin  $i$  và các khung tin này không được gửi báo nhận. Trường hợp này bắt buộc bên phát phải chờ hết time-out của khung tin thứ  $i$  rồi thực hiện phát lại khung tin thứ  $i$  và các khung tin phía sau đó.

b. Hiệu suất:  $\eta_{GBN-ARQ}$ .

Ta có:

$$\eta_{GBN-ARQ} = \begin{cases} \frac{W}{(1+2a) \cdot N_r} ; & W < 2a + 1 \\ \frac{1}{N_r} ; & W \geq 2a + 1 \end{cases}$$

• Tính  $N_r$

- Gọi  $P_f$  là xác suất 1 khung tin bị lỗi.
- Khi có lỗi phía phát sẽ thực hiện phát k khung tin ( $1 \leq k \leq W$ ).
- Giả sử truyền đến lần thứ i mới thành công ( $1 \leq i \leq \infty$ ).
- Giả sử ACK, NAK không bao giờ bị lỗi.

Vậy:

Số khung tin phải truyền lại cho đến lần thứ i là:

$$f_i = (i - 1) \cdot k + 1$$

Xác suất truyền đúng ở lần thứ i là:

$$P(i) = P_f^{i-1} \cdot (1 - P_f) \Rightarrow N_r = 1 - k + \frac{k}{1 - P_f}$$

- Tính k:

Giả sử phía phát luôn đủ dữ liệu để phát và chỉ dừng lại khi  $W=0$ .

- Nếu  $W < 2a+1$ : Khi phía phát phát xong W khung thì NAK (ACK) mới đến được bên phát do đó  $k=W$ .
- Nếu  $W \geq 2a+1$ : Khi NAK(ACK) đến được bên phát thì bên phát phát đi  $\approx 2a+1$  khung. Do đó  $k \approx 2a+1$ .

$$\text{Nên: } \eta_{GBR-ARQ} = \begin{cases} \frac{W(1-P_f)}{(1+2a)(1-P_f+W.P_f)}; & \text{Nếu } W < 2a + 1 \\ \frac{1-P_f}{1+2aP_f}; & \text{Nếu } W \geq 2a + 1 \end{cases}$$

a. Hoạt động.

- Ví dụ: dùng k=3 bit để đánh số thứ tự cho các khung tin. W=6



b. Hiệu suất.  $\eta_{SR-ARQ}$ .

Tính tương tự đối với kỹ thuật ARQ trở lại N thay  $k=1 \Rightarrow N_r = \frac{1}{1-P_f}$

Vậy:

$$\eta_{SR-ARQ} = \begin{cases} \frac{(1 - P_f) \cdot W}{1 + 2a}; & \text{Nếu } W < 1 + 2a \\ 1 - P_f; & \text{Nếu } W \geq 1 + 2a \end{cases}$$

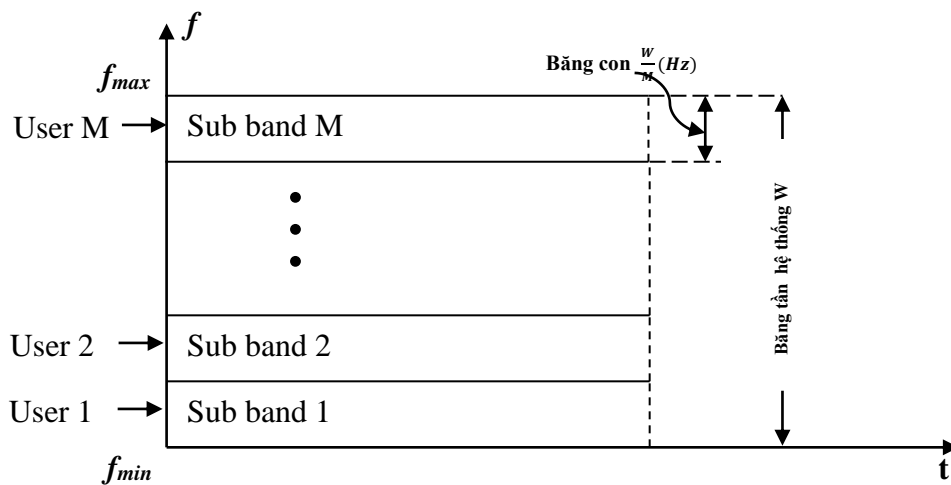
## CHƯƠNG VI. CÁC GIAO THỨC TRUY NHẬP ĐƯỜNG TRUYỀN.

### 1. Các giao thức phân kênh cố định.

#### 1.1 Giao thức FDMA.

- Băng tần của hệ thống  $W(\text{Hz})$  được chia thành các băng con bằng nhau, mỗi băng có độ rộng là  $\frac{W}{M}(\text{Hz})$ .

- Mỗi người dùng truyền tin theo các băng con dành riêng cho mình một cách tuần hoàn

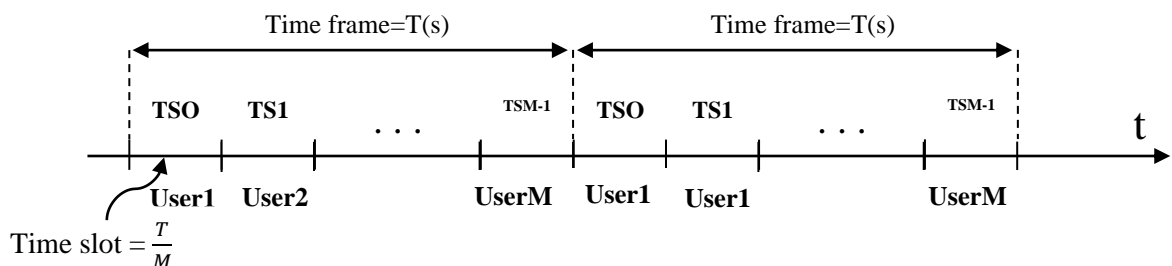


#### 1.2 Giao thức TDMA.

- Chia thời gian thành các khung thời gian bằng nhau, mỗi khung được chia thành  $M$  khe thời gian bằng nhau.

- Mỗi khung thời gian có độ rộng là  $T$  giây, mỗi khe thời gian có độ rộng là  $\frac{T}{M}$  giây

- Mỗi người dùng truyền tin trên các khe thời gian dành riêng cho mình một cách tuần hoàn



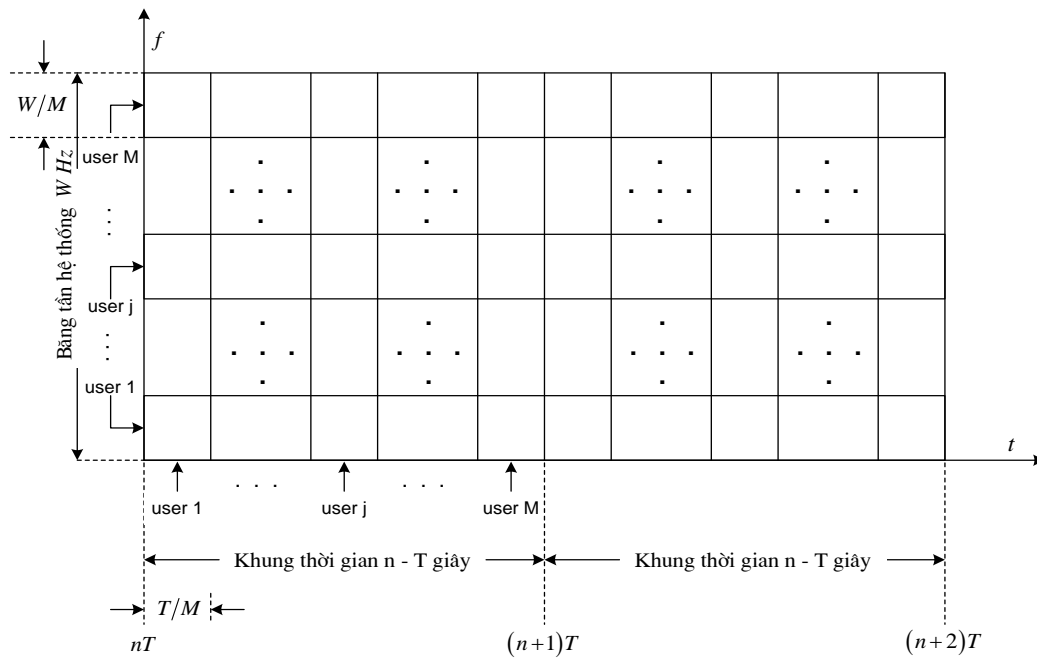
### 1.3 So sánh 2 giao thức FDMA và TDMA.

- So sánh về thông lượng

+ Giả sử tài nguyên thông tin có khả năng cho tốc độ tổng thể  $R$  (bít/s). Xét trong hệ thống có  $M$  người dùng.

+ Xét trong một khung thời gian có độ dài  $T$  giây.

Trong hình vẽ trên ta thấy, trong hệ thống FDMA, độ rộng băng tần hệ thống được chia thành  $M$  băng con. Do đó, mỗi người dùng có thể phát đồng thời với một tốc độ bít là  $\frac{R}{M}$  (bít/s).



Còn trong hệ thống TDMA, khung thời gian được chia thành  $M$  khe. Do vậy, mỗi người dùng phát theo loạt với tốc độ  $R$  (bít/s) (nhanh hơn  $M$  lần so với người dùng trong hệ thống FDMA) trong khoảng thời gian  $\frac{T}{M}$  (s)



Giả sử, mỗi gói tin có  $b$  (bít). Trong trường hợp FDMA, các gói tin có độ dài  $b$  (bít) được truyền trong  $T$  (giây) trên mỗi kênh con tách rời.

Do vậy, tốc độ bít yêu cầu là:  $R_{FD} = M \frac{b}{T}$  (bít/s) .

Trong trường hợp TDMA, các gói tin  $b$  (bít) được phát trong khoảng thời gian  $\frac{T}{M}$  (s).

Do đó, tốc độ bít yêu cầu là:  $R_{TD} = \frac{b}{T/M} = M \frac{b}{T}$  (bít/s) .

Từ hai kết quả này ta thấy rằng:  $R_{FD} = R_{TD} = R = M \frac{b}{T}$  (bít/s).

Như vậy, hai hệ thống có cùng tốc độ dữ liệu là  $R$  (bít/s).

- So sánh về độ trễ trung bình của gói tin.

Gọi  $D_{FDMA}$  và  $D_{TDMA}$  là thời gian trễ của hệ thống FDMA và TDMA.

Ta có :  $D = W + t$ .

Trong đó :

$W$  là thời gian chờ để phát 1 gói tin.

$t$  là thời gian truyền hết 1 gói tin.

+ Đối với FDMA :

Do không có thời gian chờ (vì các băng con được cấp trước).

$\Rightarrow W = 0 \Rightarrow D_{FDMA} = t = T$  (phát các gói tin trong toàn bộ khung thời gian)

+ Đối với TDMA :

Người dùng phải chờ đến khe của mình mới được phép truyền tin.

Gọi  $P_i$  là xác suất gói tin tới vào khe thứ  $i$  ( $1 \leq i \leq M$ ) thì:

$$P_i = \frac{1}{M} \text{ (xác suất phân bố đều)}$$

Số khe phải chờ cho đến khi phát tin là:  $f_i = M - i$  (khe)

Vậy thời gian trung bình phải chờ cho đến khi phát là :

$$\begin{aligned} W &= \sum_{i=1}^M P_i f_i \frac{T}{M} \\ &= \sum_{i=1}^M \frac{1}{M} (M - i) \frac{T}{M} = \frac{1}{M} \frac{T}{M} \sum_{i=1}^M (M - i) \\ &= \frac{1}{M} \frac{T}{M} \frac{(M-1)M}{2} = \frac{T}{M} \frac{(M-1)}{2} \end{aligned}$$

Do  $t = \frac{T}{M}$  (thời gian phát 1 gói tin).

$$D_{TDMA} = \frac{T}{M} \frac{(M-1)}{2} + \frac{T}{M} = T \left( \frac{1}{2} + \frac{1}{2M} \right) \leq T = D_{FDMA}$$

Vậy khi  $M$  tăng thì độ trễ càng bé hay khi hệ thống phân thành nhiều luồng thì độ trễ của hệ thống càng giảm xuống.

## 2. Các giao thức truy cập ngẫu nhiên.

### 2.1 Giao thức *Pure Aloha*.

Các gói tin có kích thước như nhau, thời gian truyền các gói tin là như nhau.

Gọi  $T$  là thời gian truyền 1 gói tin.

- Hoạt động :

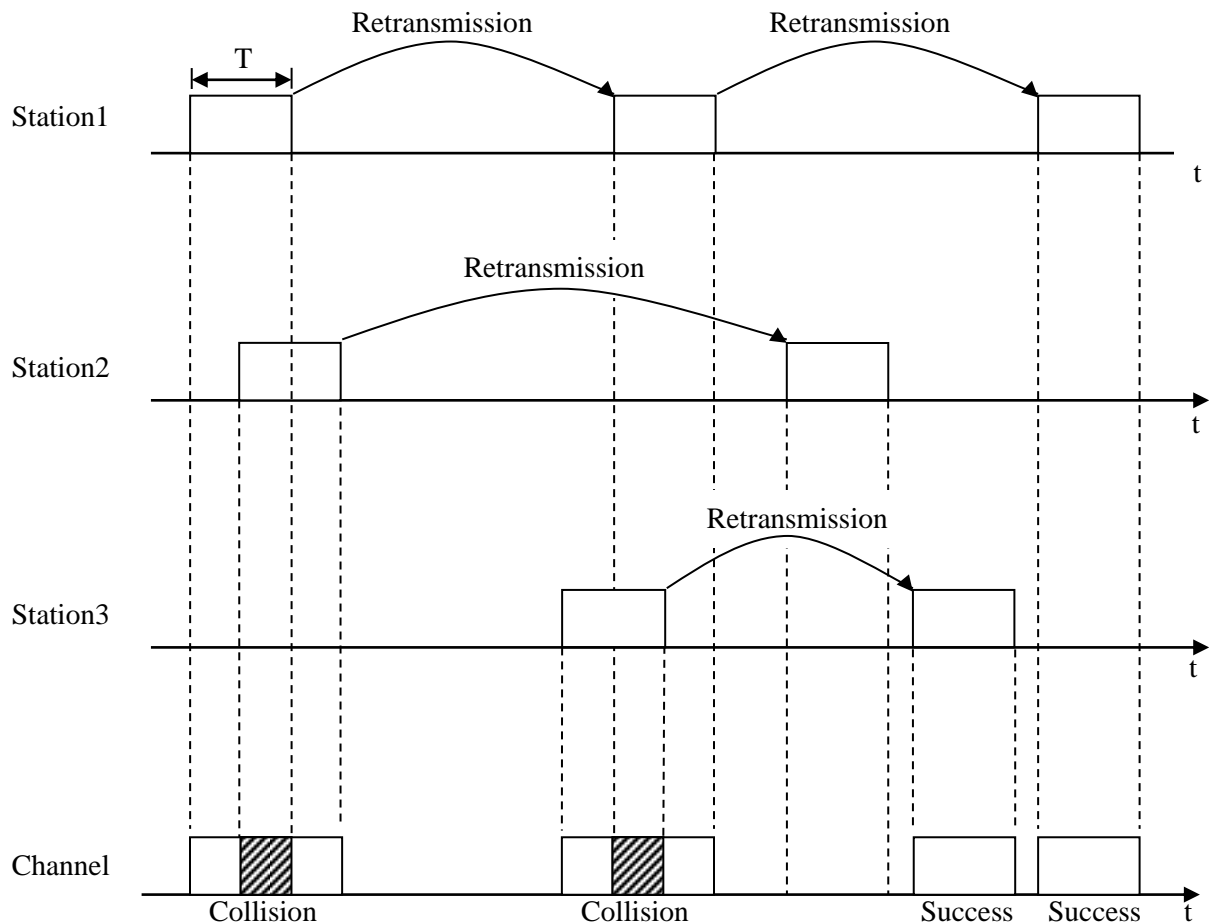
Các trạm sẽ thực hiện truyền tin ngay lập tức khi có nhu cầu cần truyền tin.

+ Trong trường hợp trên đường truyền không có tín hiệu các trạm khác thì gói tin vừa truyền coi như thành công.

+ Trong trường hợp trên đường truyền có tín hiệu từ các trạm khác thì gói tin vừa truyền coi như bị xung đột. (Collision).

Khi có xung đột xảy ra, các trạm có tín hiệu bị xung đột đó sẽ chạy thuật toán Back off (B.O) để xác định thời điểm truyền lại của các gói tin bị truyền lại trong tương lai.

Ví dụ : Hệ thống gồm 3 trạm chia sẻ kênh truyền dẫn chung :

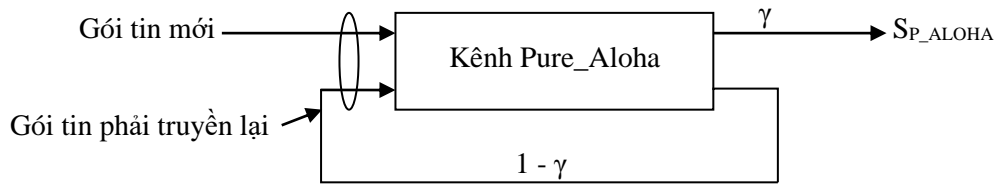


- Tính toán thông lượng.

Gọi  $S$  là thông lượng của hệ thống Pure Aloha (số gói tin phát thành công trong khung thời gian  $T$ ).

Gọi  $G$  là lưu lượng của hệ thống Pure Aloha.

Gọi  $\gamma$  là xác suất truyền thành công 1 gói tin.



( $G$  được tính bằng số gói tin được truyền đi trên 1s)

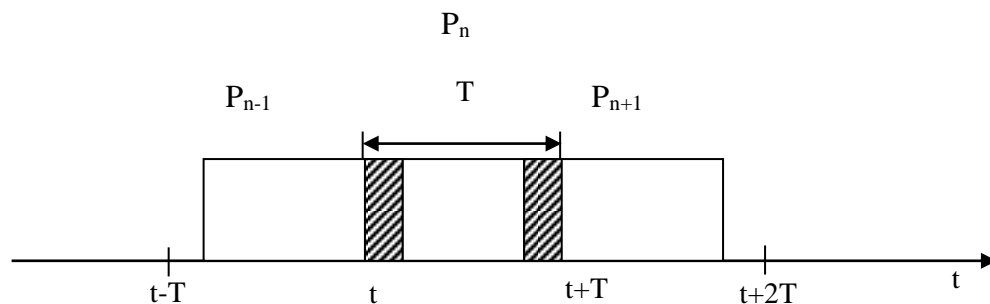
Vậy :  $S_{P\_ALOHA} = G \cdot \gamma$

Giả sử các gói tin có phân bố poát xông (Poisson).

→ Xác suất có  $k$  gói tin đến trong khoảng thời gian  $[0, t]$ .  $P = \frac{(\lambda t)^k}{k!} e^{-\lambda t}$

Với :  $\lambda$  là tốc độ dữ liệu tới.  $\lambda = \frac{G}{T}$  (packet/s)

Xét gói tin tham chiếu  $P_n$ , được truyền  $[t, t+T]$



Để cho  $P_n$  truyền thành công thì trong đoạn  $[t-T, t+T]$  phải không có trạm nào truyền tin. Tức là khoảng thời gian  $t = 2T$  gọi là khoảng thời gian xung đột.

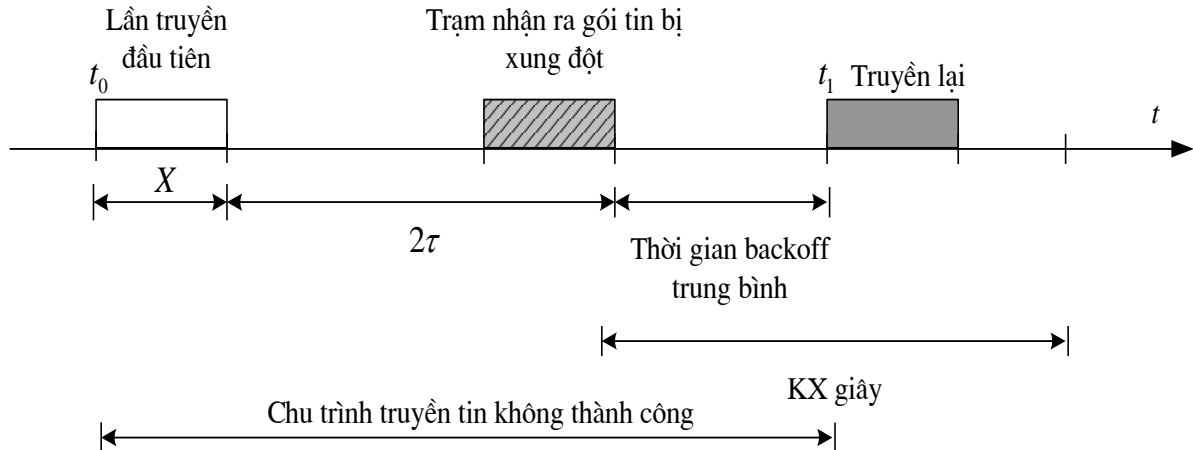
$$\text{Ta có: } \gamma = \frac{(\lambda 2T)^0}{0!} e^{-\lambda 2T} = e^{-G} \text{ với } \lambda = \frac{G}{T}$$

$$\text{Vậy: } S_{P\_ALOHA} = G \cdot e^{-2G}$$

$$(S_{P\_ALOHA})_{\max} = \frac{1}{2e} \approx 18,5\% \text{ với } G = \frac{1}{2}$$

- Tính toán độ trễ trung bình của gói tin

+ Gọi  $\tau$  là trễ truyền dẫn từ đầu cuối tới đầu cuối. Vậy sau một khoảng thời gian là  $2\tau$  giây trạm đó sẽ biết gói tin của mình có truyền thành công hay không (tương ứng với  $R$  lần truyền một gói tin)



+ Khi gói tin bị xung đột thì phải truyền lại gói tin đó. Việc truyền lại sẽ được thực hiện trong  $K$  khoảng thời gian kế tiếp (chạy thuật toán Backoff), với mỗi khoảng thời gian được chọn một cách ngẫu nhiên với xác suất truyền tin là  $\frac{1}{K}$ .

+ Gọi  $\bar{i}$  là khoảng thời gian trung bình mà gói tin phải chờ để truyền lại.

Vậy:  $\bar{i} = \sum_{i=0}^{K-1} i \times p$  [khoảng thời gian thứ i]

$$= \sum_{i=0}^{K-1} i \times \frac{1}{K} = \frac{1}{K} \sum_{i=0}^{K-1} i = \frac{1}{K} \times \frac{K(K-1)}{2} = \frac{K-1}{2} \text{ (khe)}$$

Vậy:  $\bar{i} \cdot X = \frac{K-1}{2} X$  là thời gian backoff trung bình.

Giả sử, gói tin truyền đến lần thứ  $n$  mới thành công ( $1 \leq n \leq \infty$ ).

Vậy, xác suất truyền tin đến lần thứ  $n$  mới thành công là:

$$p(n) = (1 - e^{-2G})^{n-1} \cdot e^{-2G}$$

Do đó, số lần trung bình truyền một gói tin là:

$$\bar{n} = \sum_{n=1}^{\infty} n \cdot (1 - e^{-2G})^{n-1} e^{-2G} = e^{2G}$$

Do vậy:

$$E = \bar{n} - 1 = (e^{2G} - 1)$$

Gọi  $T_u$  là chu trình thời gian mà hệ thống truyền không thành công, thì:

$$T_u = \left(1 + R + \frac{K-1}{2}\right) X = \left(R + \frac{K+1}{2}\right) X \text{ (giây)}$$

Do đó, độ trễ trung bình của gói tin được tính như sau:

$$D = X \left[ R + \frac{K+1}{2} \right] E + [1 + R] X \text{ (giây)}$$

Thay các tham  $E = (e^{2G} - 1)$  vào công thức trên, ta thu được độ trễ trung bình của khung tin được chuẩn hoá theo thời gian phát một gói tin như sau:

$$D = \left[ R + \frac{K+1}{2} \right] (e^G - 1) \cdot X + [1 + R] \cdot X \text{ (giây)}$$

## 2.2 Giao thức Slotted\_Aloha.

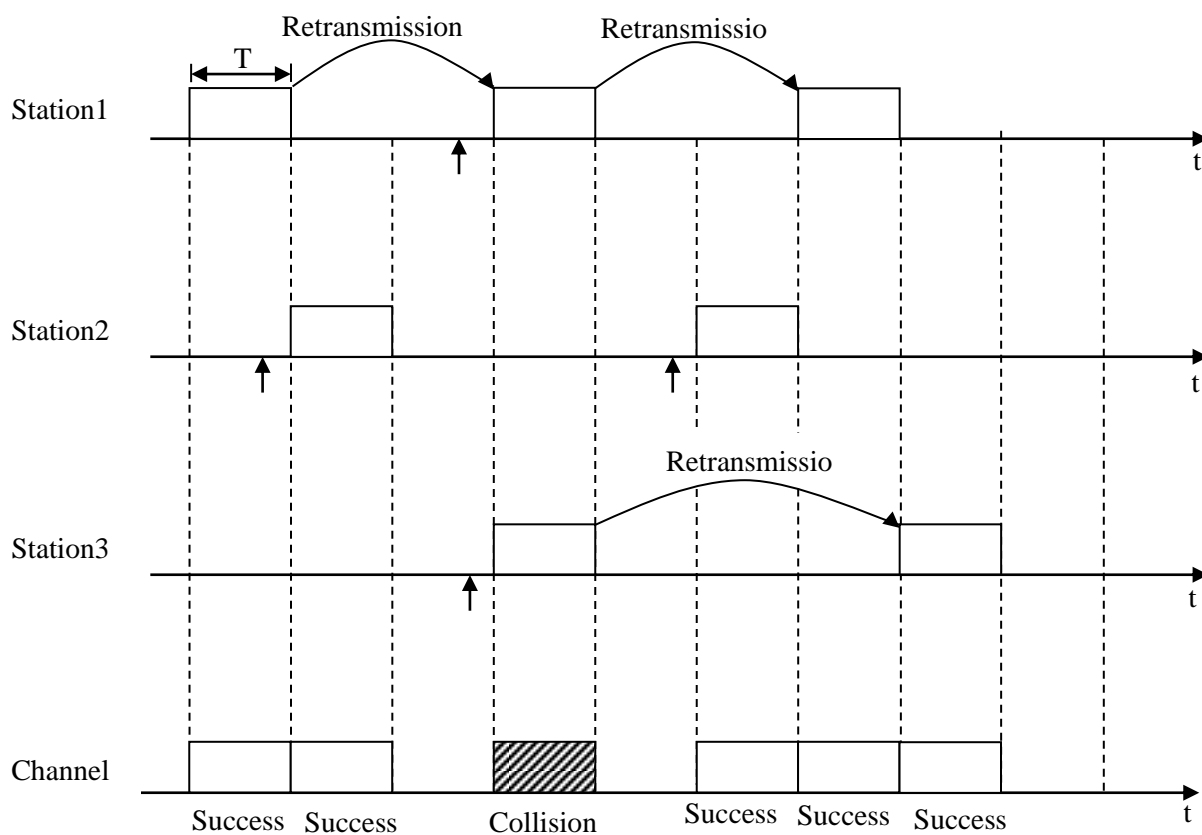
Chia thời gian thành các khe bằng nhau và bằng thời gian phát 1 gói tin có kích thước tối đa.

Gọi T là độ rộng 1 khe.

Hoạt động :

- Các trạm chỉ được phép truyền tin ở thời điểm đầu tiên của khe thời gian.
- Nếu như cần truyền tin của trạm giữa khe thì trạm đó phải chờ đến thời điểm đầu tiên của khe kế tiếp mới được truyền.
- Trong trường hợp trên đường truyền có 1 gói tin thì gói tin coi như truyền thành công. Trong trường hợp trên đường truyền có nhiều hơn 1 gói tin thì gói tin coi như bị xung đột.
- Khi có xung đột xảy ra thì các trạm có gói tin bị xung đột sẽ chạy thuật toán Back off để xác định thời điểm truyền lại trong tương lai.

Ví dụ : Có 3 trạm để chia sẻ vệ tinh dùng chung.

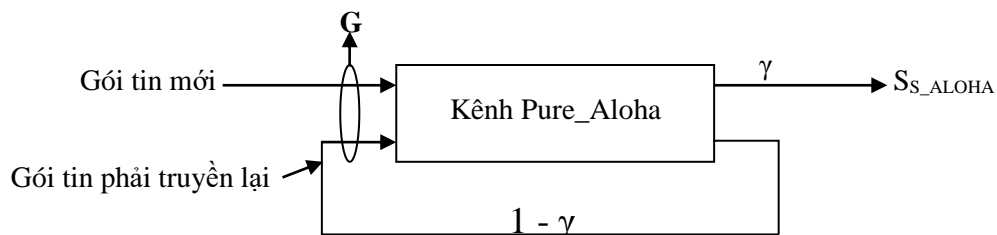


- Phân tích về thông lượng.

Gọi  $S_{S\_ALOHA}$  là thông lượng hệ thống Slotted\_Aloha.

Gọi  $G$  là lưu lượng của hệ thống Slotted\_Aloha.

Gọi  $\gamma$  là xác suất truyền thành công 1 gói tin.

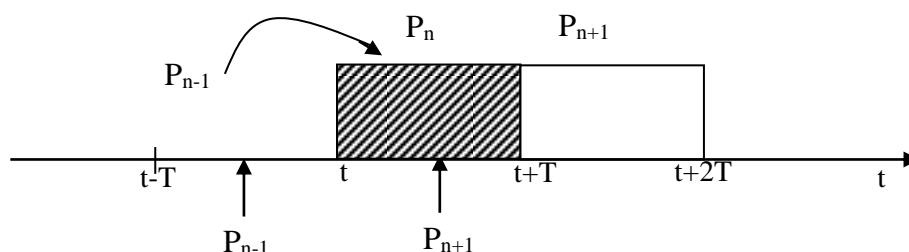




$$S_{S\_ALOHA} = G \cdot \gamma$$

Giả sử lưu lượng của kênh là 1 quá trình ngẫu nhiên theo phân phối Poisson

Xét gói tin tham chiếu  $P_n$ .



Điều kiện để  $P_n$  truyền tin thành công thì  $[t-T, t] = T$  không có trạm nào có nhu cầu truyền tin.

$$\text{Vậy: } \gamma = \frac{\lambda T^0}{0!} e^{-\lambda T} = e^{-G} \Rightarrow \gamma = \frac{\lambda T^0}{0!} e^{-\lambda T} = e^{-G}$$

Do đó:

$$S_{S\_ALOHA} = G e^{-G} \Rightarrow (S_{S\_ALOHA})_{\max} = \frac{1}{e} \approx 37\% \text{ với } G = 1$$

- Độ trễ trung bình của 1 gói tin trong giao thức Slotted\_Aloha.

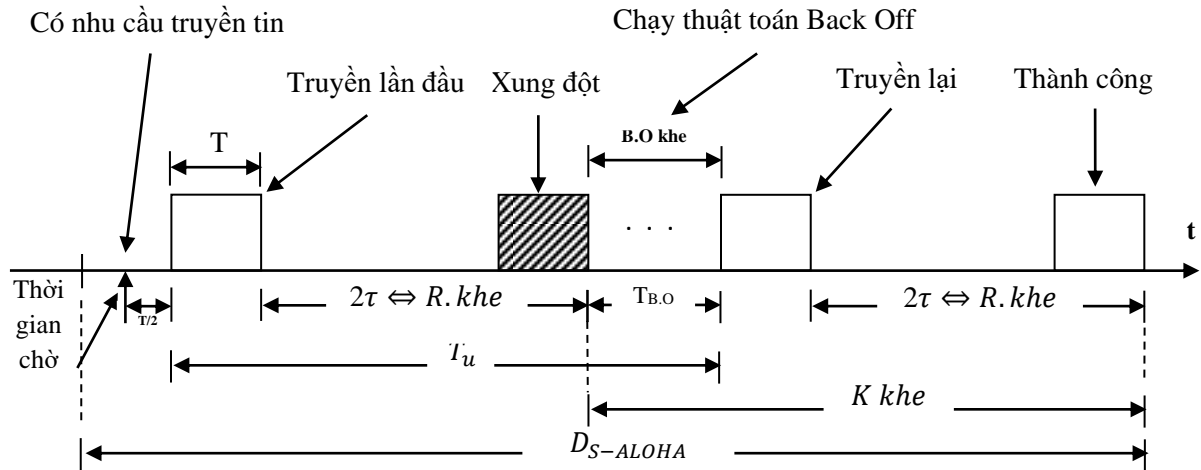
Gọi  $D_{S\_ALOHA}$  là độ trễ trung bình của 1 gói tin.

$T$  là thời gian phát 1 gói tin.

$\tau$  là độ trễ truyền dẫn từ đầu cuối đến đầu cuối.

$\Rightarrow$  Sau  $2\tau(s)$  1 trạm sẽ biết mình có truyền tin thành công hay không.

Và  $2\tau(s) = R \text{ khe thời gian.} \Rightarrow R = \frac{2\tau}{T}$



Khi bị xung đột các trạm phải truyền lại gói tin bị xung đột đó, việc truyền lại sẽ thực hiện trong k khe thời gian kế tiếp (Chạy thuật toán Back Off để tìm k) việc chạy thuật toán Back Off để xác định chọn 1 trong k khe để thực hiện truyền lại sẽ được thực hiện với xác suất  $1/k$ .

Gọi  $T_u$  là chu trình truyền tin không thành công.

$$T_u = T + 2\tau + T_{B.O} \text{ (s)}$$

$$= 1 + R + B.O \text{ (khe thời gian)}$$

Vậy:  $D_{S-ALOHA} = \frac{T}{2} + T_u \cdot \text{số lần xung đột} + 1 \text{ lần truyền thành công.}$

$$\text{Hay : } D_{S-ALOHA} = \frac{T}{2} + T_u \cdot E + T + 2\tau \text{ (s). (E là số lần xung đột)}$$

$$= \frac{1}{2} + T_u \cdot E + 1 + R \text{ (khe thời gian)}$$

Tính B.O.

Gọi  $\bar{\tau}$  là số khe trung bình mà trạm phải chờ cho tới khi truyền lại.

$$\bar{i} = \sum_{i=0}^{k-1} iP(i)$$

Với  $P(i)$  là xác suất truyền ở khe thứ  $i$ .  $P(i) = \frac{1}{k}$

$$\begin{aligned}\bar{i} &= \sum_{i=0}^{k-1} i \frac{1}{k} = \frac{1}{k} \sum_{i=0}^{k-1} i \\ &= \frac{1}{k} [1 + 2 + 3 \dots + (k-2) + (k-1)] \\ &= \frac{1}{k} \frac{(k-1)k}{2} = \frac{k-1}{2}\end{aligned}$$

Vậy:

$$\begin{aligned}T_u &= 1 + R + B.O \text{ (khe)} \\ &= 1 + R + \frac{k-1}{2} = R + \frac{k+1}{2} \text{ (khe)}\end{aligned}$$

Tính số lần truyền lại E.

Giả sử gói tin phải truyền lại cho đến lần thử  $n$  mới thành công ( $1 \leq n \leq \infty$ )

Gọi  $P_n$  là xác suất truyền thành công lần thứ  $n$ .

$$P_n = 1 - e^{-G} \quad n-1 \quad e^{-G}$$

Số lần truyền trung bình 1 gói tin:

$$\begin{aligned}
 \bar{n} &= \sum_{n=1}^{\infty} n P_n = \sum_{n=1}^{\infty} n (1 - e^{-G})^{n-1} e^{-G} \\
 &= e^{-G} \sum_{n=1}^{\infty} n (1 - e^{-G})^{n-1} \\
 &= e^{-G} \frac{1}{1 - (1 - e^{-G})} = e^G
 \end{aligned}$$

Vậy:  $E = \bar{n} - 1 = e^G - 1$

⇒ Độ trễ trung bình của 1 gói tin trong hệ thống Slotted\_Aloha là:

$$\begin{aligned}
 D_{S\_ALOHA} &= \frac{1}{2} + T_u \cdot E + 1 + R \text{ (khe)} \\
 &= \frac{1}{2} + \left(R + \frac{k+1}{2}\right) (e^G - 1) + 1 + R \text{ (khe)}.
 \end{aligned}$$

### 2.3 Họ giao thức CSMA.

Giao thức CSMA còn được gọi là giao thức LBT (Listen Before Talk – Nghe trước khi nói). Một trạm có dữ liệu cần truyền trước hết phải "nghe" xem phương tiện truyền rồi hay bận. nếu rồi thì bắt đầu truyền tin, còn nếu bận thì thực hiện 1 trong 3 giai thuật sau:

Giải thuật "Non-persistent"

Giải thuật "1-persistent"

Giải thuật "p-persistent"

#### 2.3.1 Giao thức CSMA trong trường hợp nonpersistent:

##### a. Nguyên lý hoạt động:

Khi một trạm có tin để truyền (trạm sẵn sàng truyền tin) thì cảm nhận kênh truyền theo thủ tục sau:

- + Nếu kênh truyền rồi thì tiến hành truyền tin.
- + Nếu kênh truyền bận, trạm đó sẽ làm thủ tục “ngoan cố”.
- + Trong trường hợp trên đường truyền có nhiều hơn một gói tin thì các gói tin đó bị xung đột. Khi có xung đột thì các trạm có gói tin bị xung đột chờ một thời gian ngẫu nhiên rồi tiến hành phát lại (thuật toán Backoff).

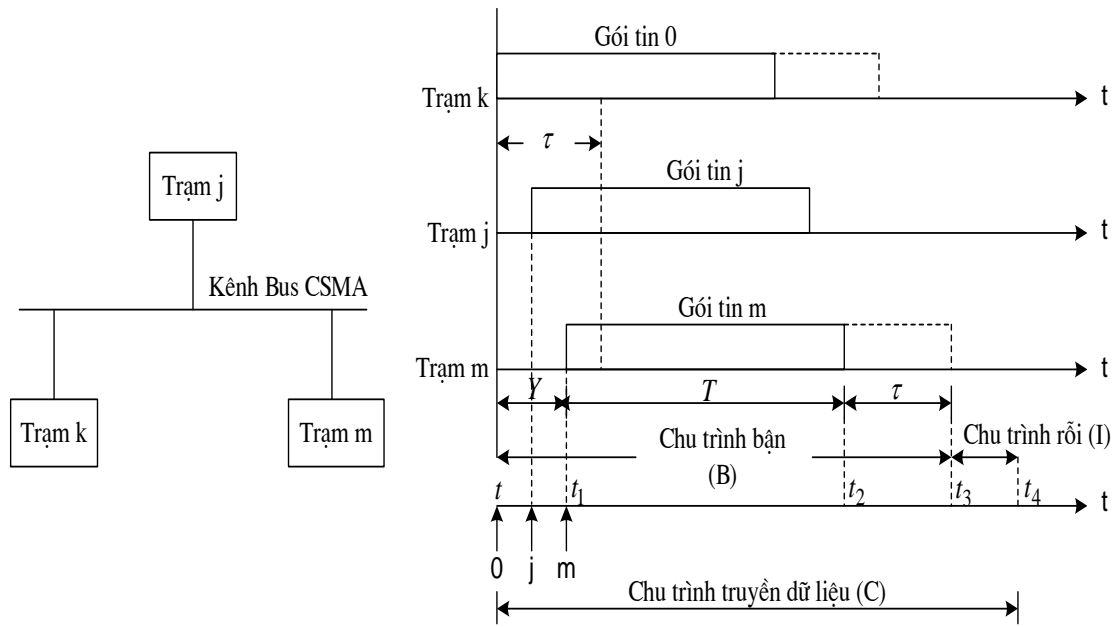
## **b. Thông lượng**

Giả sử các gói tin có cùng thời gian truyền là  $T$  giây và các gói tin tới tuân theo phân bố Poisson.

Gọi:

- +  $S$  là thông lượng của kênh truyền
- +  $G$  là lưu lượng của kênh truyền
- +  $\tau$  độ trễ truyền dẫn từ đầu cuối tới đầu cuối.
- +  $\lambda$  là tốc độ dữ liệu tới,  $\lambda = \frac{G}{T}$

Xét một mạng có topo hình BUS, sử dụng giao thức đa truy cập nonpersistent CSMA như sau:



+ Ta có thấy rằng, trạm k là trạm đầu tiên được truyền và trạm k truyền gói tin 0 ở thời điểm  $t$  (kênh truyền rỗi ở thời điểm  $t$ ). Nên sau khoảng thời gian  $\tau$  giây sẽ tới tất cả các trạm còn lại trên mạng. Do đó, trong khoảng thời gian  $\tau$  giây này nếu các trạm j và trạm m không cảm nhận được kênh truyền bận và nếu các trạm này thực hiện việc truyền tin thì các gói tin j và gói tin m sẽ gây xung đột với gói tin 0.

+ Gói tin m là gói tin cuối cùng được truyền đi ở thời điểm  $t_1 = t + Y$ , với  $0 < Y < \tau$ . Chú ý rằng, sau thời điểm  $t + \tau$ , bất cứ trạm nào có nhu cầu truyền tin sẽ cảm nhận được kênh truyền bận. Nói cách khác, kênh truyền sẽ bận trong khoảng thời gian từ thời điểm  $t$  đến  $t_3 = t + Y + T + \tau$ .

+ Khi trạm m truyền xong dữ liệu, lúc này không có trạm nào truyền dữ liệu nên kênh truyền rỗi trong khoảng thời gian từ  $t_3$  đến  $t_4$ . Sau thời điểm  $t_4$  hệ thống lại bắt đầu một chu trình mới.

Vậy, ta có:

+  $B = t_3 - t = Y + T + \tau$  là khoảng thời gian bận.

+  $I = t_4 - t_3$  là khoảng thời gian rồi.

+  $C = B + I$  là chu trình truyền dữ liệu

Mặt khác, thông lượng của hệ thống được cho bởi công thức:

$$S = \frac{\text{thời gian trung bình để phát tin thành công}}{\text{thời gian trung bình của chu trình}} = \frac{\bar{T}}{\bar{C}} = \frac{\bar{T}}{\bar{B} + \bar{I}}$$

- Tính  $\bar{T}$

Ta có  $\bar{T} = T \cdot \gamma$  (với  $\gamma$  là xác suất truyền thành công một gói tin). Mặt khác ta có xác suất truyền thành công một gói tin chính bằng xác suất trong khoảng thời

$t \div t + \tau$  không có gói tin nào tới, nên:  $\gamma = e^{-\lambda \tau} = e^{-\frac{G\tau}{T}}$

Vậy, ta có:  $\bar{T} = T \cdot e^{-\frac{G\tau}{T}}$

- Tính  $\bar{B}$

Ta có:  $B = Y + T + \tau$ .

Do đó:  $\bar{B} = \bar{Y} + T + \tau$

Do biến ngẫu nhiên  $Y$  có phân bố ngẫu nhiên trong khoảng thời gian  $[0; \tau]$ ,

nên:  $\bar{Y} = \int_0^{\tau} yp(y)dy$

Mặt khác ta có:  $p(y) = \frac{\partial F_Y(y)}{\partial y}$

Mà:  $F_Y(y) = p[Y \leq y]$ , với  $y \in [0; \tau] = p[\text{trong khoảng thời gian từ}$

$t + Y \rightarrow t + \tau$  không có gói tin nào tới]  $= \frac{(\lambda t)^k}{k!} e^{-\lambda t}$ ; với  $t = \tau - Y = e^{-\lambda(\tau - Y)}$

$$\text{Suy ra: } F_Y(y) = e^{-\lambda(\tau-Y)} \Rightarrow p(y) = \lambda e^{-\lambda(\tau-Y)}$$

$$\text{Nên: } \bar{Y} = \int_0^{\tau} y \lambda e^{-\lambda(\tau-Y)} dy = \tau - \frac{1}{\lambda} + \frac{e^{-\lambda\tau}}{\lambda}$$

$$\text{Từ đây ta có: } \bar{B} = \tau - \frac{1}{\lambda} + \frac{e^{-\lambda\tau}}{\lambda} + T + \tau$$

- Tính  $\bar{I}$

Ta có thời gian rỗi trung bình của hệ thống bằng thời gian trung bình giữa các lần tới và bằng:  $\frac{1}{\lambda}$ . Suy ra:  $\bar{I} = \frac{1}{\lambda}$

$$\text{Vậy: } \bar{C} = \bar{B} + \bar{I} = \tau - \frac{1}{\lambda} + \frac{e^{-\lambda\tau}}{\lambda} + T + \tau + \frac{1}{\lambda}$$

$$\text{Suy ra: } S = \frac{T \cdot e^{-\frac{G\tau}{T}}}{T + 2\tau + \frac{T}{G} e^{-\frac{G\tau}{T}}}$$

$$\text{Đặt } a = \frac{\tau}{T} \text{ (chuẩn hoá thời gian truyền sóng). Suy ra: } S = \frac{e^{-aG}}{1 + 2a + \frac{1}{G} e^{-aG}}$$

$$\text{Ta thấy rằng, khi } a \rightarrow 0, \text{ tức là } \tau \ll T \text{ thì: } S = \frac{G}{G+1}$$