

(COMMUNICATIE) **PROTOCOLLEN**

afspraken waaraan zender en ontvanger moeten voldoen
om over verbindingen ordelijk en foutloos bits uit te wisselen

4.1 Inleiding protocollen

- Datalink-protocollen
- Foutenafhandeling
- Venstermechanisme
- Datastroomcontrole
- Polling
- Protocol-efficiency

Datalink-protocollen

Datalink-protocollen betreft het ordelijk en foutloos overbrengen van bits over een verbinding tussen twee deelnemers; de **logische besturing** van de dataverbinding

Opgelet: Het vorige hoofdstuk betrof de **fysieke aspecten**. Boven datalink zijn nog **functies nodig**; bv. **routing** en **end-to-end afstemming**.

Protocolfuncties

Bit-georiënteerde protocollen

Rijen bits in blokken versturen zonder besturingstekens;
Kwestie van bits tellen !

Byte-georiënteerde protocollen

Op basis van tekens, bv. STX;
Zelden gebruikt buiten asynchrone transmissie in de PC-wereld

Overeenkomsten in datalinkprotocollen

1. Berichten en blokken indelen in frames
2. Herstel van transmissiefouten
3. Flowcontrol
4. Polling
5. Transparantie

Berichten en blokken indelen in frames

Het datalink-protocol splitst in kleinere berichten of datablokkken

In het protocol is vastgelegd:

- hoeveel bits (bytes) in een frame kunnen
- hoe begin en einde worden aangegeven enz.

Herstel van transmissiefouten

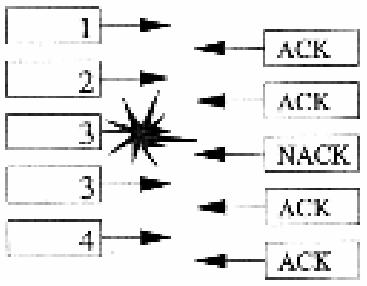
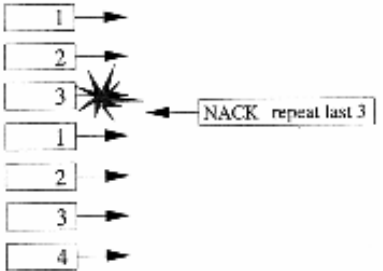
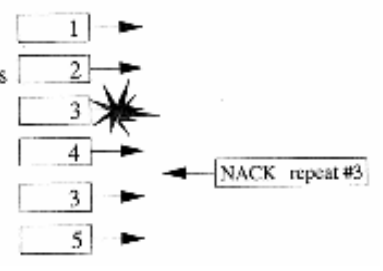
Er wordt vastgelegd:

- op welke wijze redundante informatie wordt vastgelegd
- welke pariteit of foutpolynoom wordt gebruikt
- welke procedure geldt bij het herstellen van transmissiefouten

Hertransmissie ***meest gebruikt voor herstel transmissiefouten***

ARQ Automatic Repeat reQuest

Drie verschillende hertransmissiestrategieën:

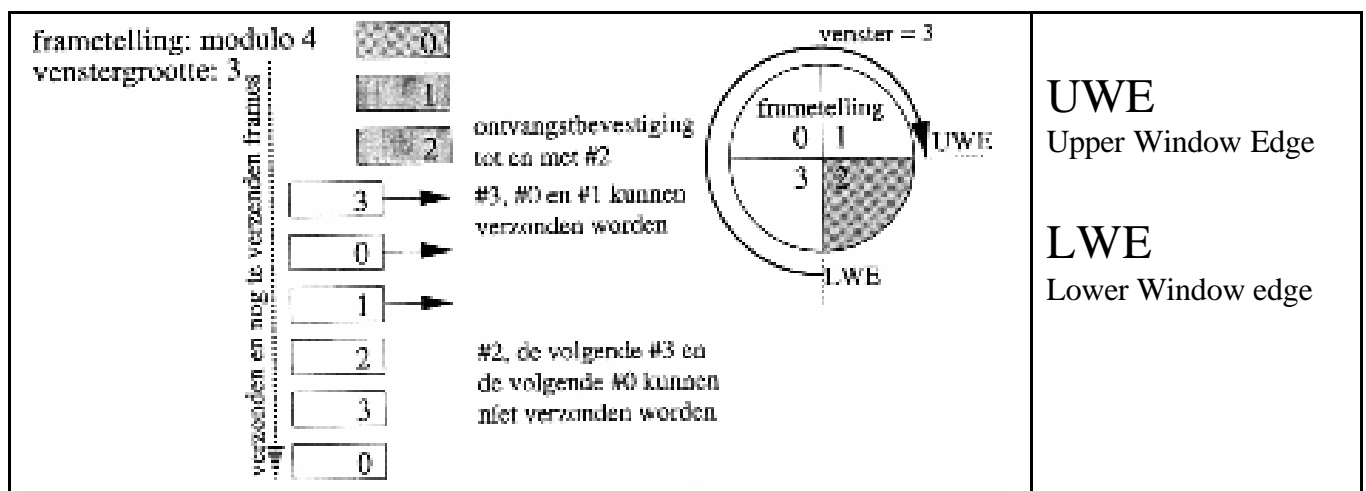
<p>stop and wait ARQ</p> 	<p>Bevestig ieder bericht; Stop-and-wait techniek</p> <ul style="list-style-type: none"> + eenvoud - het ontstaan van wachttijden door protocolmechanisme - logisch gezien een half-duplex-verbinding, => helft transmissie capaciteit wordt verspild
<p>go back N frames continuous ARQ</p> 	<p>Laatste N berichten opnieuw; Go-back_N</p> <ul style="list-style-type: none"> + bevestiging kan voor een aantal frames tegelijk + betere doorstroming (continuous ARQ) + geen opslag of volgordeproblemen - zendbuffer beperkt, # onbevestigde frames afspreken
<p>selective repeat continuous ARQ</p> 	<p>Bericht met nr # opnieuw; Selective repeat</p> <ul style="list-style-type: none"> + betere doorstroming (continuous ARQ) - opslag of volgordeproblemen

Window of Communicatievenster

Het venster of window is het mechanisme voor de protocolfunctie van het aantal onbevestigde frames.

- Alle frames krijgen een nummer (oplossing probleem volgorde)
- Door een modulo-telling voorkomt men lange nummers
vb.: Modulo 4: nummering wordt 0, 1, 2, 3, 0, 1, 2, ...
- $W = M - 1$ (venster is kleiner of gelijk dan de modulo min één) om verwarring te voorkomen.

Als de ontvangstbevestigingen sneller terug zijn kan het zenden continu doorgaan; men spreekt van sliding window.



In de praktijk gebruikt men een variëteit aan M en V-waarden, afhankelijk van het protocol en de apparatuur

Flow Control

Zorgt dat een ontvangende partij de zendende partij kan afremmen

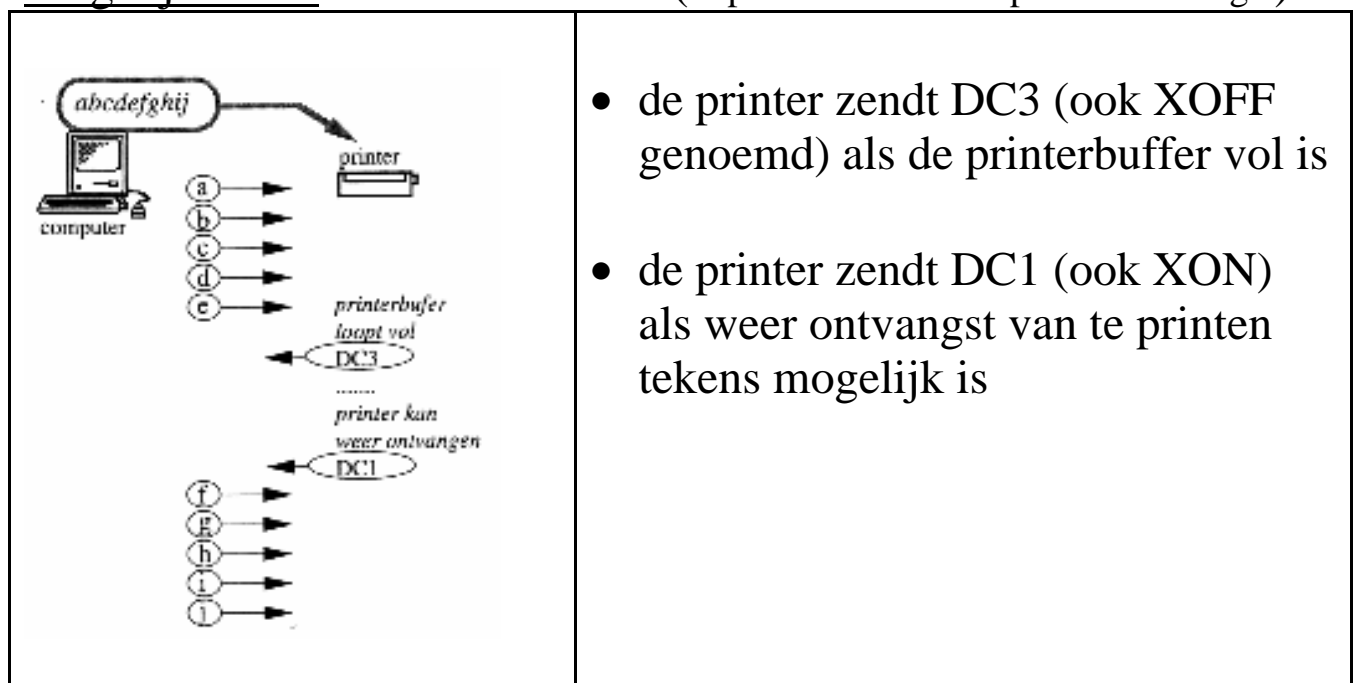
Voorbeeld:

De communicatiepoort kan sneller lopen dan een printer.

De ontvangstbuffer van de printer kan hierdoor 'overlopen'.

De printer moet de PC dus kunnen afremmen.

Mogelijkheid 1: met ASCII-code (impliceert een full-duplex verbinding !)



Mogelijkheid 2: met signalen op RS-232-C-interfaces

(hardware flow control, onafhankelijk van gebruikte code!)

Gebruik van de signalen RTS (Request-to-send) en CTS (Clear-to-send)

Mogelijkheid 3: mechanisme gebaseerd op vensters

Als de ontvanger geen ontvangstbevestiging stuurt, kan de zender niet meer verder.

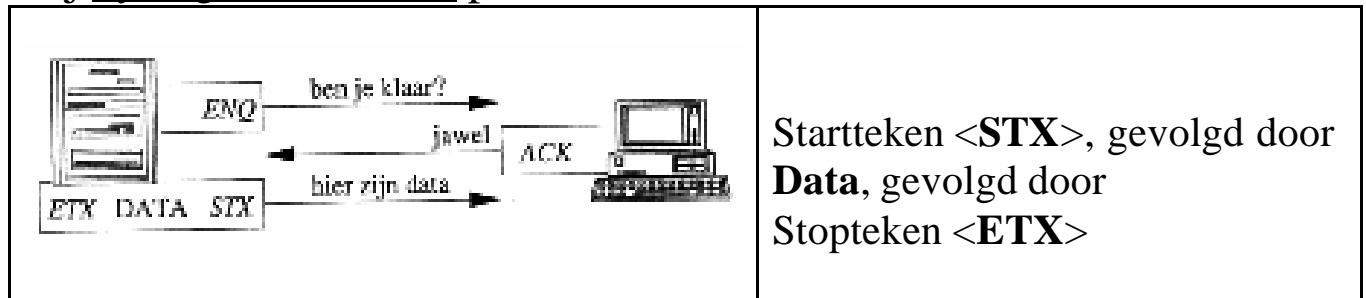
Opgelet foutenherstel en flowcontrol zijn twee verschillende protocol functies

Polzen en porren: polling

Datalink waarbij één van de systemen polst of een ander systeem:

- klaar is om data te ontvangen
- of data wil verzenden

Bij byte-georiënteerde protocollen:

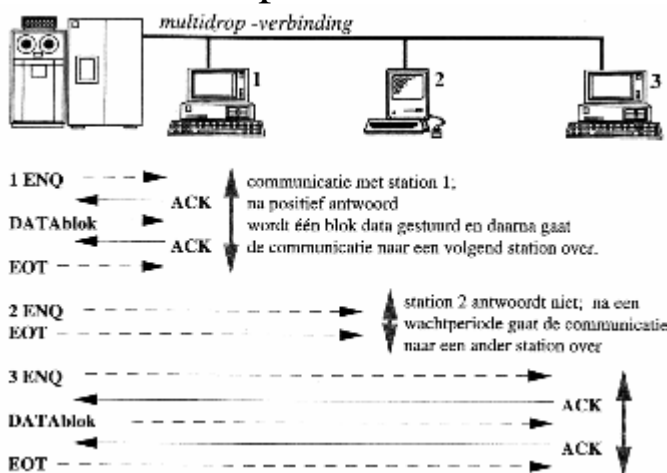


werd veel gebruikt op de datalink met meer aangesloten systemen (*multidrop* of *multipoint*-verbindingen; vb Bisync of BSC-protocol van IBM).

Met <ENQ> wordt een werkstation geselecteerd

Station antwoordt <ACK> (klaar) of <NACK> (ben er, maar niet klaar)

Als het geselecteerde station niet antwoordt binnen gestelde tijd, breekt de computer af met <EOT> en polst dan volgend station.



Master/slave prototcol: de master polst, het randapparaat antwoordt

Ook bit-georiënteerde protocollen kennen dit mechanisme, maar dan niet gebaseerd op stuurtekens.

Transparantie

Is een eigenschap, geen functie !

De datalink moet weten welke bits of bytes zijn:

- protocolinformatie ?
- data ?

Bij bitgeoriënteerde protocollen:

- een bitreeks geeft aan waar een frame begint en waar het eindigt => *** relatief eenvoudig ***
- in het protocol staat beschreven hoe de frame is opgebouwd
- probleem: de bitreeks die begin en einde van een frame aangeeft, kan niet als data gebruikt worden (bespreking later)

Bij bytegeoriënteerde protocollen

- gebruikt een groot aantal tekens (<ENQ>, <ACK>, enz.)
=> deze tekens mogen niet in de data voorkomen !
- gebruik van characterstuffing:
de zender gebruikt het besturingsteken DLE (Data Link Escape)
 - De ontvanger van een datablok zal vanaf <STX> speuren naar <ETB> of <ETX>
 - Bij een begin van een blok met <DLE> <STX> zal gespeurd worden naar <DLE> <ETB> of <DLE> <STE>
 - Een <DLE> in de tekst wordt voorafgegaan door een extra <DLE>

Protocol-efficiency

Overhead:

- Er moet naast data protocolinformatie worden overgezonden
 - De verzendende partij moet soms wachten op bevestiging
- ⇒ bij 64 kbps zal een file van 8 Kb niet in 1 sec verzonden worden.

Grote frames kunnen bijdragen tot een betere efficiency, maar bij een fout moet opnieuw een groot frame verzonden worden !

We meten de efficiëncy met **Throughput**.

Throughput is de verhouding tussen:

- de hoeveelheid verzonden data (zonder protocol overhead)
- en de tijd die dat gekost heeft.

Factoren die hierbij een rol spelen:

- Soort protocol
- Frame-lengte
- Bitfoutenkans enz.

4.2 HDLC *** hét voorbeeld ***

- Structuur van het protocol
- HDLC procedures
- NRM
- RM
- SBM

HDLC (**H**igh level **D**ata **L**ink **C**ontrol) is een bit-georiënteerd protocol) en kent vele varianten onder verschillende namen:

- LLC in LAN's
- HDLG datalink-protocollen rondom IBM-mainframes en voor digitale (ISDN)-aansluiting
- PPP voor verbinding met Internet

Ken HDLC om de geheimen van datalink te begrijpen. (is dit geen T ?)

Karakteristiek van HDLC

Gebruik:

- P2P configuraties
- Multipoint of multidrop-configuraties
- Vaste verbindingen
- Telefoonverbindingen
- Full en half-duplexverbindingen enz...

Foutendetectie:

- CRC16
- Windowing-mechanisme voor:
 - melden goede ontvangst en
 - herstel foute datablokken (Go back-N continuous ARQ-techniek)
- bevestiging van ontvangen frames door B kan met zenden van frames van B naar A ('piggy backing' => efficiency voordeel)

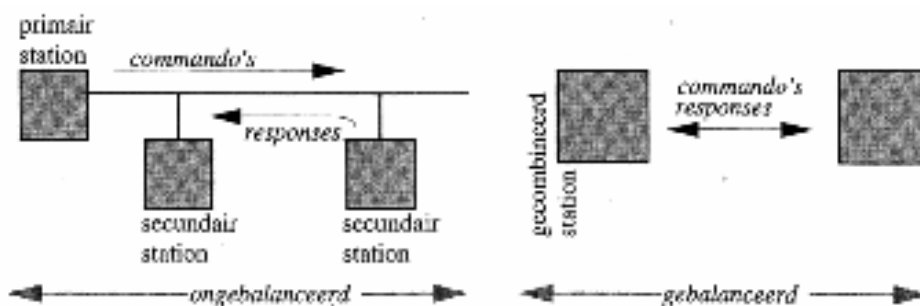
HDLC-structuur

Beschrijving op basis van 3 verschillende stations:

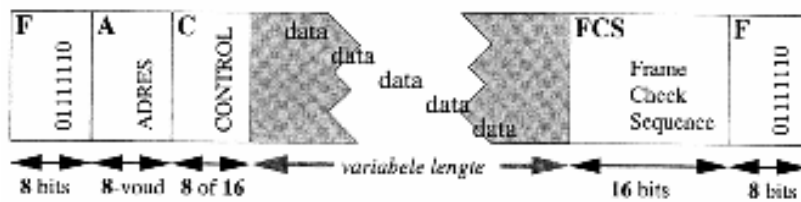
- primair: verantwoordelijk voor de verbinding
- secundair: kan alleen besturingstekens uitvoeren
- gecombineerd: kan zowel commando's als responses uitvoeren

Ongebalanceerde configuratie: een primair station kan verbonden zijn met één of meer secundaire stations (P2P of multipoint verbindingen);

Gebalanceerde configuratie: een gecombineerd station kan maar met één ander gecombineerd station worden verbonden (P2P-verbinding).



Ieder HDLC-frame heeft dezelfde structuur



Start- en sluitvlag: is een uniek bitpatroon 01111110
bitstuffing als dit patroon in andere velden voorkomt!

Het zendende station zal na een serie van 5 enen een nul toevoegen;
 het ontvangend station zal deze terug moeten verwijderen:

Zendend station		Ontvangend station	
deze reeks	wordt	deze reeks	wordt
01111110	011111 0 10	011111 0 10	01111110
0111110	011111 0 0	011111 0 0	0111110
011111110	011111 0 110	011111 0 110	011111110

Een bitreeks van een byte wordt verzonden van rechts naar links:

0	1	1	0	1	0	1	1
---	---	---	---	---	---	---	---

=> wordt verzonden: 11010110

Het adresveld heeft enkel betekenis bij ongebalancerde configuraties (multidrop); bij P2P valt niets te adresseren.

De lengte is 8 bits of een veelvoud ervan. Is de eerste bit een 0 dan volgt er nog een adres-octet.; is de eerste bit een 1 dan is dat het laatste adres octet (kettingstructuur)

Het controlfield is 8 of 16 bits lang

De lengte van het controlfield wordt vastgesteld door een commando-uitwisseling tussen zender en ontvanger (layout zie verder)

Het informatieveld bevat de data:

- is een facultatief veld, dus frames zonder data, ten behoeve van de besturing zijn mogelijk
- de lengte is niet vastgelegd
- de bitvolgorde is evenmin vastgelegd
=> zaak van de deelnemende stations

Het FCS-veld (**F**rame **C**heck **S**equen**C**e) bestaat uit de laatste 16 bits voor de sluitvlag en dient om de transmissiefouten op te sporen:

- Het is een cyclische checkcode met polynoom $X^{16} + X^{12} + X^5 + 1$
- Start en sluitvlaggen worden niet opgenomen in de berekening
- De tussengeschoven nullen worden evenmin opgenomen

Détail Control Field

Veld waarmee met behulp van commando's de verbinding (data link) wordt bestuurd

3 verschillende formaten zijn vastgelegd:

- Information frame: gebruikt voor gegevenstransport; bevat altijd gegevens
- Supervisory frame: gebruikt voor flow-control-doeleinden; bevat nooit gegevens
- Unnumbered frame: gebruikt om de verbinding te bewaken/beheren; bevat incidenteel gegevens

hit	7	6	5	4	3	2	1	0
frame	I		receive count	p/f	send count		0	
			N(R)		N(S)			
	S		receive count	p/f	functie bits		0	1
		N(R)						
U		functie bits		p/f	functie bits		1	1
<i>basis-formaat</i>								

bit	7	6	5	4	3	2	1	0	
frame	I		receive count				N(S)		0
			send count				N(R)		p/f
	S		ongebruikt			functiebits		0	1
		receive count				N(R)		p/f	
U		M1		p/f	M2		1	1	
		ongebruikt						p/f	
<i>verlengd formaat</i>									

Als er informatie wordt uitgewisseld (met een I-frame):

- Ieder I frame wordt genummerd
- Framenummers zijn mod 8 (korte versie) of mod 128 (lange versie)
- Een zender kan dus max. 7 of 127 frames onbevestigd hebben

In elk station ondersteunen twee tellers het window-mechanisme:

- N(R): de receive count; geeft framenummer welk verwacht wordt (LWE)
- N(S): de send count; geeft nummer volgend te versturen frame

In een bevestiging wordt altijd de inhoud van de receive count gebruikt, waaruit de zendende partij kan afleiden tot welk frame nummer verzonden kan worden totdat de window vol is (UWE)

HDLC-procedures

In een ongebalanceerde verbinding:

- beheert het primaire station de verbinding met commando's (C)
- de secundaire stations reageren daarop met responses (R)
- zijn er twee manieren waarop stations kunnen communiceren
 - Normal Response Mode
Secundair station mag slechts zenden na een poll van het primair station
 - Asynchronous Response Mode
Waarin een station (primair en secundair) mag zenden als de verbinding niet actief wordt gebruikt

In een gebalanceerde verbinding:

- beide stations kunnen zowel commando's als responses zenden
- de communicatie is steeds asynchronous

De P-bit komt alleen in de voor commandoframes; de F-bit alleen in de responseframe. In velden van S- en U-frames worden een aantal commando's (C) en responses (R) gecodeerd.

afkorting	naam	functie	omschrijving
I	<i>Information-frame</i>	C/R	gegevensuitwisseling
S	<i>Supervisory frames</i>		
RR	Receive Ready	C/R	'ack'; klaar voor 1-frame
RNR	Receive Not Ready	C/R	'ack'; niet klaar
REJ	REject	C/R	'nack'; go-back-N frames (ARQ)
SREJ	Selective REject	C/R	'nack'; selective repeat
U	<i>Unnumbered frames</i>		
SNRM	Set Normal Response Mode	C	set mode NRM
SNRME	SNRM Extended	C	set extended mode (C = 16 bits)
SARM	Set Asynch Response Mode	C	set mode ARM
SARME	SARM Extended	C	set mode extended
SABM	Set Asynch Balanced Mode	C	set mode ABM
SABME	SABM extended	C	set mode extended
SIM	Set Initialization Mode	C	initialiseren bij ontvanger
DISC	DIS Connect	C	stop logical link
UA	Unnumbered Acknowledgement	R	ack-response op set mode frame
DM	Disconnected Mode	R	sec-station wordt uitgebannen
RD	Request Disconnect	R	verzoek om DISC-commando
RIM	Request Initialization Mode	R	verzoek om SIM-commando
UI	Unnumbered Information	C/R	(bij besturingsinfo-transport)
UP	Unnumbered Poll	C	(bij besturingsinfo-verzoek)
RSET	ReSET	C	reset framenummers (recovery!)
XID	eXchange IDentification	C/R	status- en ID-uitwisseling
TEST	TEST	C/R	testpatroon-uitwisseling
FRMR	FRaMe Request	R	melding van foutief frame

Voorbeelden

We gaan de HDLC-mechanismen nu verduidelijken met voorbeelden.

We gebruiken daarbij een compacte notatie om de HDLC-frames te kunnen opschrijven:

<frame type> (x,y) <status P/F-bit>, <adres>

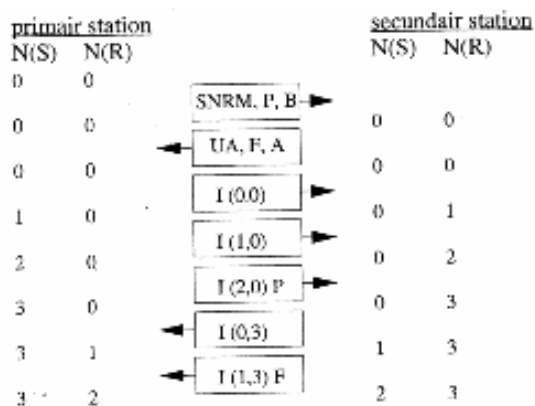
- voor frametype gebruiken we de afkortingen uit eerste kolom van de tabel HDLC frames
- x,y geeft de waarde van de tellers aan
 - x is de send count N(S)
 - y is de receive count N(R)
- P geeft aan dat de poll-bit 'aan' staat, F betekent final-bit 'aan'
- De letter geeft aan welk stationadres in het adresveld staat.

Welke kenmerken in deze verkorte schrijfwijze wel of niet voorkomen hangt af van het frametype. In een UA-frame komt bv. geen tellerstand voor

UA (unnumbered Acknowledgement / response / ack-response op set mode frame)

NRM-voorbeeld (*Normal Response Mode*)

We gaan uit van een ongebalanceerde situatie.



De zendteller loopt gelijk met de ontvangstteller

typerend voor een NRM-operatie, want het secundaire station kan slechts zenden als een poll-bit is ontvangen;

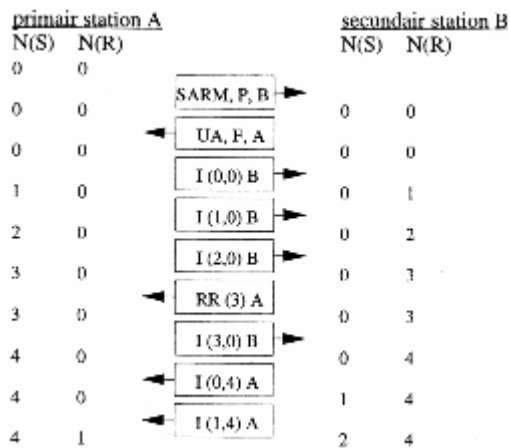
daarom kan een station geen frames bevestigen tijdens ontvangst;

hierdoor is het voor een primaire station mogelijk een multidrop-verbinding te controleren.

In feite is NRM een half-duplex protocol, ook al is de verbinding full-duplex.

ARM-voorbeeld (*Asynchronous Response Mode*)

Het secundaire station kan zenden zonder dat een poll-bit moet ontvangen worden. Een full-duplex wordt daadwerkelijk gebruikt.



Omdat het primair station moet weten welk secundair station frames zendt, zal het slecht één secundair station in ARM-mode toelaten (het secundair station kan dan actief zijn).

De betekenis van P/F-bit verandert hierdoor in een ARM-relatie: een P-bit dient enkel om de status op te vragen van een secundair station, dat altijd reageert met een F-bit.

Het RR-frame (receive ready) is een supervisory frame (doel: flow control)

- De receive count wordt overgebracht
- De verzender van het RR-frame geeft aan dat ontvangst van I-frames mogelijk is.

De tegenhanger is het RNR-frame (**R**eceive **N**ot **R**eady). Er kunnen geen frames ontvangen worden.

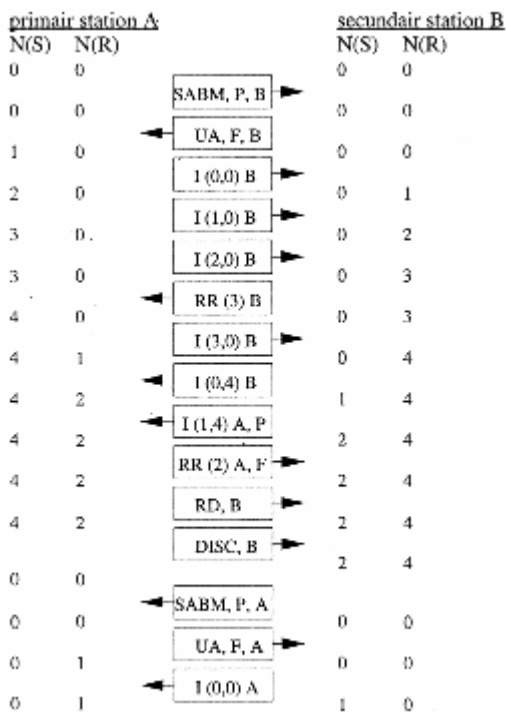
ABM-voorbeeld (*Asynchronous Balanced Mode*)

Het belangrijkste HDLC-protocol tussen twee gelijkwaardige, gecombineerde stations.

Bij protocolbotsing zijn er standaardprocedures voorzien.

In het voorbeeld verzendt station B het eerste I-frame [I(0,4)B] en meldt gelijktijdig ontvangst van frame 3. Piggy Backing.

Het RD-frame (**R**equest **D**isconnection) dat station B moet versturen beëindigt de verbinding en leidt tot resetten van de tellers.



Flow control in HDLC

Er bestaan twee mechanismen om de datastroom op de link te beheersen.

1. het supervisory frame RNR geeft aan dat de zender niet in staat is om informatie te ontvangen (vgl met XON/XOF-methode om de zender af te remmen)
2. Daarnaast draaien er ook vensters op een HDLC verbinding, de receive counters. Als een window volloopt, zal de zender geen I-frames meer zenden.