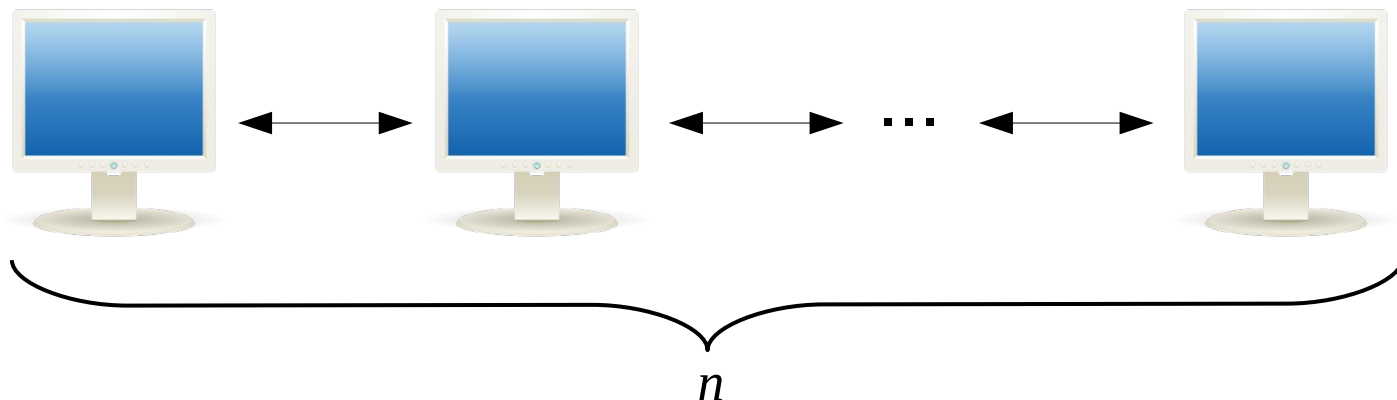


Wiederholung / Hausaufgabe: Liniennetz

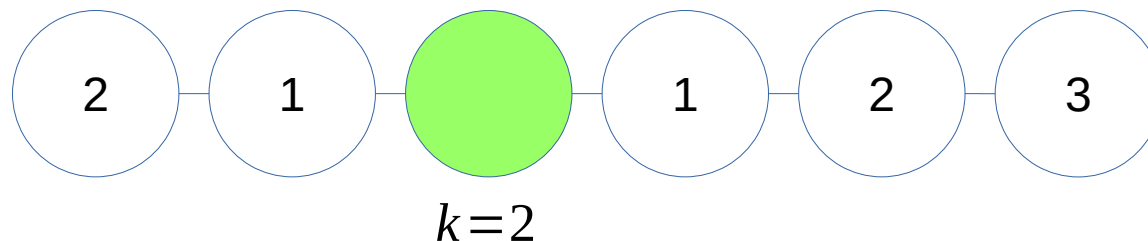
- Gegeben ist ein Liniennetz mit n Knoten:



- Jeder Knoten sendet pro Zeiteinheit zufällig ein Paket an einen der anderen $n-1$ Knoten.
- Gesucht ist der mittlere Hop-Count der Pakete pro Zeiteinheit.

Wiederholung / Hausaufgabe: Liniennetz

- Generelles Vorgehen:
 - Mittleren Hop-Count pro Knoten bestimmen (heterogene Topologie, jeweils $n-1$ Knoten einbeziehen)
 - Über alle n Knoten mitteln



Wiederholung / Hausaufgabe: Liniennetz

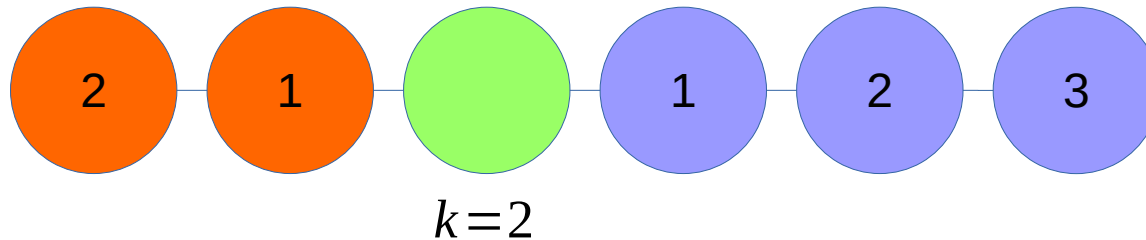
- Summe der ersten n natürlichen Zahlen (Gauß'sche Summenformel):

$$\sum_{k=1}^n k = \frac{n(n+1)}{2}$$

- Summe der ersten n Quadratzahlen:

$$\sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6}$$

Wiederholung / Hausaufgabe: Liniennetz



- Linke Summe: $\sum_{l=1}^k l = \frac{k \cdot (k+1)}{2}$

- Rechte Summe: $\sum_{r=1}^{n-k-1} r = \frac{(n-k) \cdot (n-k-1)}{2}$

- Im Mittel für Knoten k : $\frac{k \cdot (k+1) + (n-k) \cdot (n-k-1)}{2 \cdot (n-1)}$

$n > 1$

Wiederholung / Hausaufgabe: Liniennetz

- Im Mittel für alle Knoten:

$$\underbrace{\frac{1}{n} \cdot \frac{1}{2 \cdot (n-1)}}_c \cdot \sum_{k=0}^{n-1} (k \cdot (k+1) + (n-k) \cdot (n-k-1))$$

- Trennen der Summe:

$$\begin{aligned} & 2c \sum_{k=0}^{n-1} (k^2 + (1-n) \cdot k) + c \sum_{k=0}^{n-1} (n^2 - n) \\ \Leftrightarrow & 2c \sum_{k=0}^{n-1} k^2 + 2c(1-n) \sum_{k=0}^{n-1} k + \frac{n}{2} \end{aligned}$$

Wiederholung / Hausaufgabe: Liniennetz

- Damit folgt:

$$\Leftrightarrow \frac{2n-1}{6} + \frac{1-n}{2} + \frac{n}{2} = \frac{n+1}{3}$$

- Interpretation?

Sequenznummern

- *Send-And-Wait*-Szenario: ^[1]
 - Synonyme: *Stop-And-Wait*, *Idle-RQ*
 - Beteiligt: Sender **S**, Empfänger **R**
 - S sendet Rahmen mit Nutzlast (*I-Frames*) an R.
 - R bestätigt jeden I-Frame mit einem *ACK-Frame*.
 - S wartet auf den ACK-Frame und sendet erst nach Erhalt den nächsten I-Frame.

[1]: Fred Halsall: Computer Networking and the Internet

Sequenznummern

- **Aufgabe 3.1)**

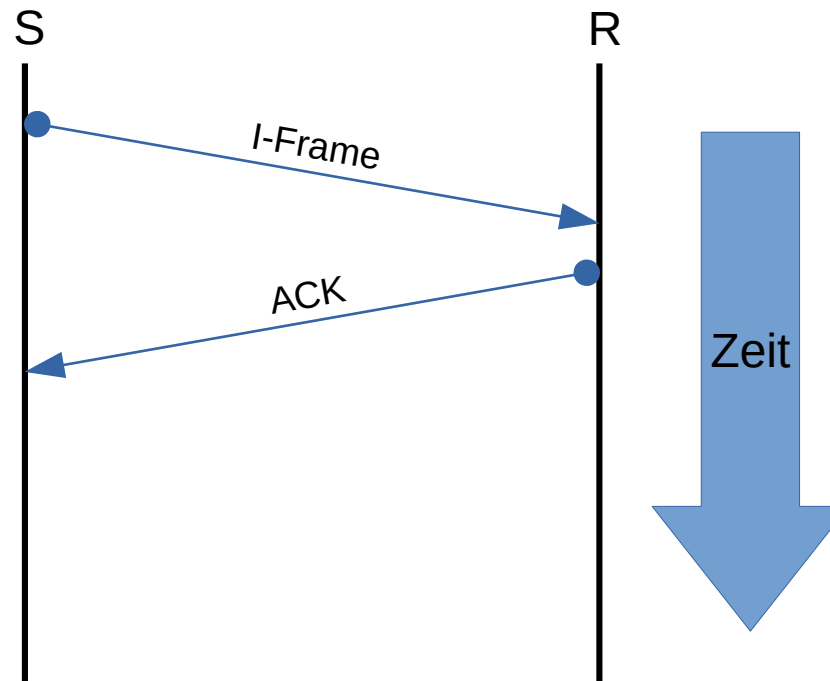
Skizzieren Sie die folgenden Szenarien und schlagen Sie Konzepte zum Umgang mit Fehlersituationen vor:

- Erfolgreiche Übertragung und Bestätigung eines I-Frames
- Verlust eines I-Frames
- Umkippen von Bits in einem I-Frame
- Verlust eines ACKs
- Duplizierung eines I-Frames oder ACKs
- Umkippen von Bits in einem ACK

[1]: Fred Halsall: Computer Networking and the Internet

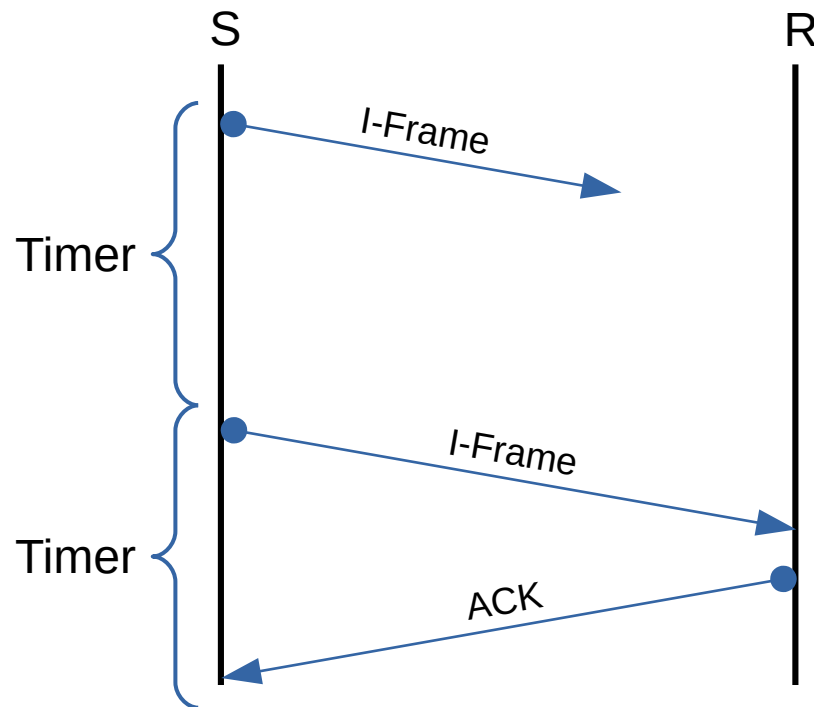
Sequenznummern

- Erfolgreiche Übertragung und Bestätigung eines I-Frames:



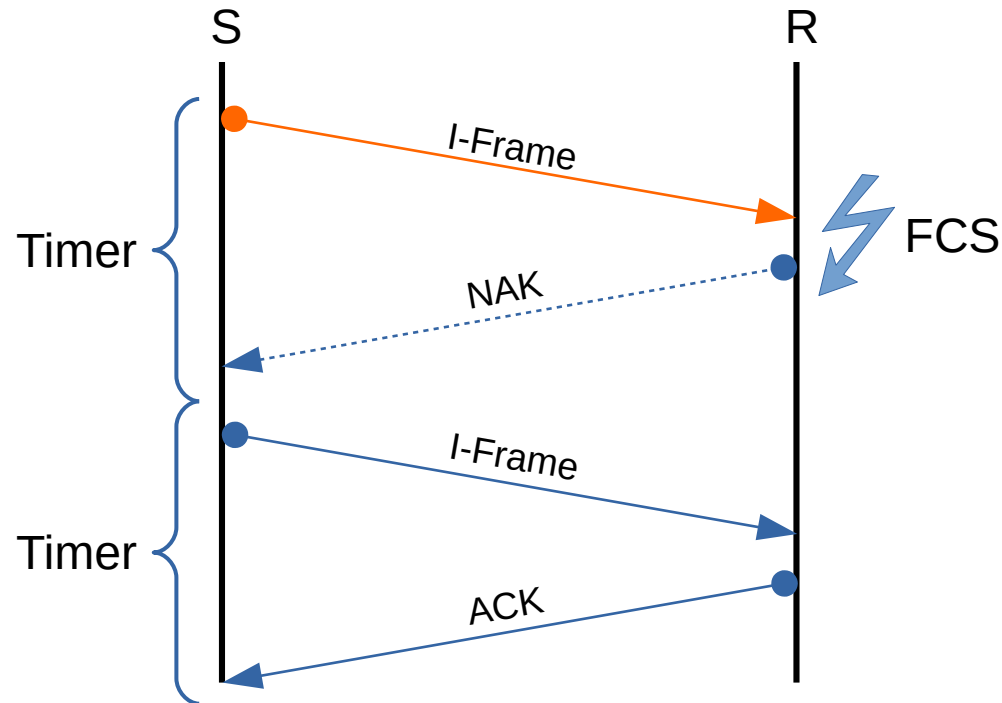
Sequenznummern

- Verlust eines I-Frames:



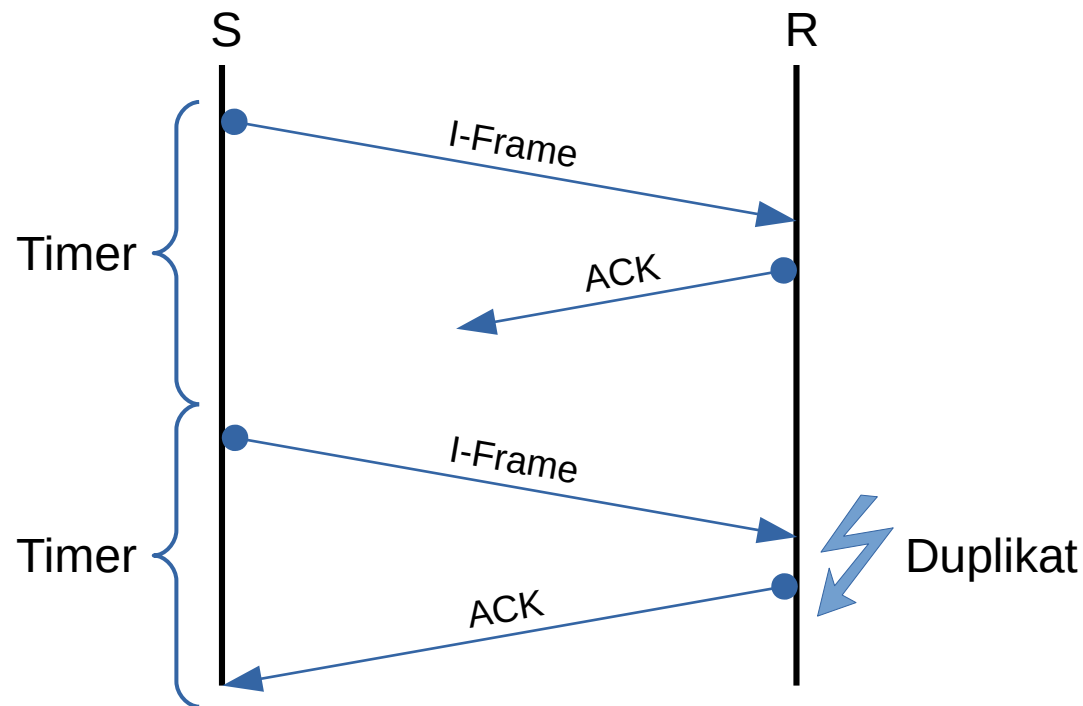
Sequenznummern

- Umkippen von Bits in einem I-Frame:



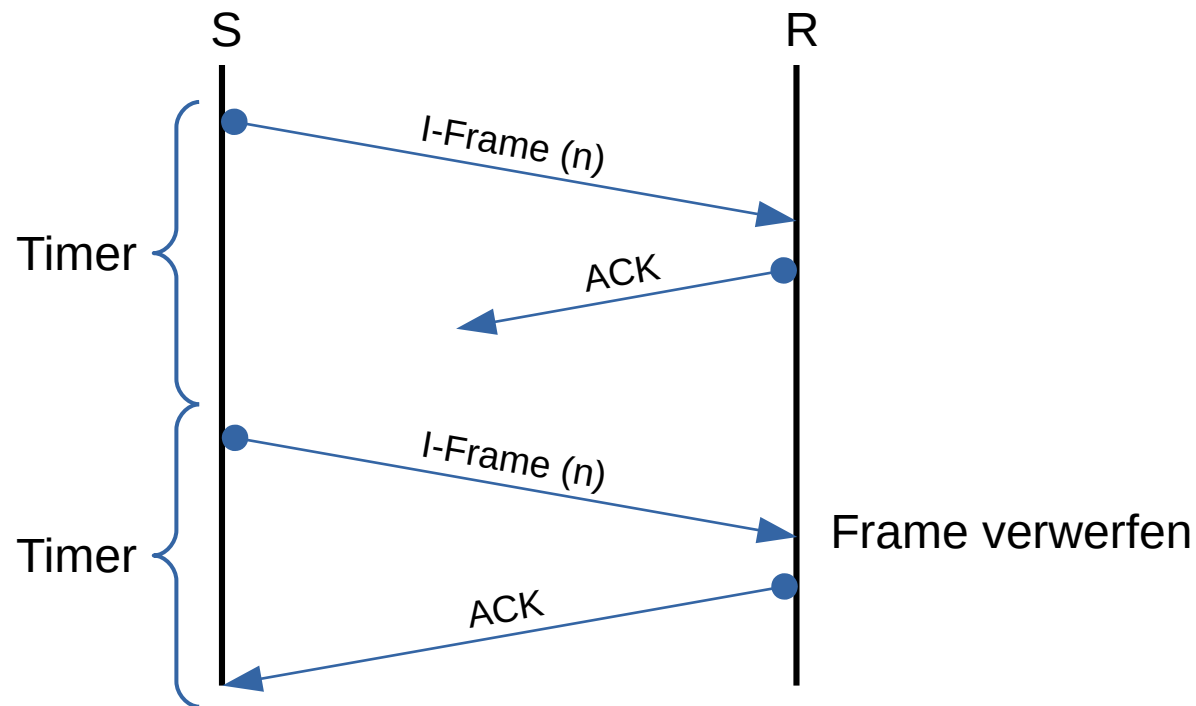
Sequenznummern

- Verlust eines ACKs:



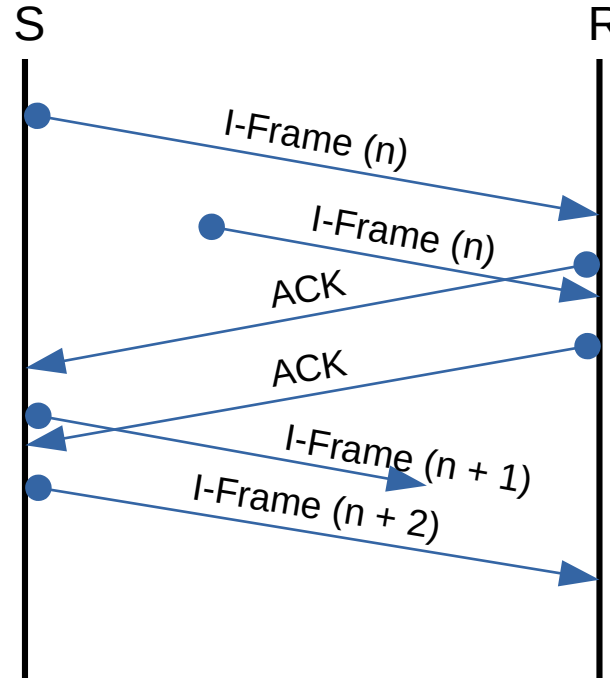
Sequenznummern

- Verlust eines ACKs:



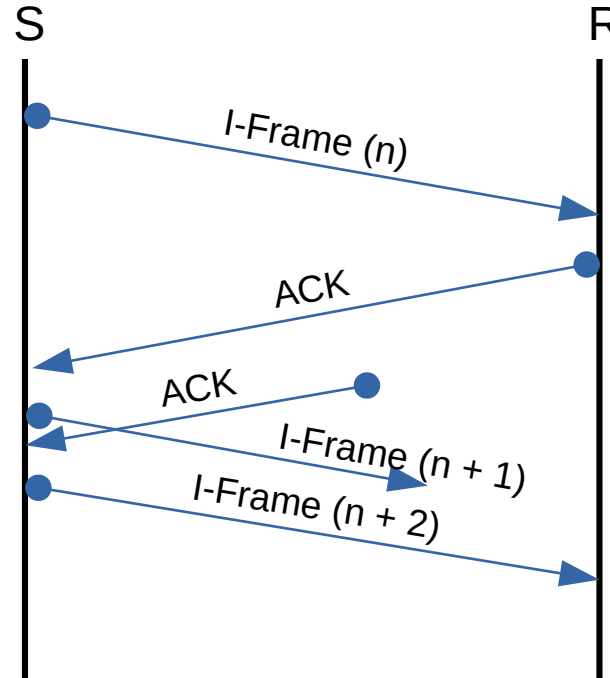
Sequenznummern

- Duplizierung eines I-Frames:



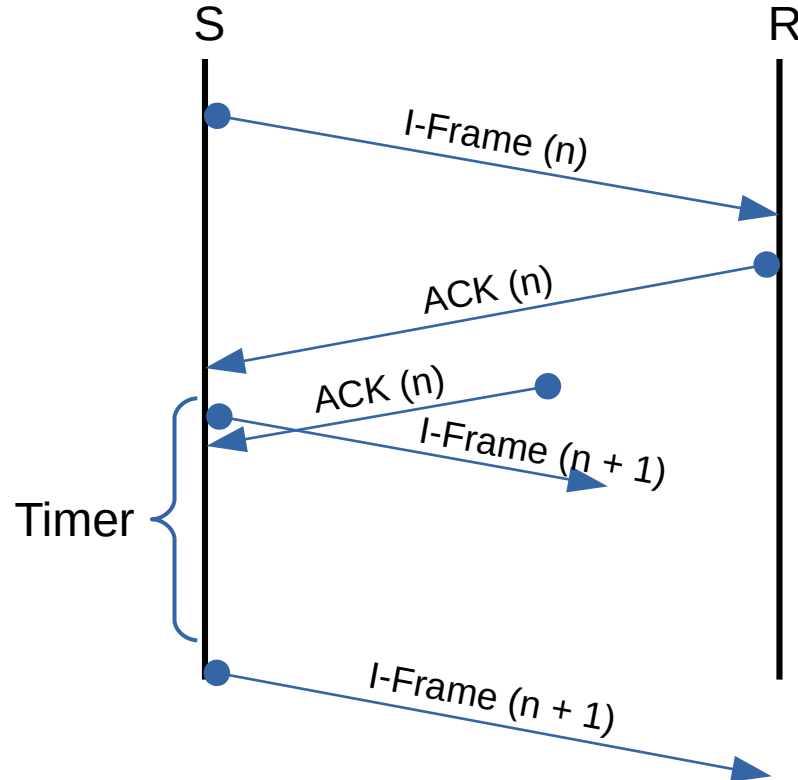
Sequenznummern

- Duplizierung eines ACKs:



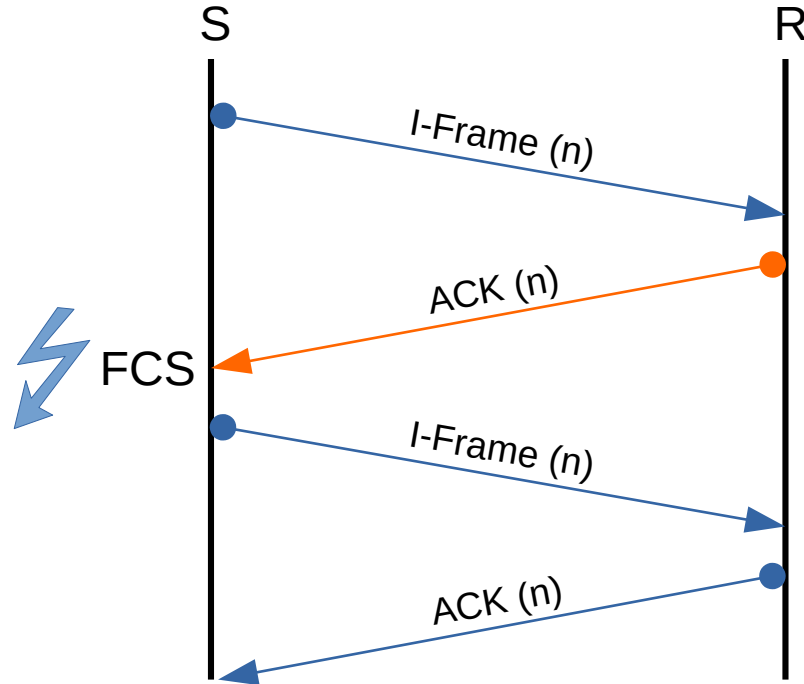
Sequenznummern

- Duplizierung eines ACKs:



Sequenznummern

- Umkippen von Bits in einem ACK:

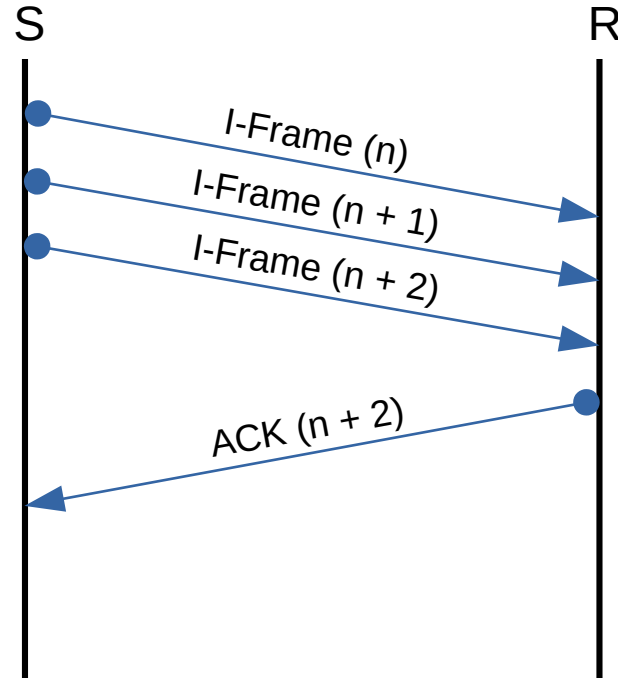


Sliding Window

- Ab jetzt: S wartet nicht mehr auf jedes ACK.
- **Aufgabe 3.2)**
 - *Erläutern Sie die zwei grundlegenden Strategien zur erneuten Übertragung verlorener I-Frames:*
 - *Go-Back-N*
 - *Selective Repeat*

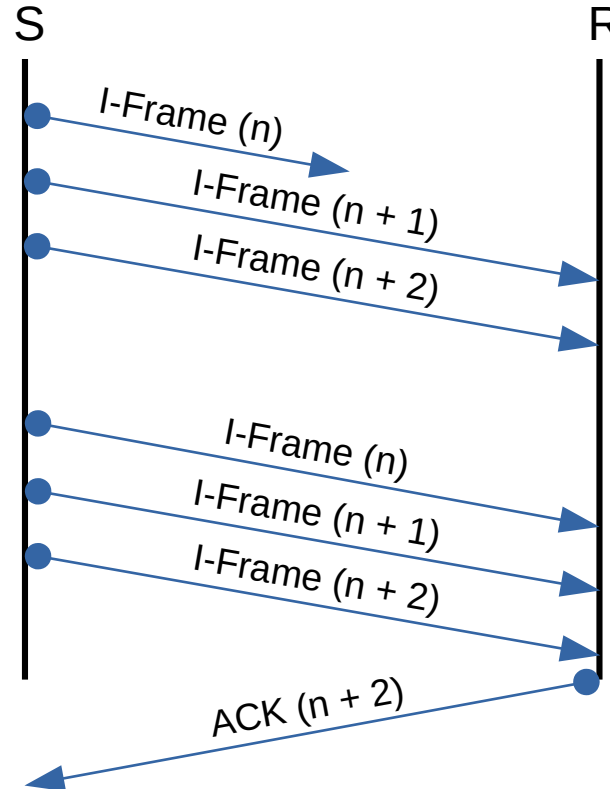
Sliding Window

- Go-Back-N:



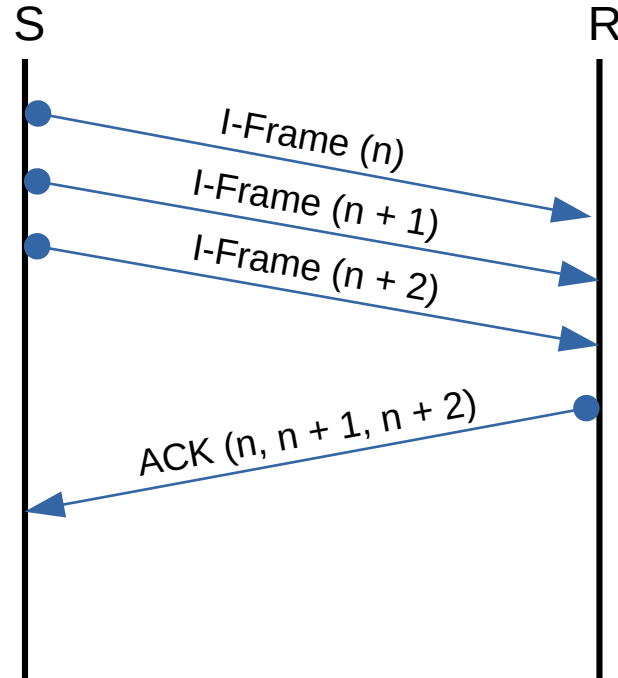
Sliding Window

- Go-Back-N:



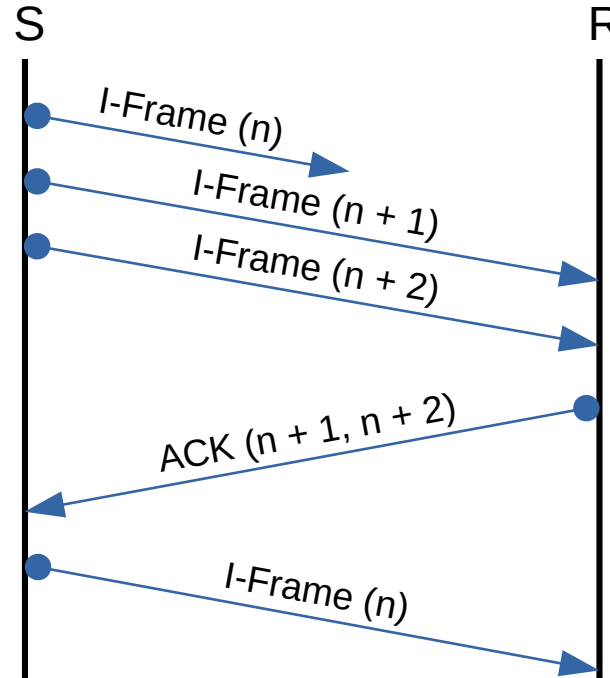
Sliding Window

- Selective Repeat:



Sliding Window

- Selective Repeat:

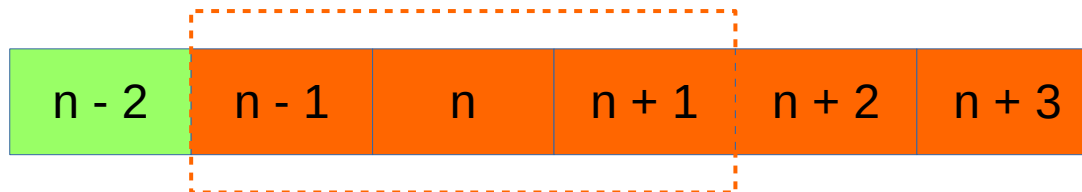


Sliding Window

- Definieren Sie den Begriff der Fenstergröße (engl. Window size).

„Die Fenstergröße bezeichnet die Menge an Daten, die der Sender unbestätigt abschicken darf, bevor er ein ACK erhält.“

- „Datenmenge“: hier die Anzahl der Frames (aber z. B. nicht bei TCP).
- „Sliding Window“:

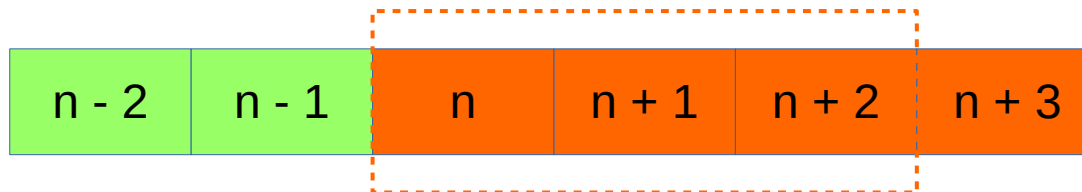


Sliding Window

- Definieren Sie den Begriff der Fenstergröße (engl. Window size).

„Die Fenstergröße bezeichnet die Menge an Daten, die der Sender unbestätigt abschicken darf, bevor er ein ACK erhält.“

- „Datenmenge“: hier die Anzahl der Frames (aber z. B. nicht bei TCP).
- „Sliding Window“:

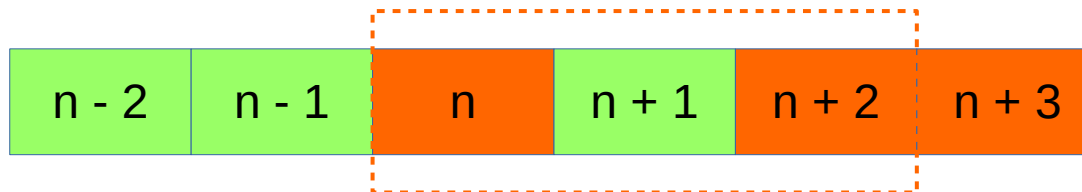


Sliding Window

- Definieren Sie den Begriff der Fenstergröße (engl. Window size).

„Die Fenstergröße bezeichnet die Menge an Daten, die der Sender unbestätigt abschicken darf, bevor er ein ACK erhält.“

- „Datenmenge“: hier die Anzahl der Frames (aber z. B. nicht bei TCP).
- „Sliding Window“:

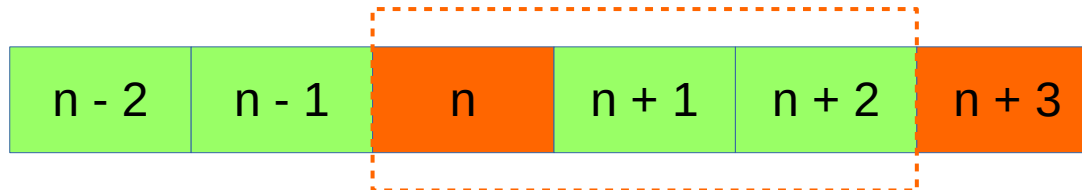


Sliding Window

- Definieren Sie den Begriff der Fenstergröße (engl. Window size).

„Die Fenstergröße bezeichnet die Menge an Daten, die der Sender unbestätigt abschicken darf, bevor er ein ACK erhält.“

- „Datenmenge“: hier die Anzahl der Frames (aber z. B. nicht bei TCP).
- „Sliding Window“:

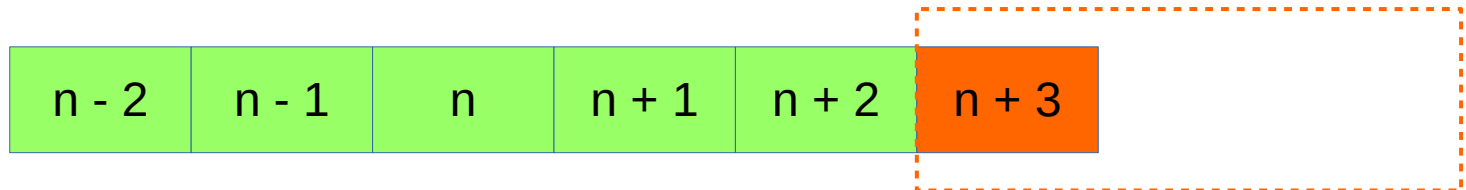


Sliding Window

- Definieren Sie den Begriff der Fenstergröße (engl. Window size).

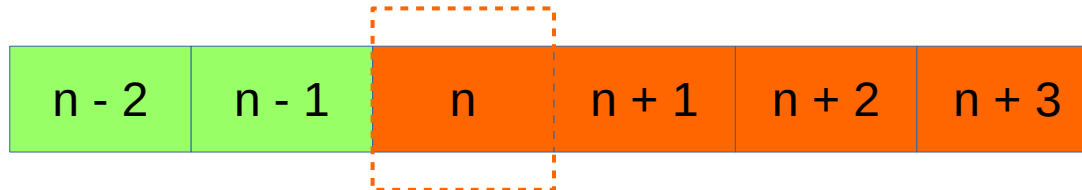
„Die Fenstergröße bezeichnet die Menge an Daten, die der Sender unbestätigt abschicken darf, bevor er ein ACK erhält.“

- „Datenmenge“: hier die Anzahl der Frames (aber z. B. nicht bei TCP).
- „Sliding Window“:

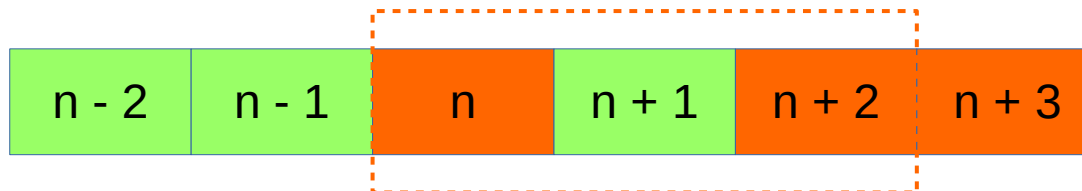


Sliding Window

- Und der Empfänger?
 - Für Go-Back-N: Ein Zeiger reicht aus

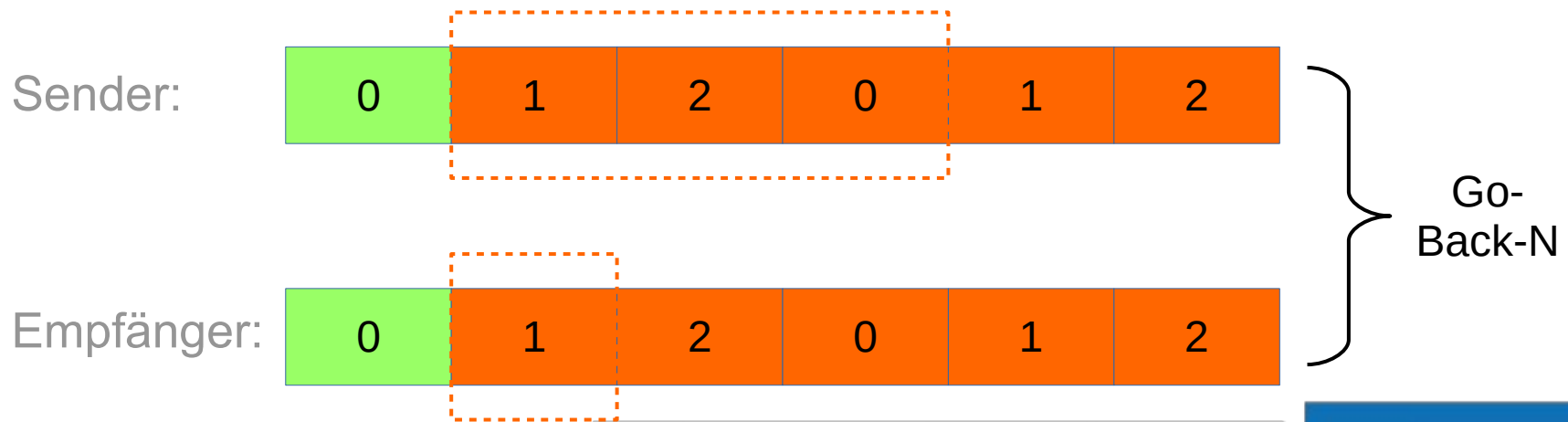


- Für Selective-Repeat: Ein Fenster für alle gültigen Frames



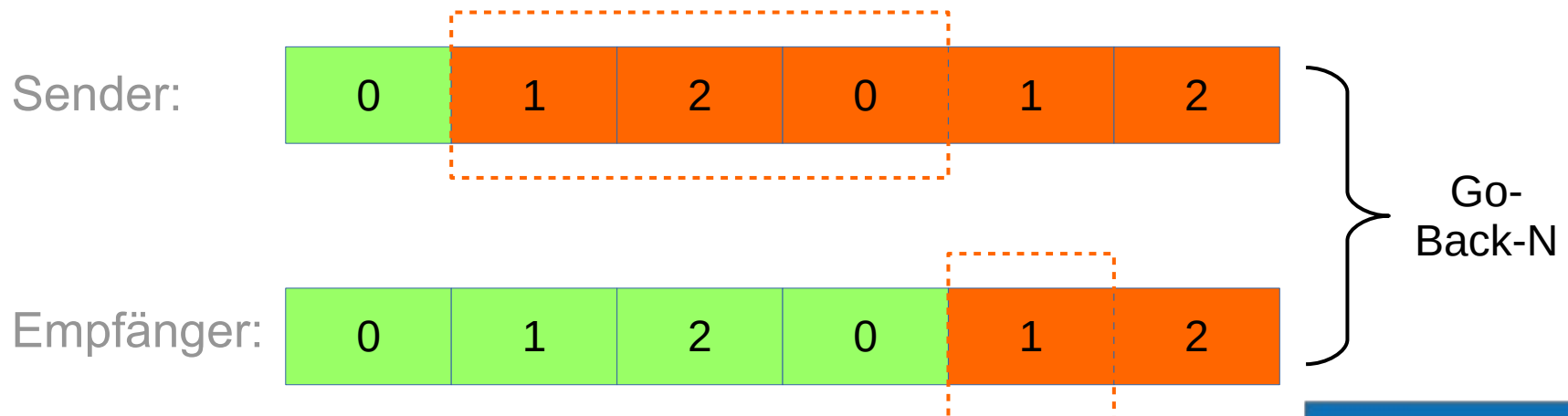
Sliding Window

- Protokolldesign: Bitfeld für die Sequenznummer
- Beschränkt, z. B. n Bits $\rightarrow 2^n$ unterschiedliche Sequenznummern
- *Wie hängt die Wahl der Fenstergröße mit dem Wertebereich für die Sequenznummer zusammen? Berücksichtigen Sie, ob Go-Back-N oder Selective Repeat zum Einsatz kommt.*
- Tipp: Worst-Case-Szenario konstruieren!



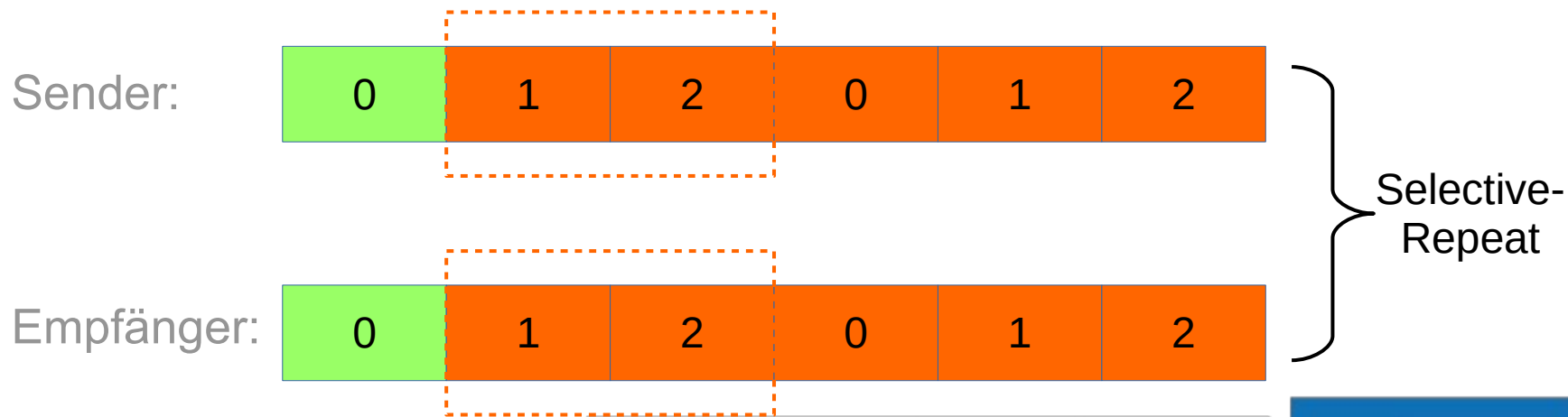
Sliding Window

- Protokolldesign: Bitfeld für die Sequenznummer
- Beschränkt, z. B. n Bits $\rightarrow 2^n$ unterschiedliche Sequenznummern
- *Wie hängt die Wahl der Fenstergröße mit dem Wertebereich für die Sequenznummer zusammen? Berücksichtigen Sie, ob Go-Back-N oder Selective Repeat zum Einsatz kommt.*
- Tipp: Worst-Case-Szenario konstruieren!



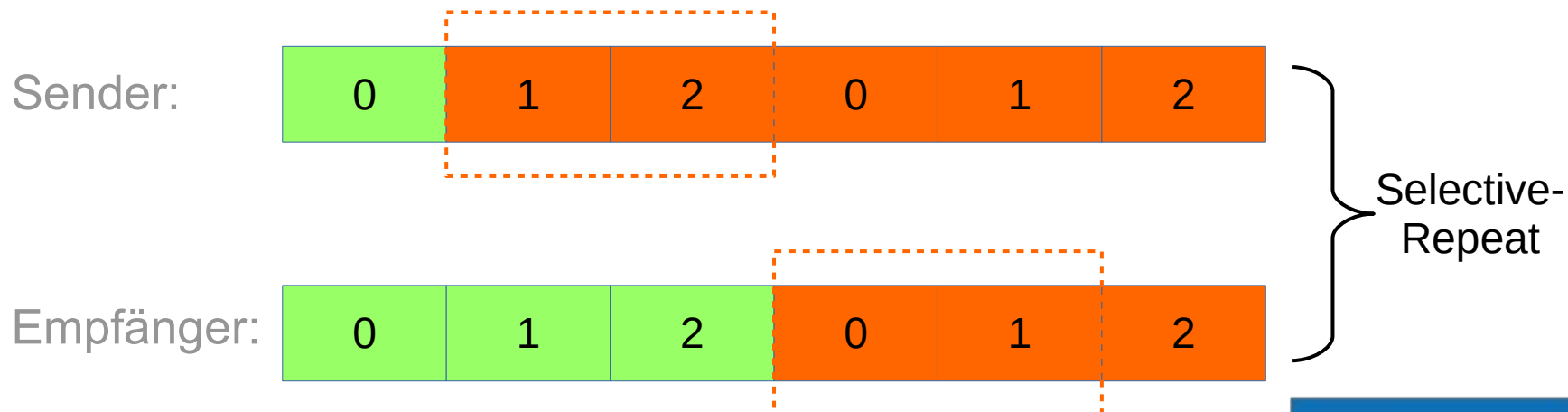
Sliding Window

- Protokolldesign: Bitfeld für die Sequenznummer
- Beschränkt, z. B. n Bits $\rightarrow 2^n$ unterschiedliche Sequenznummern
- *Wie hängt die Wahl der Fenstergröße mit dem Wertebereich für die Sequenznummer zusammen? Berücksichtigen Sie, ob Go-Back-N oder Selective Repeat zum Einsatz kommt.*
- Tipp: Worst-Case-Szenario konstruieren!



Sliding Window

- Protokolldesign: Bitfeld für die Sequenznummer
- Beschränkt, z. B. n Bits $\rightarrow 2^n$ unterschiedliche Sequenznummern
- *Wie hängt die Wahl der Fenstergröße mit dem Wertebereich für die Sequenznummer zusammen? Berücksichtigen Sie, ob Go-Back-N oder Selective Repeat zum Einsatz kommt.*
- Tipp: Worst-Case-Szenario konstruieren!



Sliding Window

- Fazit: Für Fenstergröße F gilt:
 - **Go-Back-N**: Mindestens $F + 1$ unterschiedl. Sequenznummern
 - **Selective Repeat**: Mindestens $2F$ unterschiedl. Sequenznummern

- Zur Wiederholung zuhause:

Skizzieren Sie das Sende- und das Empfangsfenster für ein Übertragungsszenario mit Selective Repeat und einer Fenstergröße von 4.

Sliding Window

- Weitere Aspekte des Sliding-Window-Konzepts:
 - *Erläutern Sie die Begriffe der Fluss- und Verstopfungskontrolle. Stellen Sie den Zusammenhang zur Fenstergröße her.*
 - **Flusskontrolle** (engl. *flow control*): Ende-zu-Ende!
Abhängig von der Puffergröße und Verarbeitungsgeschwindigkeit des Empfängers
 - **Verstopfungskontrolle** (engl. *congestion control*): Zwischenstationen!
Abhängig vom Durchsatz (!) des gesamten Netzwerkes und dessen derzeitiger Auslastung
- Window-Size als Funktion von beiden Größen dynamisch aushandeln!

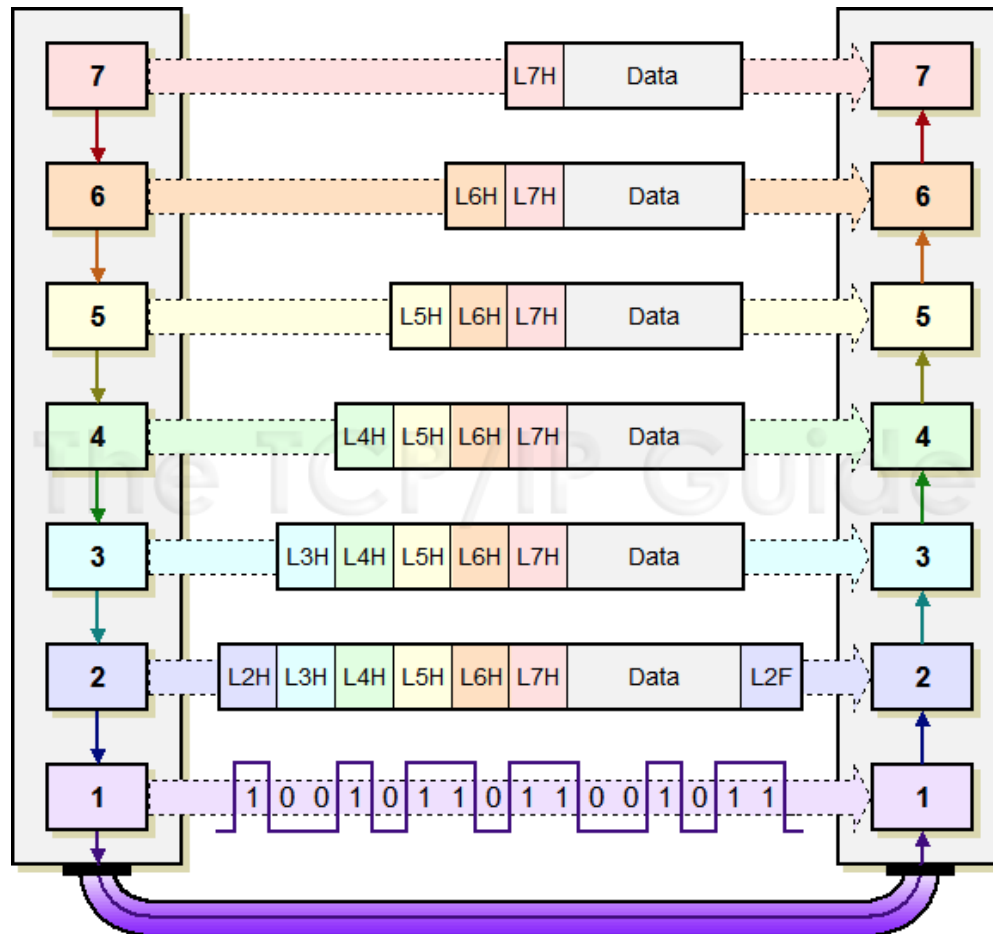
HDLC

- **Aufgabe 3.3)**

Wofür steht die Abkürzung HDLC? Ordnen Sie das Protokoll in das OSI-Schichtenmodell ein.

- High-Level Data Link Control

HDLC



HDLC

- **Aufgabe 3.3)**

Wofür steht die Abkürzung HDLC? Ordnen Sie das Protokoll in das OSI-Schichtenmodell ein.

- High-Level Data Link Control
- Schicht 2 / Sicherungsschicht / Data Link Layer
 - Framing: Aufteilen der Daten in Frames
 - Fehlererkennung: FCS
 - Sequenzierung
 - Flusskontrolle
 - Zugriff auf das Medium
 - Allgemein formuliert: *Zuverlässige Übertragung*

HDLC

- High-Level Data Link Control
- Schicht 2 / Sicherungsschicht / Data Link Layer
- ISO 13239
- Verbindungsorientiert oder verbindungslos
- Punkt-zu-Punkt (oder Punkt-zu-Multipunkt)
- Anwendungen:
 - X.25 (LAPB)
 - ISDN (LAPD)
 - PPP

HDLC

- HDLC-Frame:

Flag 8 bit	Address ≥ 8 bit	Control 8 / 16 bit	Payload ?	FCS 16 / 32 bit	Flag 8 bit
---------------	--------------------	-----------------------	--------------	--------------------	---------------

- *Erläutern Sie die Begriffe Simplex, Halbduplex und Vollduplex. Nennen Sie Beispiele für jede Kategorie.*
- HDLC spezifiziert drei Übertragungsmodi:
 - *Normal Response Mode (NRM) per Halbduplex*
 - *Asynchronous Response Mode (ARM) per Vollduplex*
 - *Asynchronous Balanced Mode (ABM) per Vollduplex*

HDLC

- Frame-Typen, am *Control*-Feld unterscheidbar:
 - I-Frame: Datenübertragung, *Piggyback*-Bestätigung

0	1	2	3	4	5	6	7
0	Sequenz-Nummer (Sender)			P/F- Bit	Sequenz-Nummer (Empfänger)		

HDLC

- Frame-Typen, am *Control*-Feld unterscheidbar:
 - S-Frame: ACK / NAK / Flusskontrolle

0	1	2	3	4	5	6	7
1	0	Funktionsbits		P/F-Bit	Sequenz-Nummer (Empfänger)		

0	0	Receive Ready (RR)
1	0	Receive Not Ready (RNR)
0	1	Reject (REJ)
1	1	Selective Reject (SREJ)

HDLC

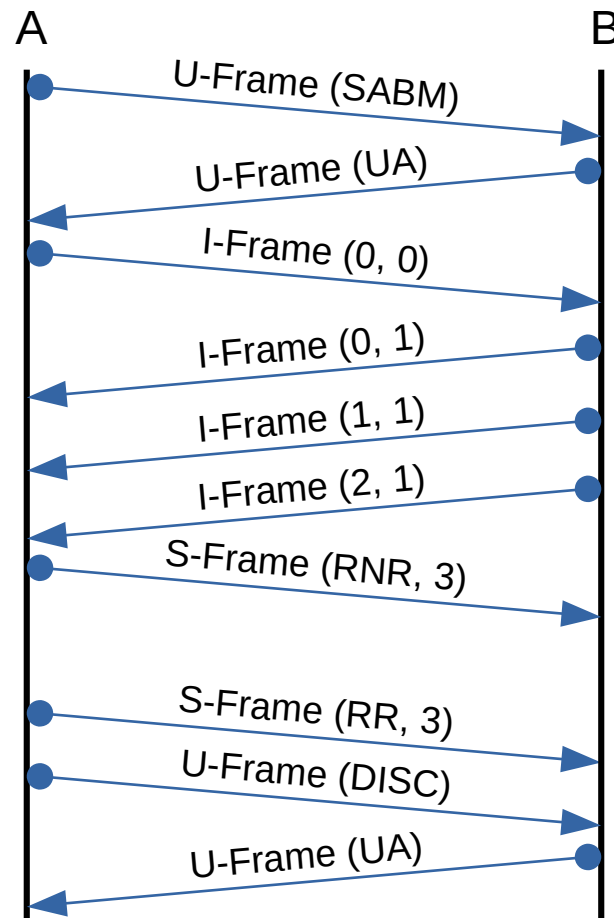
- Frame-Typen, am *Control*-Feld unterscheidbar:
 - U-Frame: Steuerungsinformationen

0	1	2	3	4	5	6	7
1	1	Funktionsbits		P/F-Bit	Funktionsbits		

0	0	0	0	1	Set normal response mode (SNRM)
1	1	0	0	0	Set asynchronous response mode (SARM)
1	1	1	0	0	Set asynchronous balanced mode (SABM)
...

HDLC

- Beispiel:



HDLC

- Bit-Stuffing:
 - *Welche Vorteile bietet das Einrahmen eines HDLC-Frames mit Flags im Gegensatz zu einem Längenfeld?*
 - Flag (**0x7E** bzw. **01111110**) darf nirgendwo sonst im Rahmen auftreten
 - Nach FCS-Berechnung, direkt vor der Übergabe an Schicht 1: **Bit-Stuffing**
 - Nach fünf aufeinanderfolgenden Einsen eine Null einfügen
 - Empfänger: Null nach fünf Einsen transparent entfernen, erst danach FCS berechnen

HDLC

- Bit-Stuffing:
 - Beispiel: ASCII-Texte
 - 0x7E: ~ (Tilde)
 - Und sonst?
 - *Stellen Sie das Payload-Feld eines HDLC-Frames „on the wire“ dar, der folgende Nachricht enthält:*

Ermäßigung? Nö!

→ ISO-8859-1 (Latin-1)

→ niederwertigstes Bit (LSB)

immer zuerst

Scan-code	ASCII hex dez	Zeichen	Scan-code	ASCII hex dez	Zch.	Scan-code	ASCII hex dez	Zch.	Scan-code	ASCII hex dez	Zch.
	00 0	NUL ^@		20 32	SP		40 64	@	0D	60 96	·
	01 1	SOH ^A	02	21 33	!	1E	41 65	A	1E	61 97	a
	02 2	STX ^B	03	22 34	"	30	42 66	B	30	62 98	b
	03 3	ETX ^C	29	23 35	#	2E	43 67	C	2E	63 99	c
	04 4	EOT ^D	05	24 36	\$	20	44 68	D	20	64 100	d
	05 5	ENQ ^E	06	25 37	%	12	45 69	E	12	65 101	e
	06 6	ACK ^F	07	26 38	&	21	46 70	F	21	66 102	f
	07 7	BEL ^G	0D	27 39	'	22	47 71	G	22	67 103	g
0E	08 8	BS ^H	09	28 40	(23	48 72	H	23	68 104	h
0F	09 9	TAB ^I	0A	29 41)	17	49 73	I	17	69 105	i
	0A 10	LF ^J	1B	2A 42	*	24	4A 74	J	24	6A 106	j
	0B 11	VT ^K	1B	2B 43	+	25	4B 75	K	25	6B 107	k
	0C 12	FF ^L	33	2C 44	,	26	4C 76	L	26	6C 108	l
1C	0D 13	CR ^M	35	2D 45	-	32	4D 77	M	32	6D 109	m
	0E 14	SO ^N	34	2E 46	.	31	4E 78	N	31	6E 110	n
	0F 15	SI ^O	08	2F 47	/	18	4F 79	O	18	6F 111	o
	10 16	DLE ^P	0B	30 48	0	19	50 80	P	19	70 112	p
	11 17	DC1 ^Q	02	31 49	1	10	51 81	Q	10	71 113	q
	12 18	DC2 ^R	03	32 50	2	13	52 82	R	13	72 114	r
	13 19	DC3 ^S	04	33 51	3	1F	53 83	S	1F	73 115	s
	14 20	DC4 ^T	05	34 52	4	14	54 84	T	14	74 116	t
	15 21	NAK ^U	06	35 53	5	16	55 85	U	16	75 117	u
	16 22	SYN ^V	07	36 54	6	2F	56 86	V	2F	76 118	v
	17 23	ETB ^W	08	37 55	7	11	57 87	W	11	77 119	w
	18 24	CAN ^X	09	38 56	8	2D	58 88	X	2D	78 120	x
	19 25	EM ^Y	0A	39 57	9	2C	59 89	Y	2C	79 121	y
	1A 26	SUB ^Z	34	3A 58	:	15	5A 90	Z	15	7A 122	z
01	1B 27	Esc ^[33	3B 59	;		5B 91	[7B 123	{
	1C 28	FS ^\	2B	3C 60	<		5C 92	\		7C 124	
	1D 29	GS ^]	0B	3D 61	=		5D 93]		7D 125	}
	1E 30	RS ^^	2B	3E 62	>	29	5E 94	^		7E 126	~
	1F 31	US ^_	0C	3F 63	?	35	5F 95	_	53	7F 127	DEL

Latin-1(ä) = E4

Latin-1(ß) = DF

Latin-1(ö) = F6