

# پروتکل خط ارتباطی Data Link Protocol

پروتکل High Level Data Link یا HDLC یکی از معروف ترین پروتکل های لایه پیوند داده است.  
انواع:

**SDLC**

**HDLC**

**LAPs**

**LANs**

## پروتکل High Level Data Link یا HDLC

در سال ۱۹۷۹ توسط انجمن ISO معرفی شد.

از استانداردهای لایه DataLink است.

Bit Oriented است.

کلیه پروتکل‌های Bit Oriented از این گرفته شدند. مانند:

**LAPB, LAPD, PPP**

کلیه روش‌های **Point-to-Point** , **Full-Duplex** , **Half-Duplex** , **Multipoint** را پشتیبانی می‌کند.

## پروتکل High Level Data Link یا HDLC

در سال ۱۹۷۹ توسط انجمن ISO معرفی شد.

از استانداردهای لایه DataLink است.

Bit Oriented است.

کلیه پروتکل‌های Bit Oriented از این گرفته شدند. مانند:

**LAPB, LAPD, PPP**

کلیه روش‌های **Point-to-Point** , **Full-Duplex** , **Half-Duplex** , **Multipoint** را پشتیبانی می‌کند.

انواع ایستگاه‌ها در این روش:

اولیه **Primary**: مسئولیت کنترل خطوط به عهده دارد و همواره فرمان **Command** از طرف این ایستگاه ارسال می‌گردد.

ثانویه: عموماً تحت کنترل ایستگاه اولیه بوده و فریم‌هایی که این ایستگاه‌ها ارسال می‌کنند فریم پاسخ **Respond** است و در پاسخ به فرمان است.

ترکیبی: این نوع ایستگاه‌ها ترکیبی از هر دو نوع فوق هستند، یعنی هم توانایی ارسال فرمان را دارند و هم در جواب فرمان دیگران، پاسخ ارسال می‌کنند.

انواع پیکربندی خطوط ارتباطی:

**Balanced**: در این پیکربندی دو ایستگاه ترکیبی وجود دارد که با هم در ارتباط هستند. ارتباط این دو ایستگاه به هر دو صورت **Full-Duplex** و **Half-Duplex** امکان پذیر است. پیکربندی **Point-to-Point** است.

**Unbalanced**: در این پیکربندی در یک اولیه چندین ثانویه کنترل می کند. این پیکربندی **Multipoint** است.

### NRM ( Normal Response Mode )

از پیکربندی Unbalanced استفاده میکند . در این حالت Primary شروع به ارسال Command میکند و Secondary ها در صورتی که فرمان را دریافت کنند حق پاسخ دادن دارند . در واقع در این Secondary ها از طرف Primary ، Poll میشوند تا بتوانند اطلاعات خود را ارسال کنند .

### ABM ( Asynchronous Balance Mode )

از پیکربندی Balanced استفاده میکند . در این روش هر دو ایستگاه میتوانند برای هم بصورت همزمان اطلاعات ارسال کنند . در واقع دائما در حال Poll و Select یکدیگر هستند.

### ARM ( Asynchronous Response Mode )

از پیکربندی Unbalanced استفاده میکند . در این روش Secondary ها بدون نیاز به دریافت مجوز از طرف Primary ها ، میتوانند اطلاعات را به Primary ها بفرستند.

## انواع فریم‌ها در HDLC:

- I – Frame ( Information )

از این فریم برای ارسال Data استفاده میشود . الگوی این فریم در شکل زیر آمده است :

Flag	Address	Control	Information	FCS	Flag
------	---------	---------	-------------	-----	------

- S – Frame ( Supervisor )

از این فریم جهت ارسال اطلاعات کنترلی مانند ARQ ، استفاده میگردد .

Flag	Address	Control	FCS	Flag
------	---------	---------	-----	------

- U – Frame ( Un-Numbered )

از این فریم برای ارسال اطلاعات مدیریتی استفاده میگردد . به عنوان مثال اگر بخواهیم مد کاری را از

ABM به ARM تغییر دهیم ، از این فریم استفاده میکنیم . الگوی این فریم دقیقا مانند I-Frame

میباشد با این تفاوت که اطلاعات مدیریتی در همان فیلد Information ارسال میگردد.



## فیلد Flag:

از این فیلد در فریم جهت مشخص کردن **ابتدا و انتهای** فریم استفاده می‌گردد.  
یک الگوی ۸ بیتی به صورت زیر در نظر می‌گیرد:

**Flag Pattern:** ◊۱۱۱۱۱◊

هرگاه این الگو در گیرنده دریافت شود به مفهوم شروع یا پایان یک فریم است.  
این الگو نباید در بین داده در حال ارسال وجو داشته باشد.

چرا؟ کامپیوتر گیرنده دچار **اشتباه شده** و تصور می‌کند که فریم تمام شده است.

جهت رفع این مشکل از الگوریتم **Bit Stuffing** استفاده می‌شود و باعث می‌شود  
در بین داده‌های ارسالی این الگو بوجود نیاید.

## عملکرد این الگوریتم **Bit Stuffing**:

شش بیت "یک" متوالی در فرستنده.

نکته:

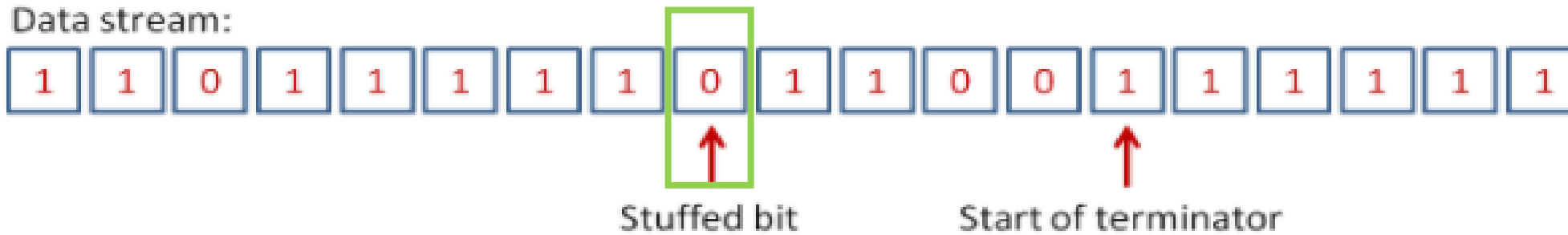
**Idle** بودن خط در این پروتکل، با **High** کردن خط، پیاده سازی می گردد.

**فرستنده:**

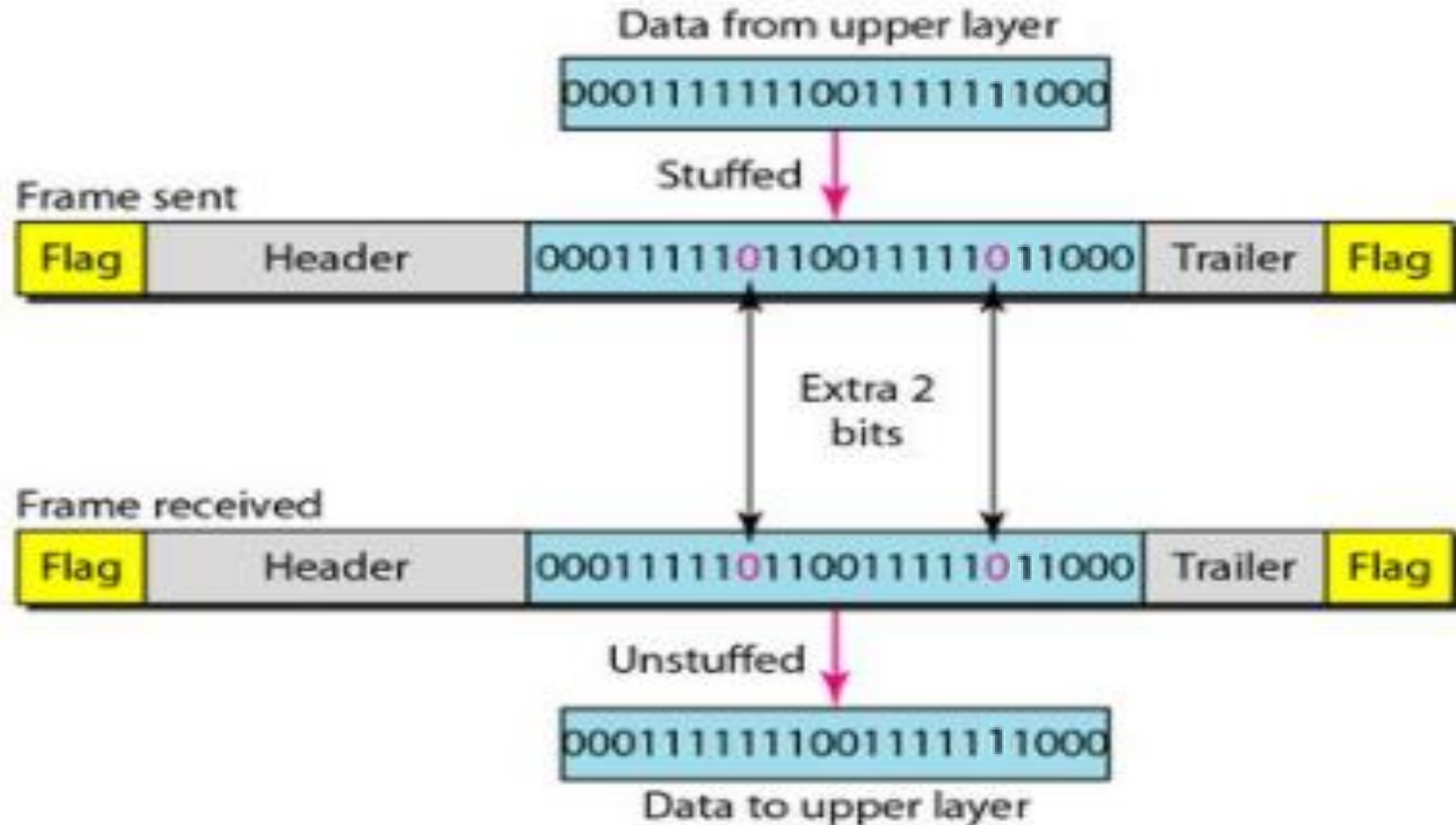
Original data:



Data stream:

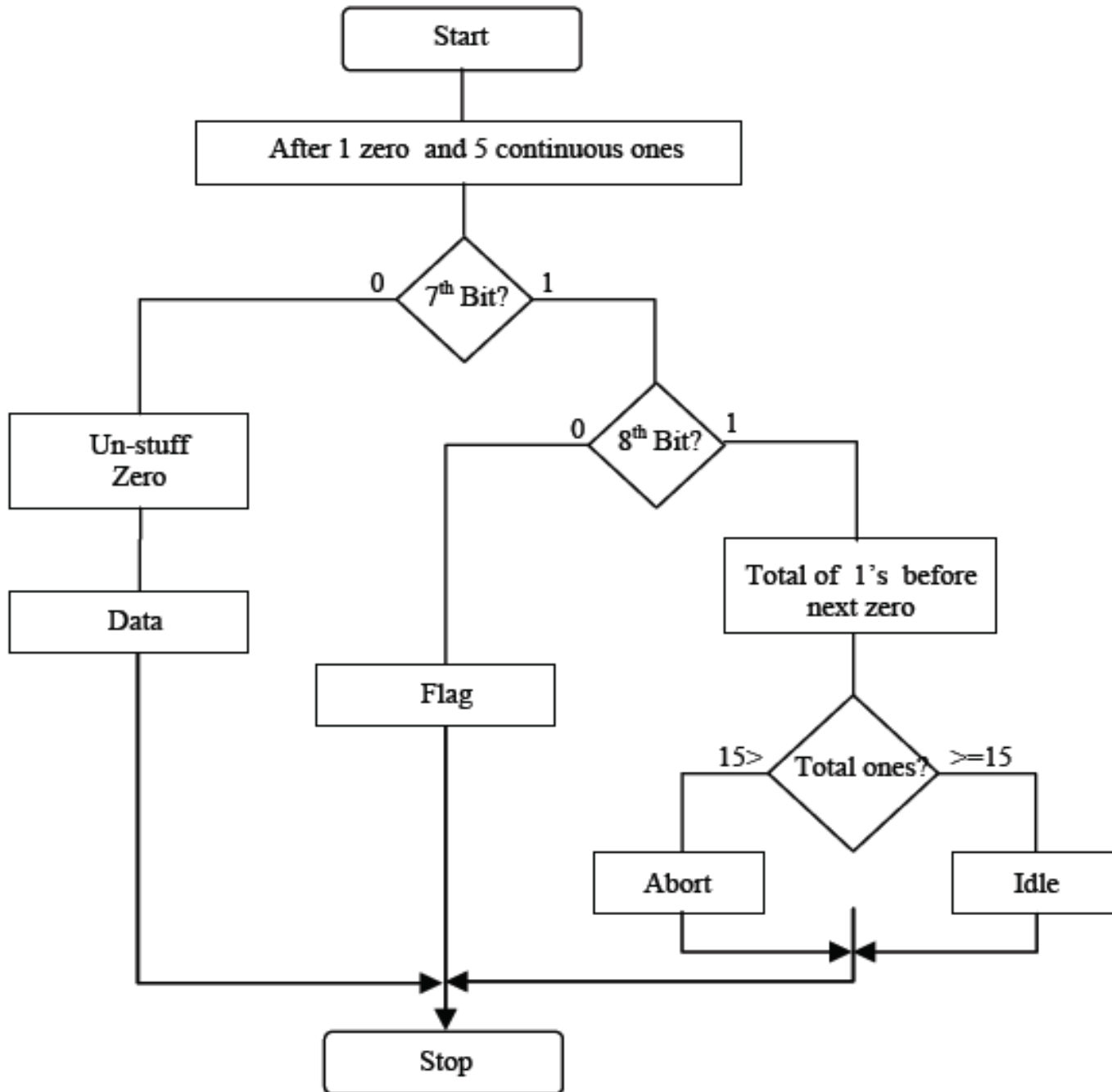


# مثال الگوریتم Bit Stuffing



# عملکرد این الگوریتم Bit Stuffing

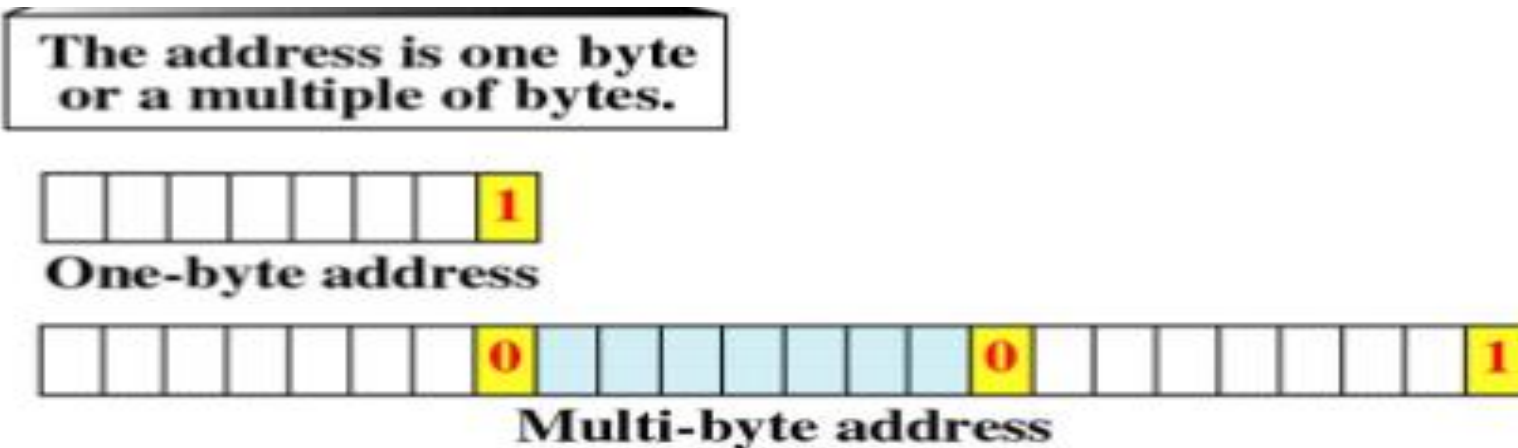
درگیرنده: ←



# الگوی فیلد آدرس:

در حالت پیش فرض طول این فیلد ۸ بیت است که **هفت بیت** برای آدرس است. اگر ارتباط به صورت **Point-to-Point** باشد همواره در این فیلد صفر قرار داده شده و بیت هشتم "۱" می شود.

اگر ارتباط به صورت **Multi-address** باشد، آدرس گیرنده در این فیلد قرار داده می شود و بیت هشتم صفر شده، و آدرس بعدی قرار داده شده و در نهایت در بیت هشتم آدرس آخرین گیرنده، "۱" قرار داده شده و ارسال می گردد.



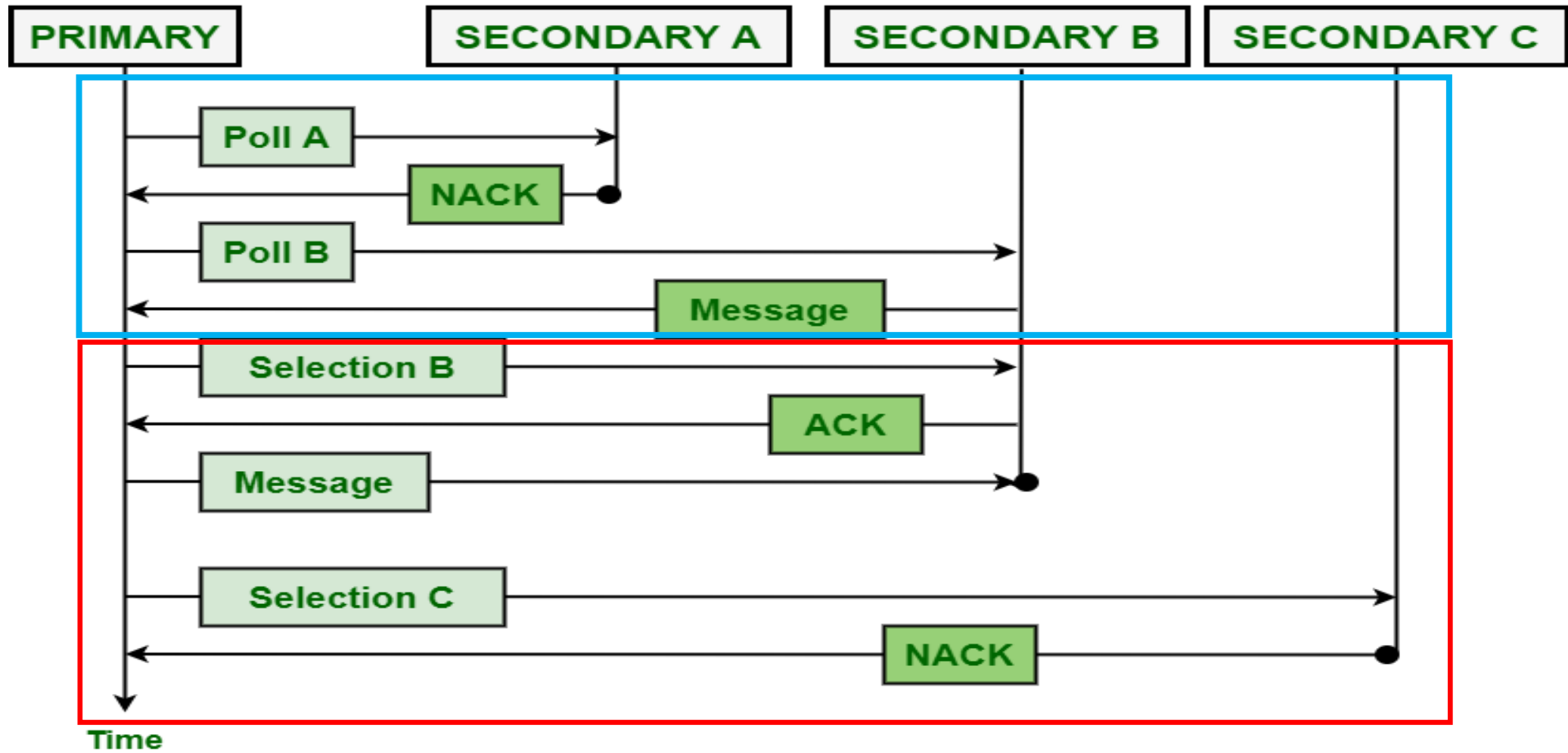
## مفهوم **Selecting** (جهت ارسال) و **Polling** (جهت دریافت داده)

All data communication must be made through the primary device

If the **primary wants to receive** data, it asks the secondaries if they have anything to send; this function is called **polling**.

If the **primary wants to send** data, it tells the target secondary to get ready to receive; this function is called **selecting**

# مفهوم Selecting (جهت ارسال) و Poll (جهت دریافت داده)



**Poll/Select**

# الگوی فیلد کنترل:

I: Information  
 S: Supervisory  
 U: Unnumbered

1	2	3	4	5	6	7	8
0	N(S)			P/F	N(R)		
1	0	S		P/F	N(R)		
1	1	M		P/F	M		

N(S) = Send sequence number  
 N(R) = Receive sequence number  
 S = Supervisory function bits  
 M = Unnumbered function bits  
 P/F = Poll/final bit



**Code N(R)**

<u>Code</u>	<u>Command</u>
<b>00</b>	<b>RR</b> Receive ready
<b>01</b>	<b>REJ</b> Reject
<b>10</b>	<b>RNR</b> Receive not ready
<b>11</b>	<b>SREJ</b> Selective-reject



## الگوی فیلد کنترل:

چنانچه کامپیوتر Secondary از طرف Primary ، Poll ، شود بیت PF به "۱" ست میشود و چنانچه کامپیوتر Secondary در حال ارسال پاسخ به Primary باشد این بیت را "۰" میکند. این بیت تنها زمانی که آخرین فریم از سوی Secondary در حال ارسال است دوباره "۱" یک میشود که به مفهوم Final است .

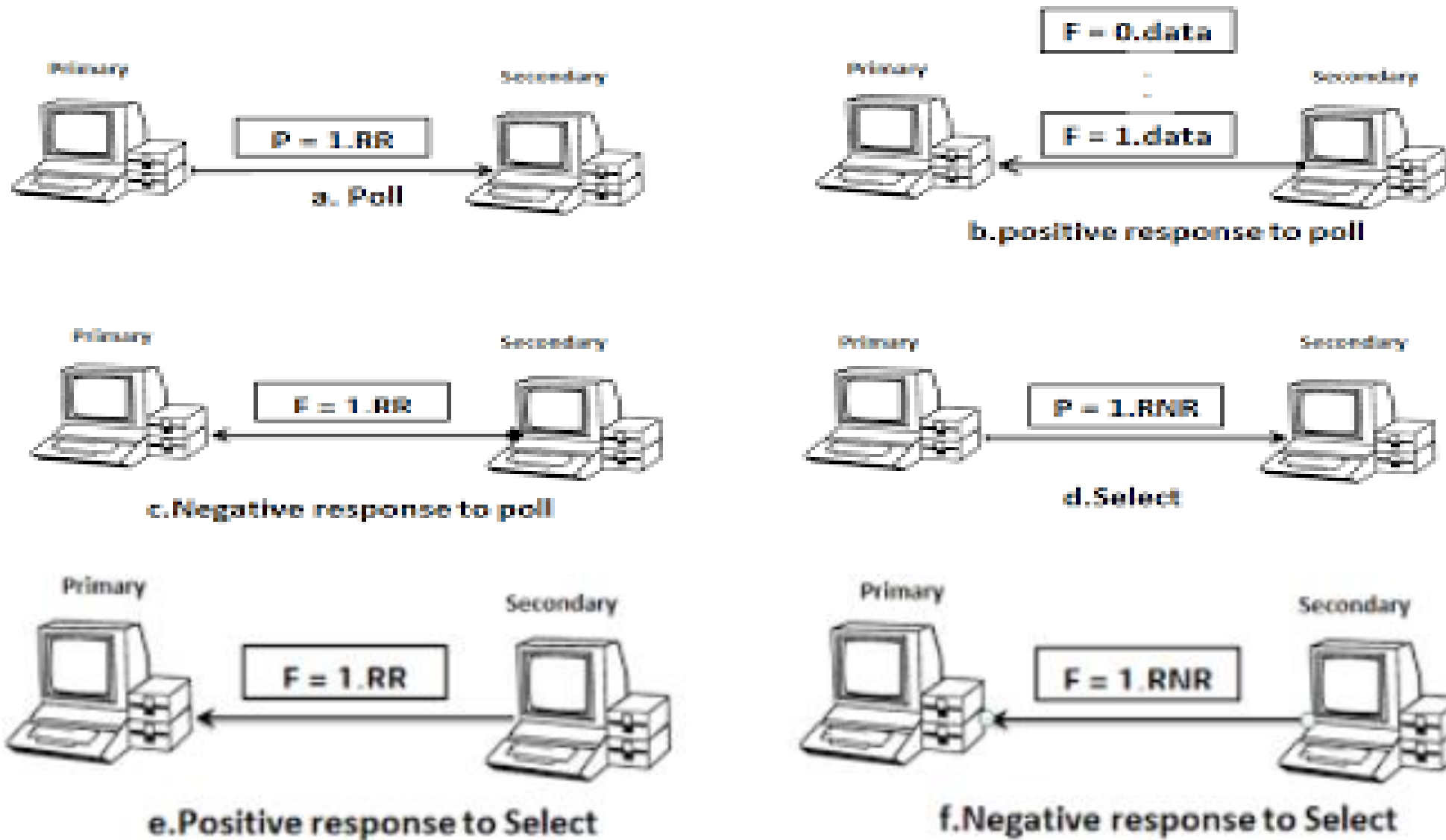
سه بیت Sequence Number ، N(S) فریمی که در حال ارسال است را مشخص میکند

Table: Types of S-frame

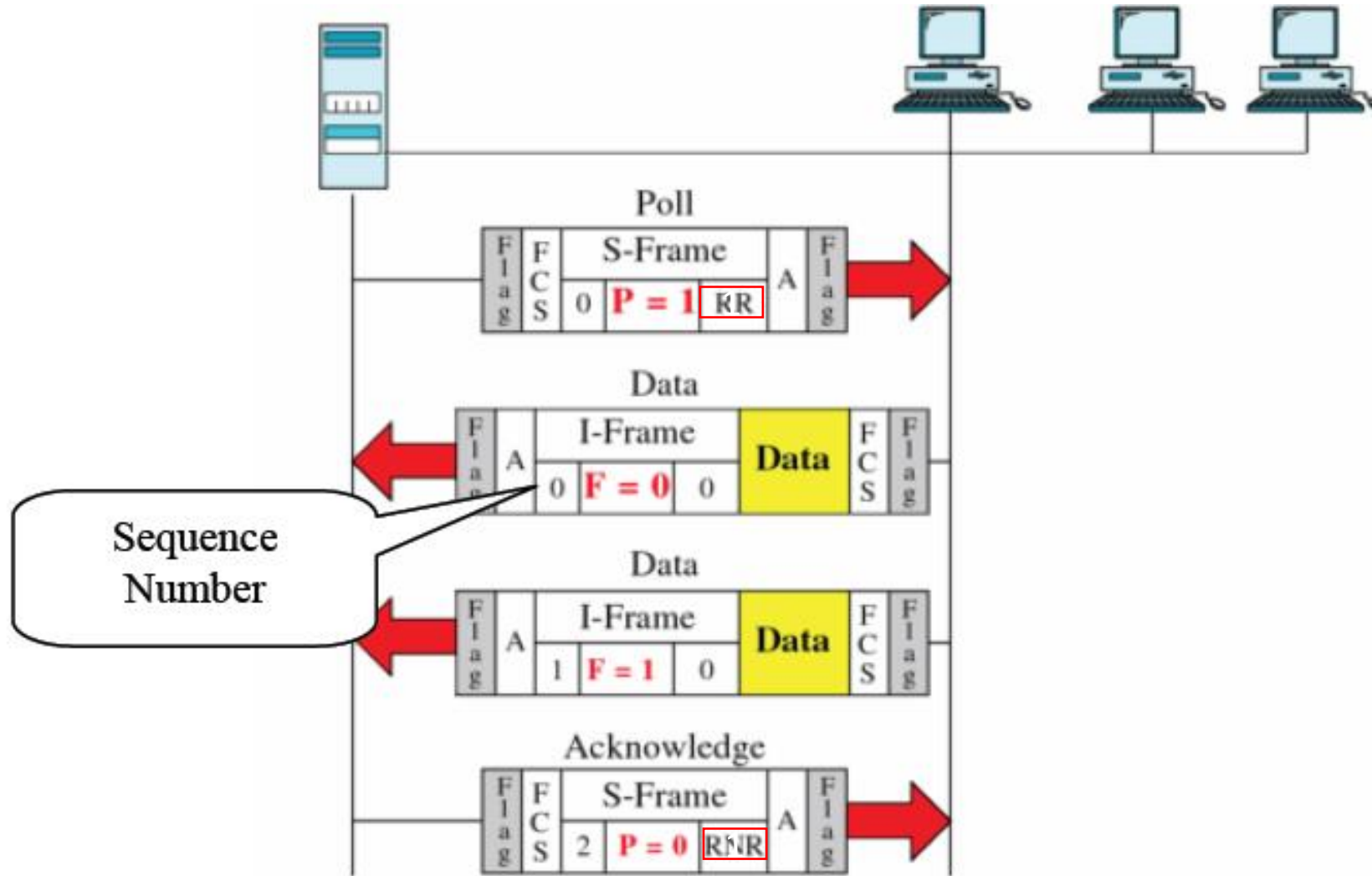
Code	Command
00	RR Receive Ready
01	REJ Reject
10	RNR Receive Not Ready
11	SREJ Selective Reject

بیت های Code در S-Frame

# کدهای Selecting (جهت ارسال) و Polling (جهت دریافت داده)



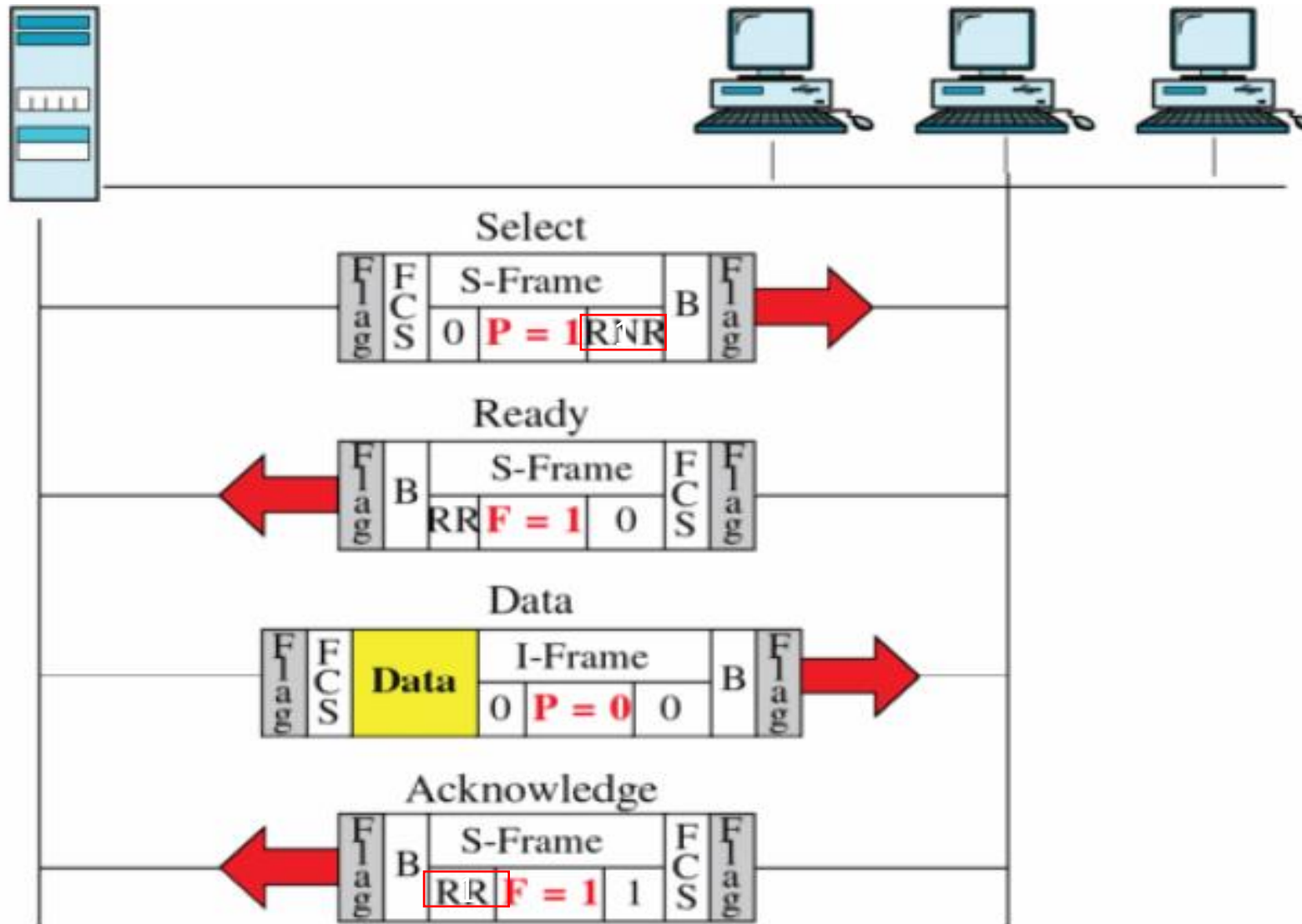
# مفهوم Poll (جهت دریافت داده):



## مفهوم Polling (جهت دریافت داده):

ابتدا فریم Poll برای A ارسال میگردد بنابر این بیت PF یک شده است ، در این فریم بیت Code به RR ست شده است که به معنی آماده بودن Primary جهت دریافت دیتا از ایستگاه A است. کامپیوتر A در پاسخ به Primary شروع به ارسال فریم ها میکند و البته بیت PF را نیز "۰" میکند. در فریم آخر بیت PF را دوباره "۱" میکند که این بار مفهوم Final دارد. Primary باید جوابی را ارسال کند که یک S-Frame است . RNR معنی ACK را دارد و با ارسال Sequence Number شماره ۲ ، میگوید ۲ فریم را با موفقیت گرفته است و منتظر دریافت فریم با شماره ۲ است . همچنین P را نیز به "۰" ست کرده است ، یعنی اطلاعات را سالم گرفته است ( این در واقع یک تایید از طرف Primary است و نه درخواست ارسال اطلاعات ) .

# مفهوم Selecting (جهت ارسال):



## مفهوم Selecting (جهت ارسال):



## مفهوم Selecting (جهت ارسال):

در این مثال Primary میخواهد به Secondary ها دیتا ارسال کند بنابراین ابتدا فریم Select از طرف Primary برای ایستگاه B ارسال میگردد با این تفاوت که بیت Code به RNR ست شده است یعنی گیرنده آماده دریافت اطلاعات باشد. ایستگاه B در پاسخ به Primary فریم S را ارسال میکند و بیت PF را نیز "۱" میکند که به معنی تمام شدن ارسال فریم از طرف B است. همچنین بیت Code را نیز به RR ست میکند که به معنی آماده بودن B جهت دریافت دیتا از Primary است و منتظر دریافت فریم با شماره یک میباشد (  $N(R)=1$  ).

حال Primary فریم I را ارسال کرده و دیتا را به همراه این فریم به B میفرستد. دوباره یک فریم ACK از طرف B به Primary ارسال میگردد که در آن ایستگاه B برای دریافت فریم های بعدی اعلام آمادگی میکند ( بیت Code را به RR تنظیم میکند. )