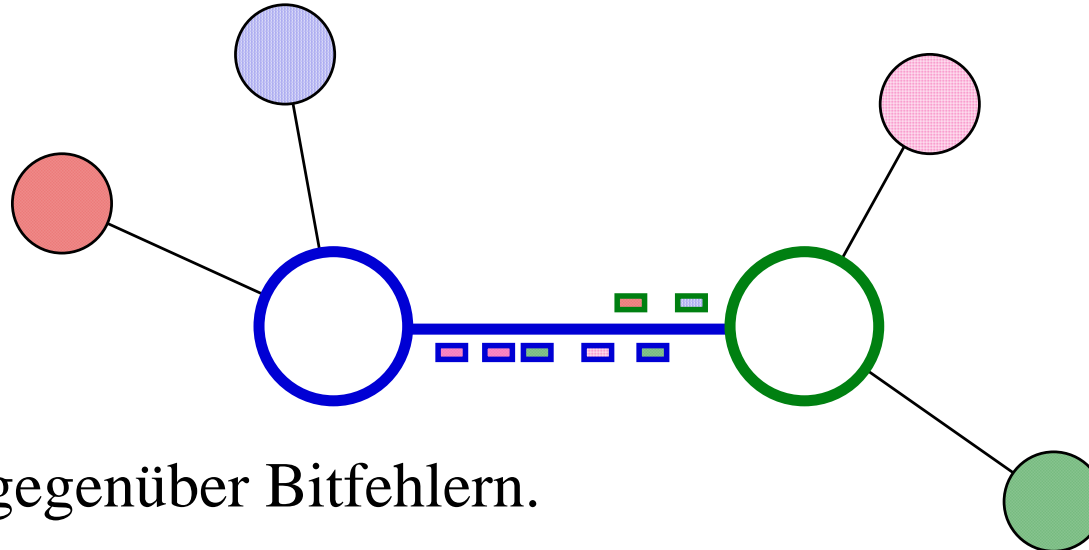


7. Sicherungsprotokolle

7.1. Grundsituation

7.1.1 Punkt-zu-Punkt-Szenarium

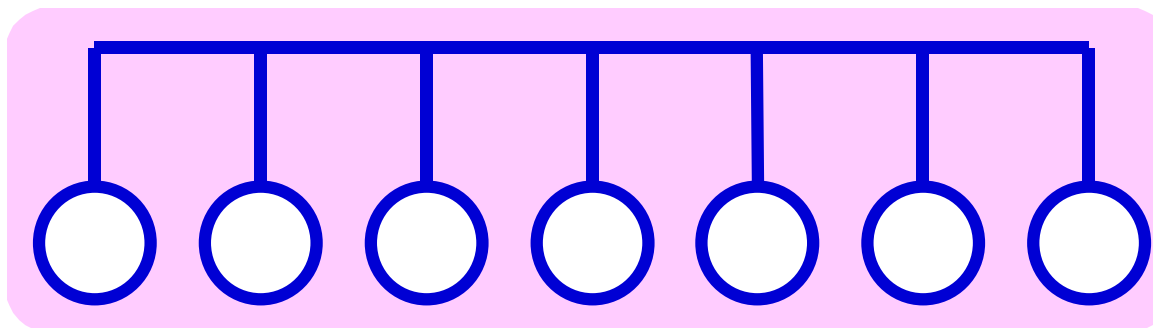
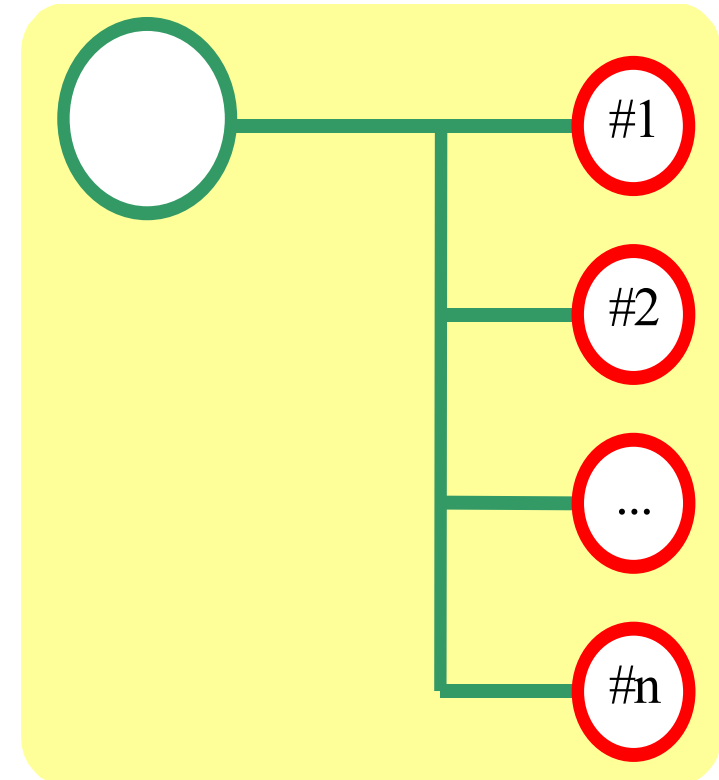
- Zwei Knoten sind direkt durch einen "physikalischen" Übertragungskanal ("Data-Link") verbunden und tauschen Nachrichten aus:



- Absicherung gegenüber Bitfehlern.
- Es dürfen keine Meldungen verloren gehen.
- Das Weiterleiten von Meldungen ist nicht Aufgabe der Sicherungsschicht, sondern wird von der Netzwerkschicht erledigt.

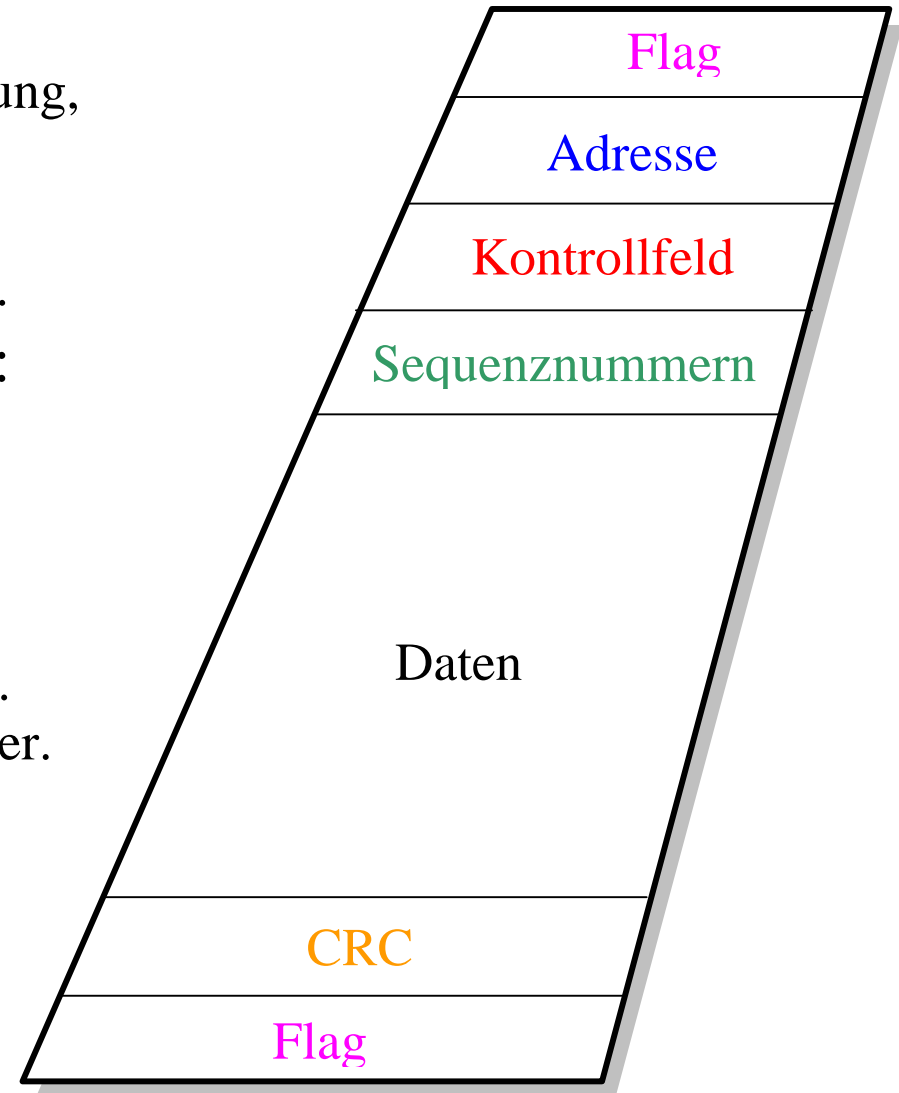
7.1.2 Mehrere Stationen pro Link

- Adressierung der Stationen erforderlich.
- Die Stationen erkennen an sie gerichtete Meldungen anhand der Adresse.
- Asymmetrisches Szenarium:
 - **mehrere Sekundärstationen (Slaves)** =>
 - eine Primärstation (Master).
- Symmetrisches Szenarium:
 - "Peer to Peer" Rollenverteilung,
 - Verschiedene Mechanismen zur Vergabe der Sendeberechtigung (MAC-Sublayer).
 - Viele gleichberechtigte Stationen am einem **Übertragungsmedium (LAN)**:



7.1.3 Exemplarisches Rahmenformat

- "Link-Layer Protocol Data Unit" / DL_PDU.
- **Flag:**
 - Oktettsynchronisierung & Rahmenbegrenzung,
 - z.B. Bitmuster '01111110'.
- **Adresse:** Identifikation der Station
 - insbesondere in Mehrpunktkonfigurationen.
- **Control/Steuerung:** Typ der Meldung:
 - eventuell nur Informationstransport,
 - Receive not ready, Receive Ready,
 - Connect, Disconnect, Reset ...
- **Sequenznummern:**
 - um evtl. verlorene Meldungen festzustellen.
 - Bestätigungen ebenfalls mit Sequenznummer.
- **Daten:**
 - durch die Schicht nicht weiter interpretiert.
- **CRC:**
 - Prüfsumme für Bitfehler.



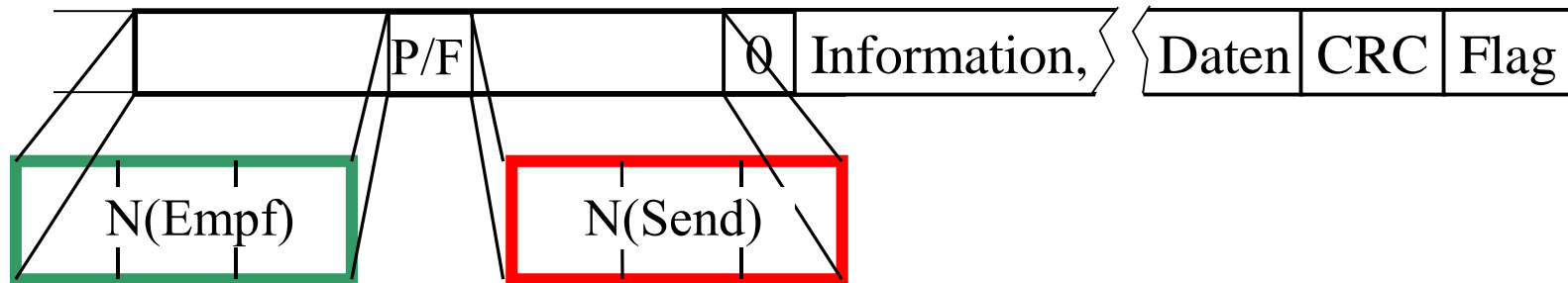
7.2. HDLC - High-Level Data Link Control

7.2.1 Allgemeines

- Synchrones Sicherungsprotokoll, entstanden aus SDLC (IBM).
- Bestehender physikalischer Übertragungskanal wird vorausgesetzt.
- Protokollelemente:
 - Sequenznummern, Flusskontrolle, Prüfsumme, Bitstuffing.
- Einsatzbereich:
 - Sicherungsschicht von X.25 (Datex-P),
 - D-Kanal im ISDN (Signalisierung),
 - LocalTalk (Macintosh),
 - B-Kanäle im ISDN.
- USART-Schaltkreise:
 - Universal Synchronous/Asynchronous Receiver/Transmitter,
 - Vielkanal HDLC Schaltkreise,
 - SCC Zilog 8530 ...,
 - ISDN Chips.
- Die UARTs imPC unterstützen kein HDLC, da keine HW für CRC ...

7.2.2 Kontrollfeld für I-Rahmen in HDLC Protokollen

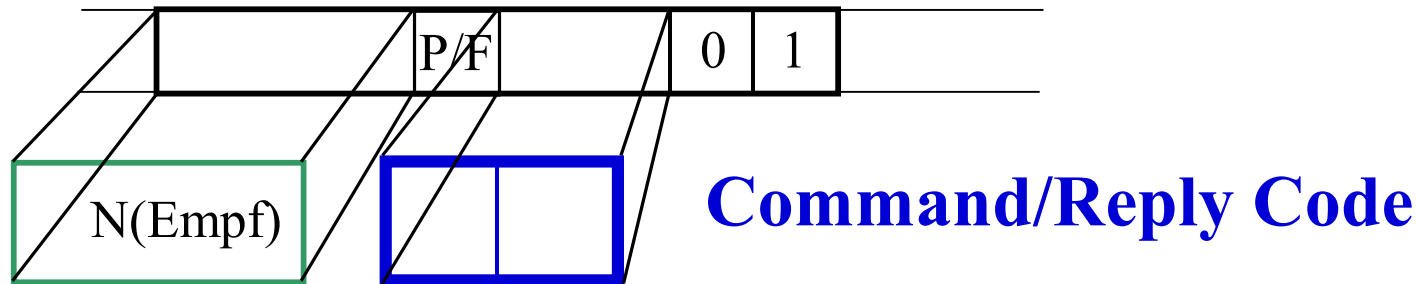
- I-Frame = Information Frame.



- Nutzdaten im Informationsfeld.
- **Sequenznummer** in Vorwärtsrichtung.
- 'Huckepack' **Bestätigung** für Gegenrichtung.
- Poll/Final Bit zum Umschalten der Übertragungsrichtung.

7.2.3 Kontrollfeld für S-Rahmen in HDLC Protokollen

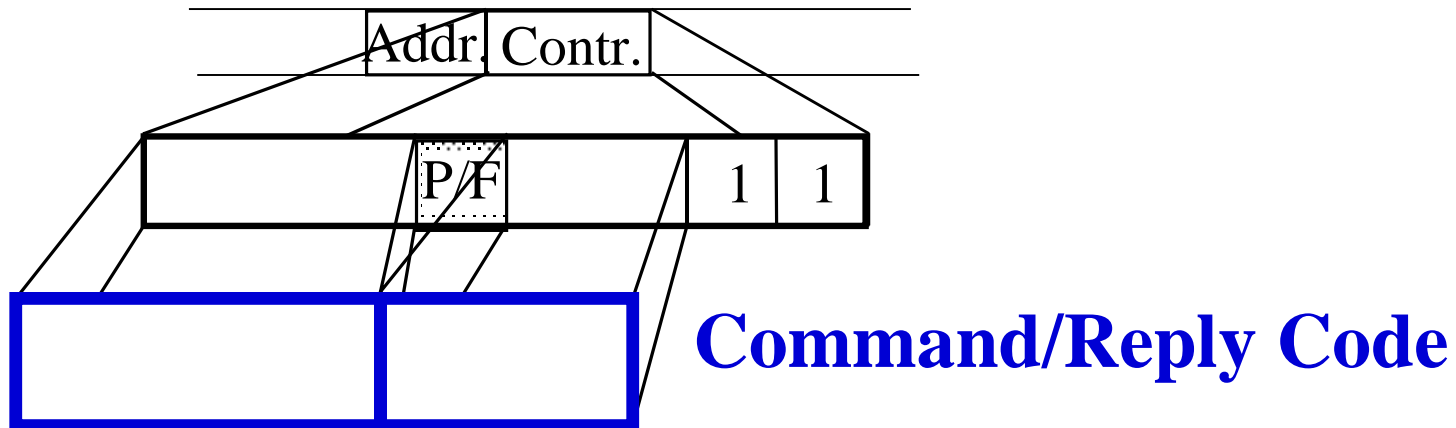
- S-Rahmen = Supervisory Frame.



00	RR	C/RReceive Ready
01	RNR	C/RReceive Not Ready
10	REJ	C/RReject ab Rahmen N(...)
11	SREJ	C/RSelective Reject (optional)

- Separate Bestätigung, falls nicht Huckepack - N(Empf).
- Anfordern von Paket-Wiederholungen.
- Explizite Flusskontrolle.

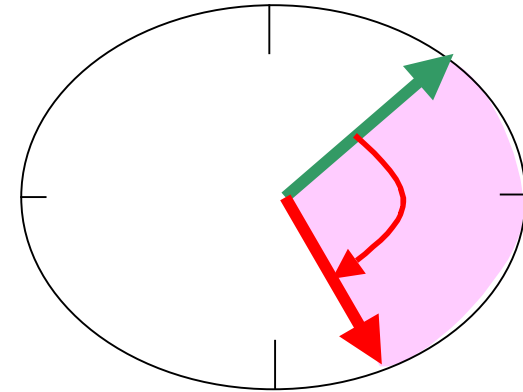
7.2.4 U-Rahmen (Unnumbered Frame)



000 01	SIM	C/-	Set Init. Mode	RIM	-/R	Request Init. Mode
000 11	SARM	C/-	Set Async. Resp. Mode	DM	-/R	Disconnected Mode
010 00	DISC	C/-	Disconnect	RD	-/R	Request Disconnect
100 01	CMDR	-/R	Command Reject	FRMR	-/R	Frame Reject
000 00	UI	C/R	Unnumbered Information			
001 00	UP	C/-	Unnumbered Poll			
011 00	UA	-/R	Unnumbered Ack			
001 11	SABM	C/-	Set Async. Bal. Resp. M.			
100 00	SNRM	C/-	Set Normal Resp. Mode			
101 11	XID	C/R	Exchange Identification			

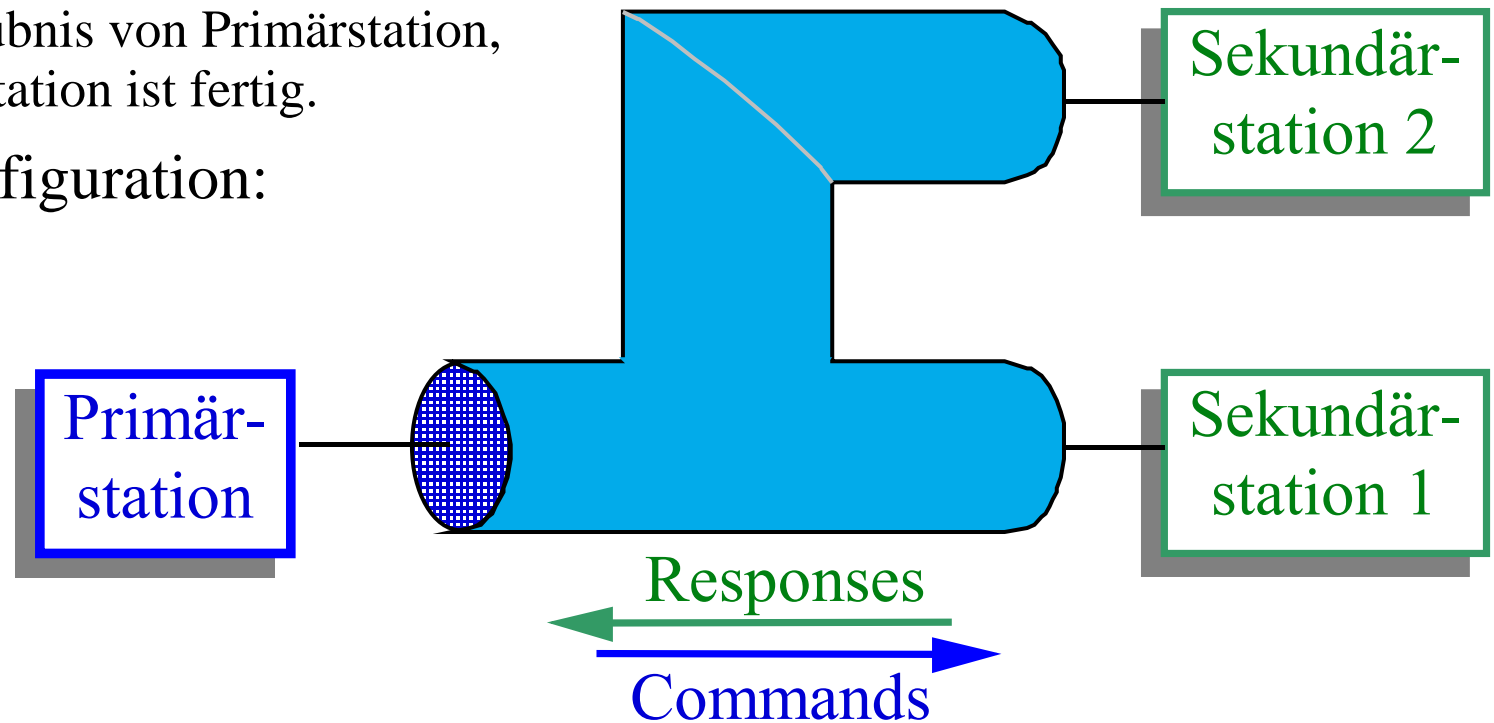
7.2.5 Sequenznummern

- Zähler für N_{Send} :
 - Sequenznummer des vorliegenden Pakets.
- Zähler für Q_{Empf} :
 - Bestätigung für Paket[$Q-1$] in Gegenrichtung,
 - Sequenznummer des erwarteten Paketes.
- Zurückhalten von Bestätigungen bremst den Datenfluss.
- Nicht mehr als W_{max} Rahmen dürfen unbestätigt sein ($W_{\text{max}} \sim 3$).
- Bestätigte Rahmen freigegeben, unbestätigte aufbewahren/wiederholen.
- Eventuell 16 Bit Kontrollfeld mit 7-Bit Sequenznummern, anstelle des 8 Bit Kontrollfeldes für 3-Bit Sequenznummern.
- Die Zählung ist Modulo 8 oder Modulo 128.



7.2.6 Ablauf im Normal Response Mode

- Ursprünglich die einzige Rollenverteilung für HDLC
- In der Vergangenheit bediente ein Zentralrechner mehrere blockorientierte Terminale.
- Sekundärstationen senden nur nach Aufforderung.
- Poll/Final Bit im Meldungskopf:
Poll: Sendeerlaubnis von Primärstation,
Final: Sekundärstation ist fertig.
- Mehrpunktkonfiguration:

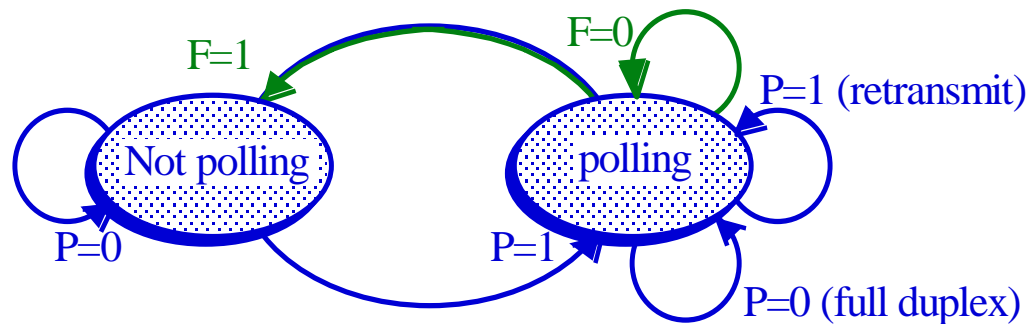


7.2.7 Einfacher Zustandsautomat für Poll/Final Bit

- Darstellung des Protokolls nicht als Petri-Net sondern als Zustandsübergänge in einem Automaten.

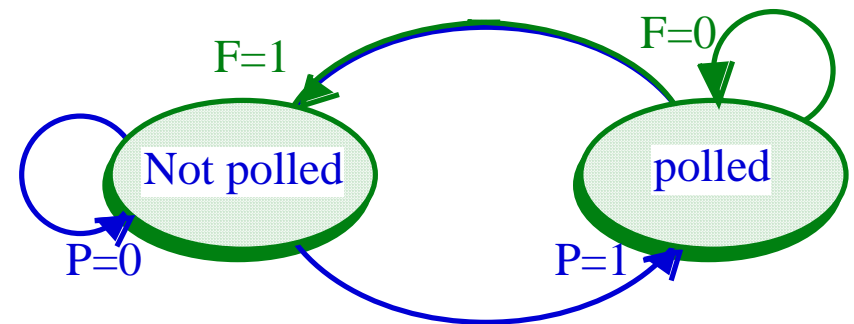
In der Primärstation:

- F empfangen,
- P senden.



In der Sekundärstation:

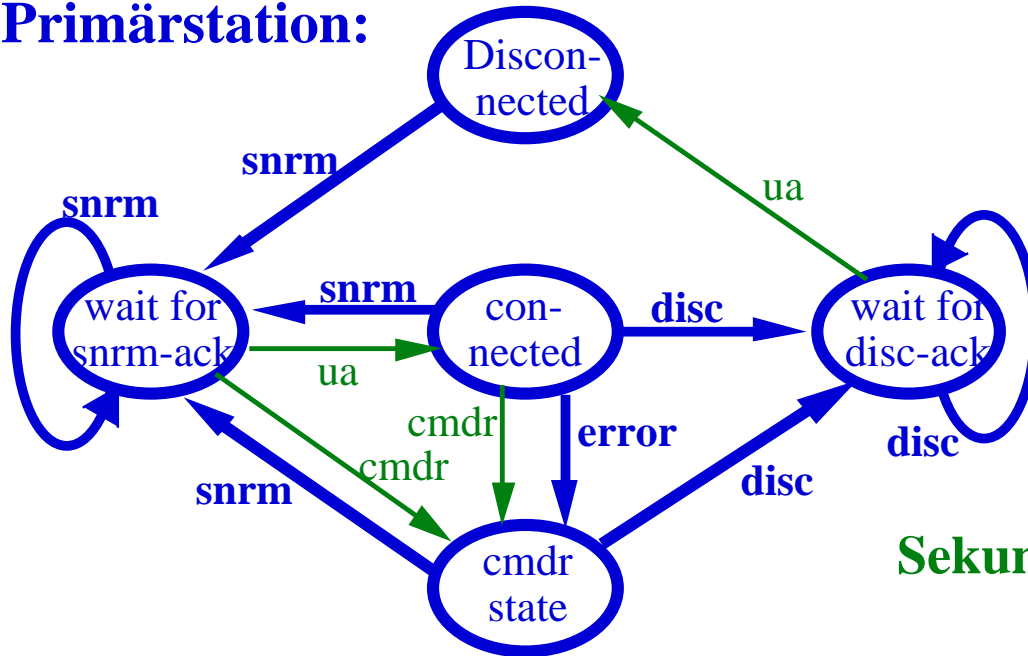
- P empfangen,
- F senden.



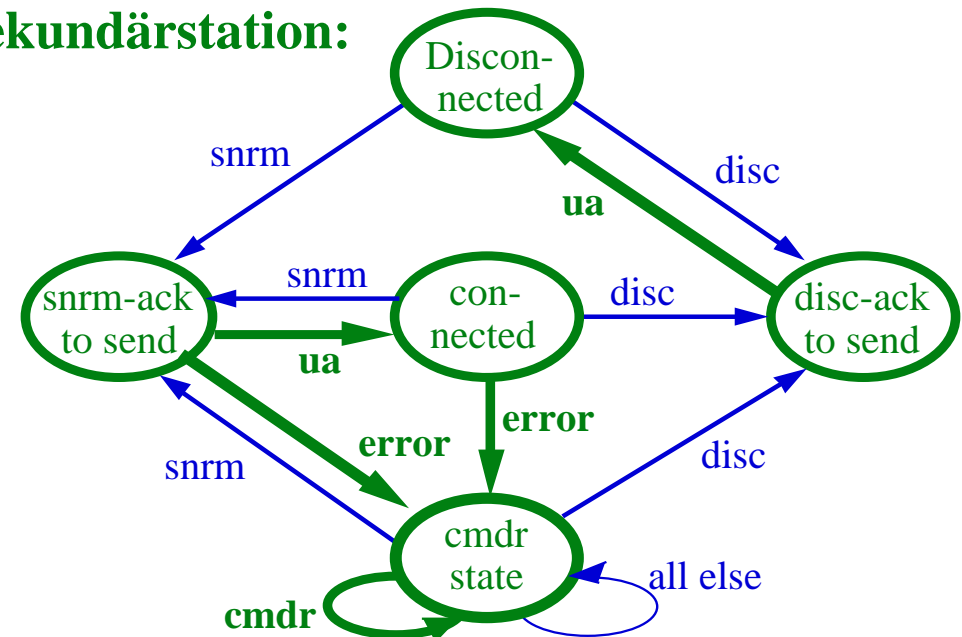
- Bei Fullduplex-Betrieb kann das nächste Terminal schon angesprochen werden, bevor das aktuelle Terminal fertig geantwortet hat.

7.2.8 Verbindungsaufbau & -abbau

Primärstation:

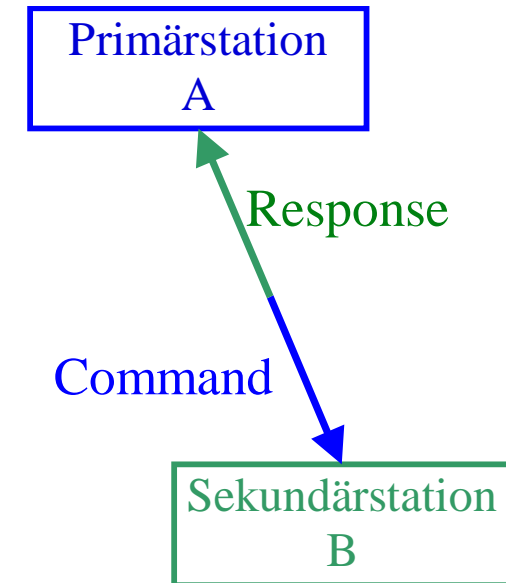


Sekundärstation:

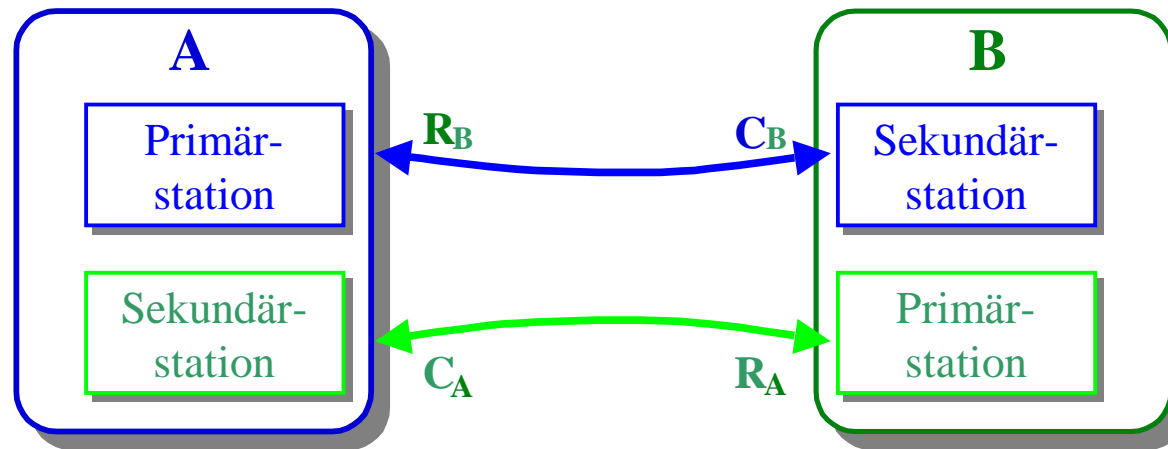


7.2.9 Typischer Protokollablauf für NRM

B, RR(0)-P	=>	B wecken
	<=	B, RIM-F
B, SIM-P	=>	Brauche Initialisierung
	<=	B, UA-F
		Kann Initialisieren
		OK Set
		Download ...
B, SNRM-P	=>	NRM setzen, Reset N _{send}
	<=	B, UA-F
		OK NRM
B, RR(0)-P	=>	Bitte übertragen
B, I(0, 0)	<=>	B, i(0, 0)
		Übertragung zu B
		Übertragung zu A
B, I(1, 1)	<=>	B, i(1, 1)
		Duplexübertragung A-B
		Duplexübertragung A-B
B, I(2, 2)	<=>	B, i(2, 2)
B, I(3, 3)	=>	Bestätigung von i(1) ...
B, RR(3)	=>	Langer Frame zu B
	<=	B, RR(4)
	<=	B, i(3, 4)-F
		Bestätigung von i(2)
		Bestätigung von I(3)
		Übertragung zu A
		*** CRC Error: Rahmen verloren
B, RR(3)-P	=>	timeout, B pollen
	<=	B, i(3, 4)-F
		Wiederholung i(3)
B, DISC-P	=>	Abbau
	<=	B, UA-F
		Abbau OK



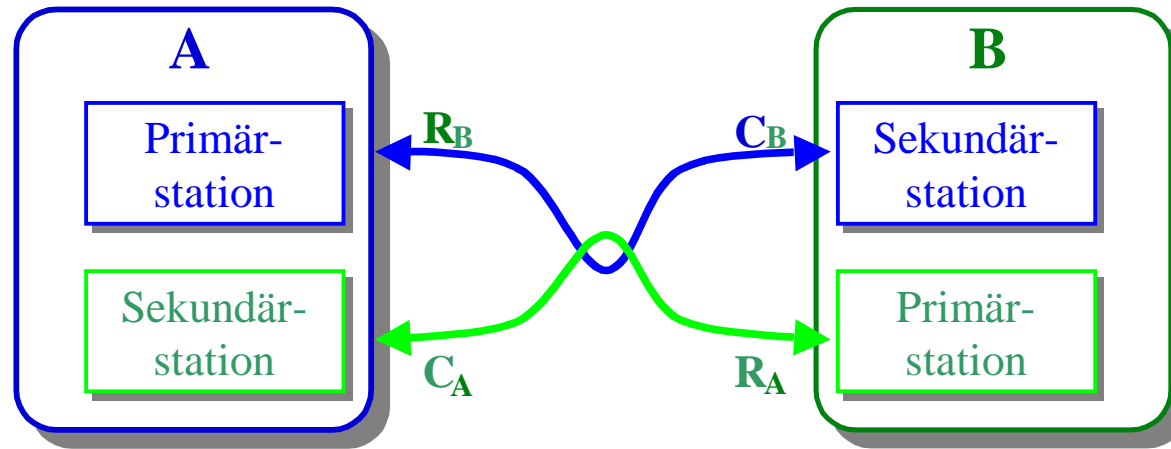
7.2.10 Asynchronous Response Mode



- Symmetrische Rollenverteilung zwischen den Partnern.
- 2 Übertragungsprotokolle nebeneinander.
- Initialisierung mit SARM für jede Richtung:

B, SARM-P	=>	ARM, Reset N_s
	<=	B, UA-F
B, RR(0)	=>	1. Richtung offen
	<=	A, SARM-P
A, UA-F	=>	OK ARM
	<=	A, SARM-P
	=>	OK ARM
	<=	B, UA-F
	...	Duplex DÜ

7.2.11 Async. Balanced Response Mode



- Huckepack Bestätigungen möglich.
- Initialisierung mit SABM für eine Richtung genügt:

B,SABM-P	\Rightarrow	ABM, Reset Ns
	\Leftarrow	B,UA-F
	...	Duplex DÜ

7.2.12 HDLC Adressierung

- Original HDLC verwendet kein Adressenpaar (Destination,Source), sondern immer die Adresse der Sekundärstation.
- Bei Normal Response Mode:
 - Primärstation sendet Commands,
 - Sekundärstationen senden Replies.
- Primärstation:
 - sendet Commands und empfängt Replies mit Partneradresse,
 - Hat eigentlich keine Adresse,
- Sekundärstation empfängt Commands und sendet Replies über die eigene Adresse.
- Bei symmetrischen Varianten:
 - Commands und Replies anhand der Adresse unterscheiden:
 - Reply immer, wenn Senderadresse,
 - Command, wenn Empfängeradresse.

7.2.13 Fehlerbehandlung

- Rahmen mit falscher Prüfsumme oder mit Abbruchsequenz verwerfen.
- Reject erfolgt typisch nach Empfang des Rahmens $N+1$ anstatt N .
- Überlasteter Empfänger verwirft Rahmen.
- Erneute Übertragung nach:
 - Selective Reject SREJ (optional),
 - Reject (REJ),
 - Time-Out.

7.2.14 Weitere Befehle & Antworten

- Erneute Initialisierung wird durch die Primärstation eingeleitet nach:
 - Command Reject (CMDR),
 - Frame Reject (FRMR),
 - internem Fehler.
- Optional:
 - DM - Verbindungswunsch ablehnen.
 - RSET - Sequenznummern auf Null.
 - XID - Knotenident. austauschen.
 - RIM - Request Initialization Mode.
 - SIM - Set Initialization Mode.
- Erweitertes Kontrollfeld verlangen mit SNRME, SARME, SABME:
 - 7 Bit lange Sequenznummern ..
- Feste Vereinbarung möglich für:
 - 16 Bit Adressfeld (evtl. Sender und Empfängeradresse),
 - erweitertes Kontrollfeld,
 - Initialisierungsbedürfnisse,
 - maximale Paketlänge,
 - Fenstergrösse.

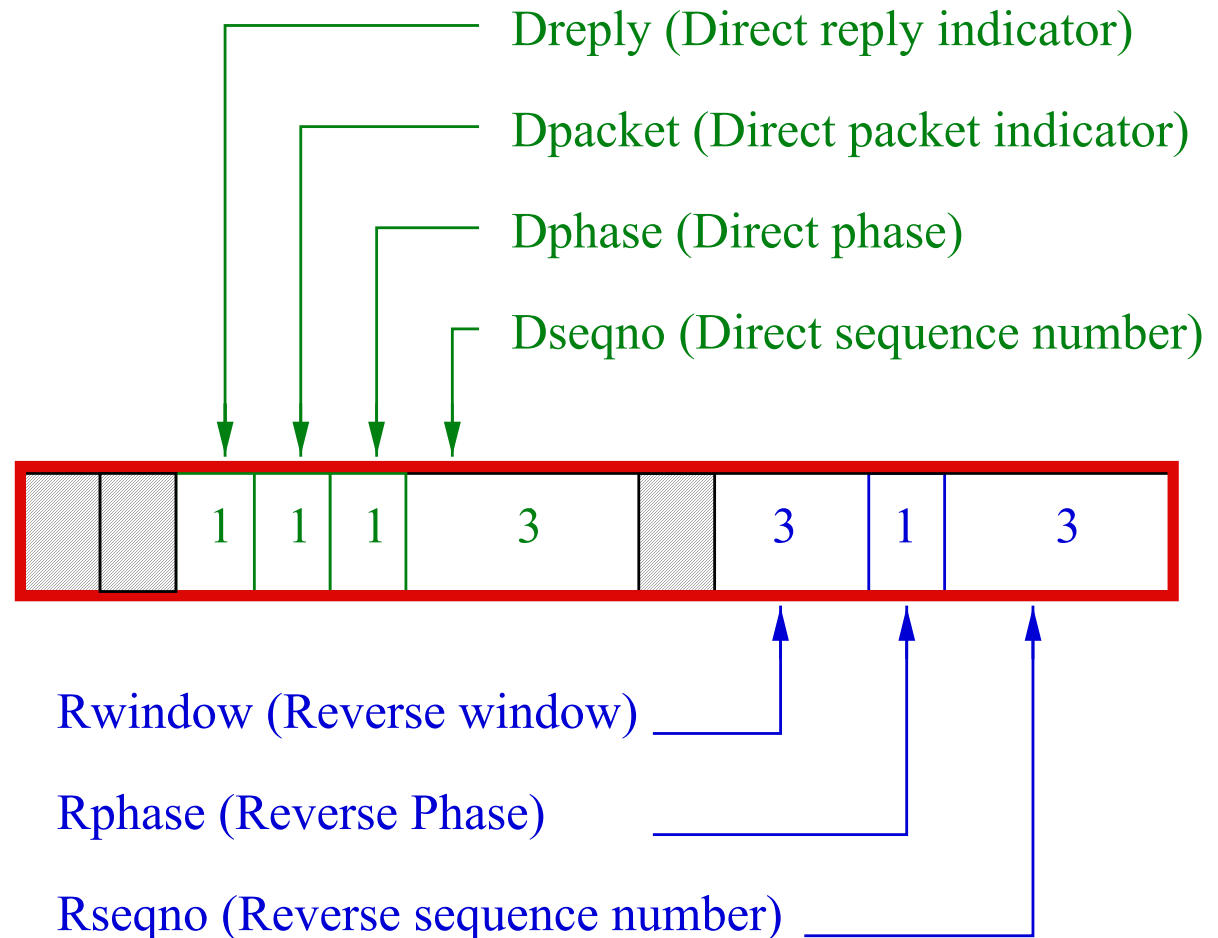
7.3. Link-Protokoll nach Fletcher

- Nach Fletcher, J. G. :
 - *"Serial Link Protocol Design: A Critique of the X.25 Standard, Level 2"*;
 - ACM Comp. Comm. Review, ACM Symp. Montreal, June 1984, Vol. 14 No. 2.
- Verbesserte Funktionalität:
 - dynamische Kontrolle der Fensteröffnung,
 - Bessere Erholung von Fehlersituationen,
 - Timeout Intervalle unkritisch.
- Vereinfachung:
 - leichter zu implementieren, zu validieren, zu verstehen,
 - symmetrisch, keine Primär- & Sekundärstationen,
 - Kontrollfeld enthält Zustand und nicht Befehl.
- DL-PDU:
 - Adressfeld separat für Service Access Point,
 - Kontrollfeld,
 - Daten,
 - FCS ..

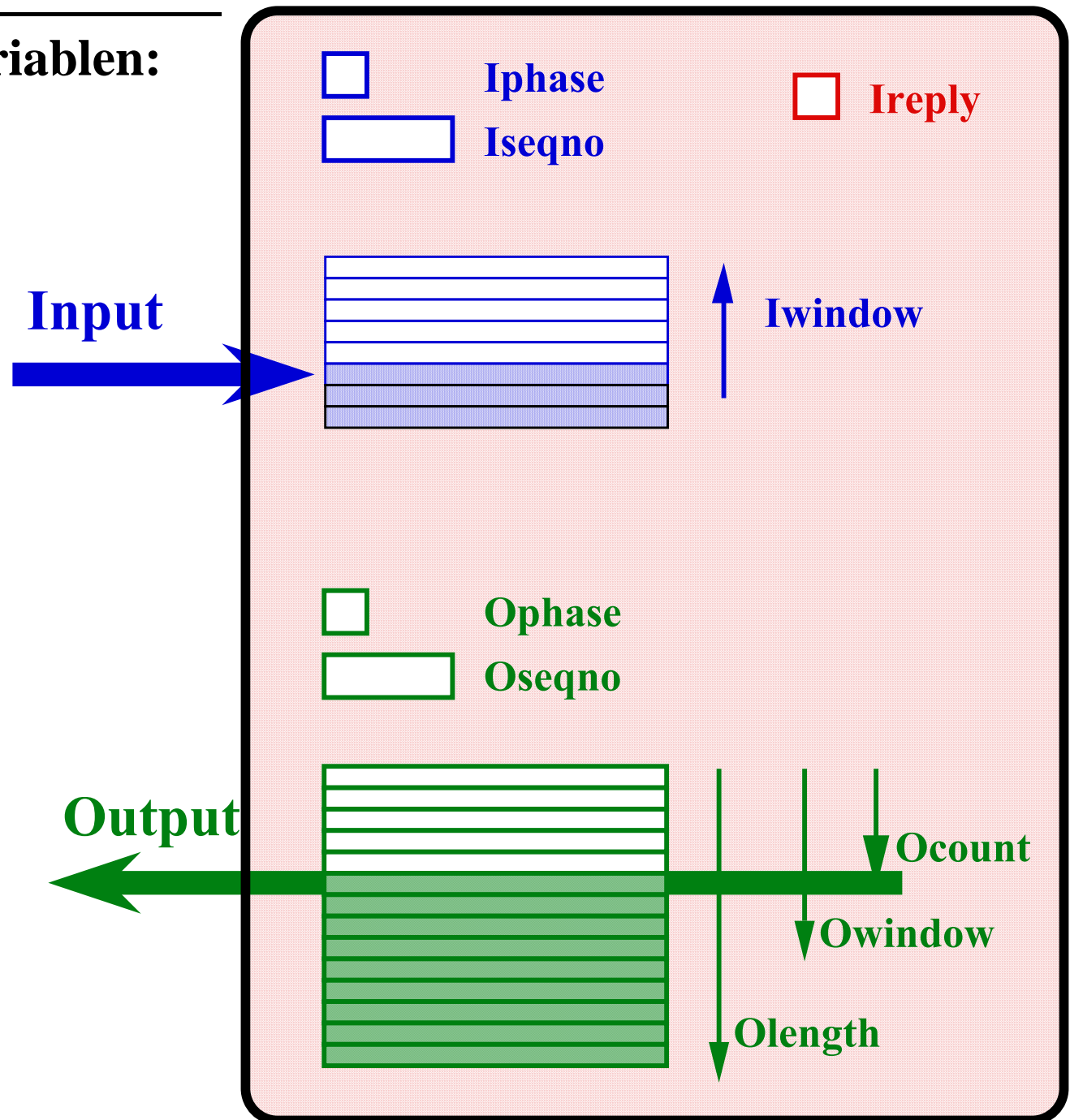


7.3.1 ALP Kontrollfeld

- Phase ändert für jede Wiederholungssequenz.
- Direct Reply Indicator verlangt Antwort.
- Keine Befehlscodes im Meldungskopf.
- Full-duplex Kanal.



7.3.2 ALP Statusvariablen:



7.3.3 Variablen des Empfangsteils:

Iseqno: nächste erwartete Paketnummer.

Iwindow: Öffnung des Empfangsfensters.

Ireply: Gegenstation soll antworten.

Iphase: laufende Übertragungsphase für die Empfangsrichtung. Nach jedem Fehler wird die Phase invertiert ($I_{\text{phase}} = \text{not } I_{\text{phase}}$).

7.3.4 Variablen des Sendeteils:

Oseqno: Nummer des nächsten Ausgabepakets.

Owindow: Öffnung des Sendefensters.

Ocount: Anzahl abgeschickter jedoch noch unbestätigter Pakete.

Olength: Anzahl Pakete, die noch nicht sicher übertragen wurden.

Ophase: laufende Übertragungsphase für die Senderichtung. Diese Phase wird durch die Gegenstation invertiert (Dphase-Bit im Paket).

7.3.5 ALP Protokollbeschreibung

- Pascal-artig:

```
procedure FrameOut;  
begin (* Perform FrameOut whenever we have  
1) a new output packet ready    2) received an input frame  
3) changed Iwindow              4) periodically                5) startup *)  
while Ocount < min( Owindow, Olength )  
    or Ireply do begin (* send Frame *)  
    f      := newframe( );  
    Ireply := false ;  
    Dphase(f) := Ophase;  
    Dseqno(f) := (Oseqno + Ocount) mod M;  
    if Ocount < min( Owindow, Olength ) then  
        (* include a packet in the frame *)  
        data( f )      := Oqueue[ Ocount];  
        Ocount        := Ocount + 1  
        Dpacket( f )   := true ;  
    end (* Ocount *);  
    if Olength > 0 then Dreply( f ) := true end ;  
        (* Not all output packets  
        have as yet been accepted *)  
        Rphase( f )   := Iphase;  
        Rseqno( f )   := Iseqno;  
        Rwindow( f ) := Iwindow;  
        send( f )  
    end (* While *)  
end (* FrameOut *).
```

```

procedure NormalFrameIn;
begin  f := receive( );      (* do this after receipt of each non-erroneous frame *)
      if Dseqno(f) = Iseqno and Dphase(f) = Iphase then begin
          (* input packet in sequence, no retransmit *)
          if Dpacket(f) and Iwindow > 0 then begin (*Dpacket="data included" & win open*)
              Iseqno := (Iseqno + 1) mod M; forward( data( f ) )
          end
          else (* no data included or window closed*)
      else  Iphase := 1 - Dphase(f);  (* Input packet was lost: indicate retransmission *)
      if ((Rseqno(f) - Oseqno) mod M) • Ocount then begin (* regular reverse seq. # recvd *)
          while Rseqno(f) • Oseqno do begin (* discard acknowledged output packets *)
              pop( Oqueue);
              Oseqno := (Oseqno + 1) mod M;
              Ocount := Ocount - 1; Olength := Olength - 1
          end (* while • *);
          Owindow := Rwindow(f)
      end (* if Rseqno within range *);
      if Rphase(f) • Ophase then begin (* Prepare to retransmit rejected output packets *)
          Ophase := Rphase(f);  Oseqno := Rseqno(f);  Ocount := 0
      end (* if Rphase *);
      Ireply := Dreply(f) or ( Olength > 0 );
      (* State was unsatisfactory or unackn. packets *)
      deallocate(f)
end (* NormalFrameIn *);

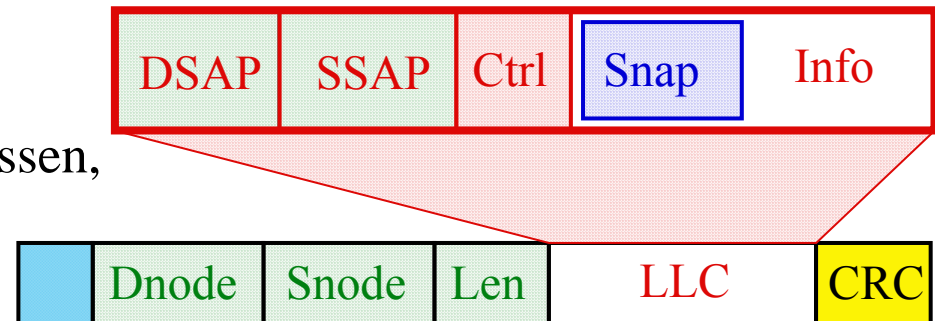
```

7.4. LLC Protokoll

- Linkprotokoll u.a. auf Ethernet-LANs (802.3).

- Eingebettet in Ethernet-Rahmen:

- Präambel (8 Bytes),
- Dnode/Snode: 6 Byte Netzadapter-Adressen,
- Längelfeld (• 1500 Bytes),
- LLC - PDU,
- CRC-32:



- LLC-PDU:

- DSAP / SSAP als Dienstzugangspunkte,
- SAPs wählen einen Protokollhandler,
- Ctrl-Kontrollfeld ähnlich HDLC,
- SNAP = optionales Subnetz-Access-Protokoll.

- Verschiedene LLC-Typen:

- 1 - Unnummerierte Nachrichten (Ctrl=UI),
- 2 - Verbindungsaufbau wie balanced HDLC,
- 3 - AC-PDUs mit 1 Bit Sequenznummer.

- Typ 2 verwendet 7 Bit Sequenznummern.

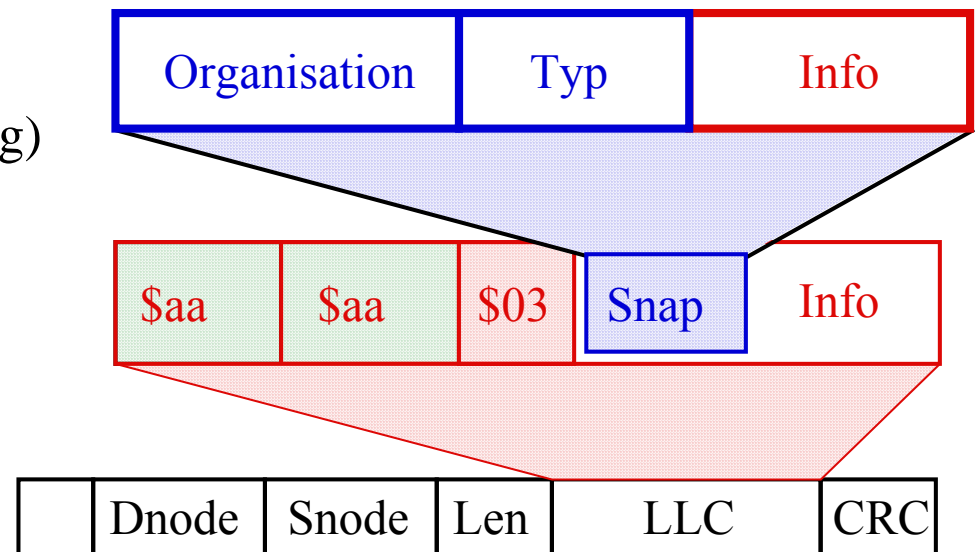
- Typ 2 Rahmentypen ähnlich HDLC:
 - nur symmetrisches Szenarium,
 - I, RR, RNR, REJ, UI, UA, DISC,
 - SABME, XID, TEST, DM, FRMR.

7.4.1 SNAP Protokollvarianten

- Protokollbezeichner wählen Protokollhandler der nächsten Schicht aus.
- Nur 6 Bit des DSAP sind nutzbar, deshalb Erweiterungsmöglichkeit durch SNAP (802.3 Encapsulation, "Subnetwork Access Protocol"):

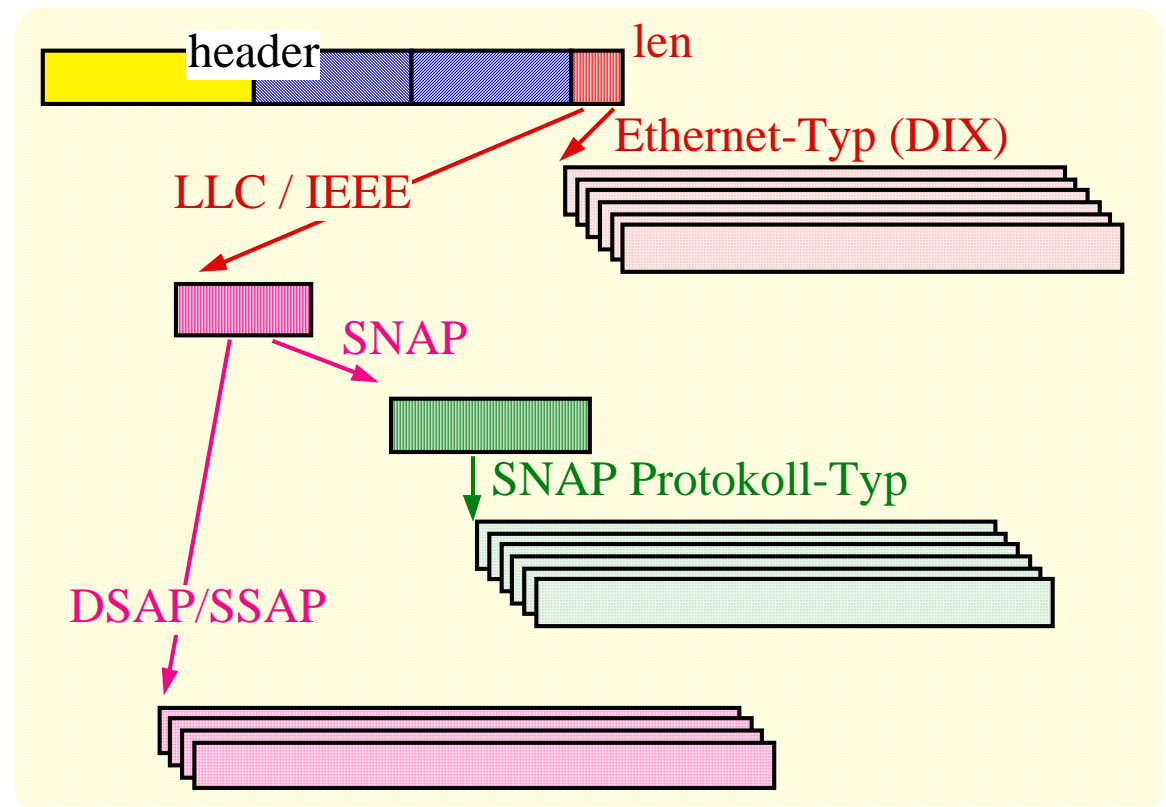
- SNAP Protokollbezeichner z.B.:

00 00 00 , 08 00	DoD IP
00 00 00 , 08 06	ARP (Adressermittlung)
00 00 00 , 80 f3	Apple ARP
08 00 07 , 80 9b	AppleTalk



7.4.2 Ethernet-Rahmenvarianten

- Automatische Unterscheidung der Varianten anhand des Längenfeldes:
- Inkompatible Varianten selten (Raw ...)
- Ethernets nach IEEE 803.3 Empfehlung:
 - echtes Längenfeld mit Länge < 1536 Bytes,
 - immer LLC Protokoll eingebettet (802.2),
 - SNAP Erweiterung fakultativ.
- Original Ethernet II (DIX):
 - EtherType Spec.(• 1536),
 - damit entfällt das Längenfeld,
 - mit oder ohne LLC möglich,
 - verbreitet & kompatibel,
 - historisch.



- **EtherType Protokoll-Identifizierer (Hex):**

0-05ff	IEEE802.3 Length Field
0800	Dod Internet Protocol (IP)
0806	Address Resolution Protocol (ARP)
0bad	Banyan Systems
6003	DECNET Phase IV, DNA Routing
6004	DEC Local Area Transport (LAT)
8019	Apollo DOMAIN
8035	Reverse Addr. Res. Protok. (RARP)
8038	DEC LanBridge Management
803d	DEC Ethernet CSMA/CD Encrypt.
809B	EtherTalk (AppleTalk over Ethernet)
80F3	AppleTalk Addr. Res. Protok. (AARP)
80F4/5	Kinetics
8137/8	Novell Inc.

- **LLC-DSAP Identifizierer (Hex):**

06	IP
e0	IPX
f0,f1	Netbeui

7.5. PPP Protokoll

- z.B. als Zugangsprotokoll über Modem (=> RFC 1548 als Referenz).
- Verschiedene Netzwerkprotokolle auf eine Leitung multiplexen.
- HDLC mit CRC-16 und wahlweise Bit- oder Bytestuffing.
- Für synchrone & asynchrone Modems.
- 16 Bit Protokoll-Bezeichner.
- Authorisierungsprotokoll.
- Header-Komprimierung.

