FreeDV Treffen vor Weihnachten

1. Begrüßung (Andreas, DM4AB)
2. Planung weitere **Funktermine**, nächste **Plenartermine** 2022 (gemeinsam)

3. **Stimmungsbild** in der Gruppe (jeder)

4. Konzept-**Landkarte** FreeDV
5. den TRX richtig **aussteuern**
6. **Übertragungsfehler** korrigieren
7. Symbole: **bit/s** vs. **Baud**

8. Fragen, Ergänzungen, Wünsche (alle)

2. wöchentliche Funktermine und Plenartreffen

wir haben umfangreich abgestimmt und ermittelt:

- wir treffen uns in Zukunft am

  Donnerstag ab 19:30 Uhr auf 3.625 kHz

  zum wöchentlichen Funken (Ausnahme: 27. Januar 2022)

- unsere nächsten Plenartermine sein werden
  - Freitag, 7. Januar, ab 19:30 Uhr
  - Donnerstag, 27. Januar, ab 19:30 Uhr

Kalender 2022

Januar							
KW	Mo	Di	Mi	Do	Fr	Sa	So
52						1	2
1	3	4	5	6	7	8	9
2	10	11	12	13	14	15	16
3	17	18	19	20	21	22	23
4	24	25	26	27	28	29	30
5	31						

FreeDV Treffen vor Weihnachten

1. Begrüßung (Andreas, DM4AB)
2. Planung weitere **Funktermine**, nächste **Plenartermine** 2022 (gemeinsam)
3. **Stimmungsbild** in der Gruppe (jeder)
4. Konzept-**Landkarte** FreeDV
5. den TRX richtig **aussteuern**
6. **Übertragungsfehler** korrigieren
7. Symbole: **bit/s** vs. **Baud**
8. Fragen, Ergänzungen, Wünsche (alle)

4. Konzept-Landkarte FreeDV

bisher benutzt

	Date	Codec		Modem	Pilot	Diversity	RF BW	Raw bits/s	Fehler-Korrektur	Text	SNR min	Multipath
1600	2012	Codec2	1300	14 DQPSK	DBPSK	-	1125	1600	Golay (23,12)	X	4	poor
700C	2017	Codec2	700C	14 carrier coherent QPSK	-	yes	1500	1400	-----	-	2	good
700D	2018	Codec2	700C	17 carrier coherent OFDM/QPSK	-	-	1000	1900	LDPC (224,112)	X	-2	fair
700E	2020	Codec2	700C	21 carrier coherent OFDM/QPSK	-	-	1500	3000	LDPC (112,56)	X	1	good
2020	2019	LPCNet 1733		31 carrier coherent OFDM/QPSK	-	-	1600	3000	LDPC (504,396)	X22.2	2	poor

4. Heute Abend…

5. den TRX richtig aussteuern

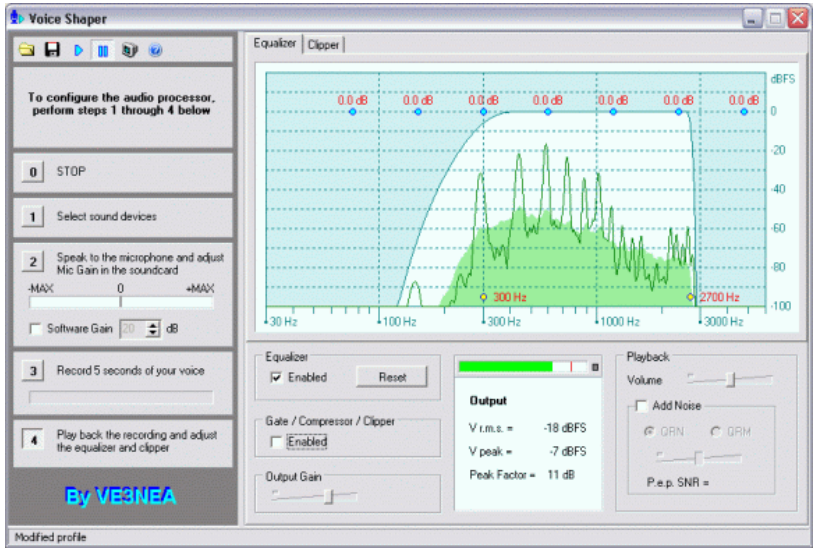
	Date	Codec		Modem	Pilot	Diversity	RF BW	Raw bits/s	Fehler-Korrektur	Text	SNR min	Multipath
1600	2012	Codec2	1300	14 DQPSK	DBPSK	-	1125	1600	Golay (23,12)	X	4	poor
700C	2017	Codec2	700C	14 carrier coherent QPSK	-	yes	1500	1400	-----	-	2	good
700D	2018	Codec2	700C	17 carrier coherent OFDM/QPSK	-	-	1000	1900	LDPC (224,112)	X	-2	fair
700E	2020	Codec2	700C	21 carrier coherent OFDM/QPSK	-	-	1500	3000	LDPC (			
2020	2019	LPCNet 1733		31 carrier coherent OFDM/QPSK			1600	3000	LDPC (504,396)	X22.2	2	poor

6. Fehler korrigieren

7. bit/s vs. Baud

5. den TRX richtig aussteuern

- bei SSB Phonie verwenden wir (typisch) einen Kompressor oder Sprachprozessor
- dadurch wird die **mittlere Leistung des Sprachsignals** erhöht (leise Passagen werden verstärkt, laute Passagen nicht so stark, eventuell sogar bedämpft)
- gute Anregung dazu: **Voice Shaper**, http://www.dxatlas.com/vshaper/



- wenn das Sprachsignal „zu sehr aussteuert“, dann reduziert die TX-ALC den Aussteuerungs-Pegel
- bei SSB Phonie will der OP das sehen, weil dann der Sprachprozessor richtig arbeitet und die mittlere Sendeleistung erhöht wird
- Überkompression verschlechtert real die Sprachqualität („sehr präsent“, DX-/ Pileup-Modulation)

wie ist das bei Datenmodi?

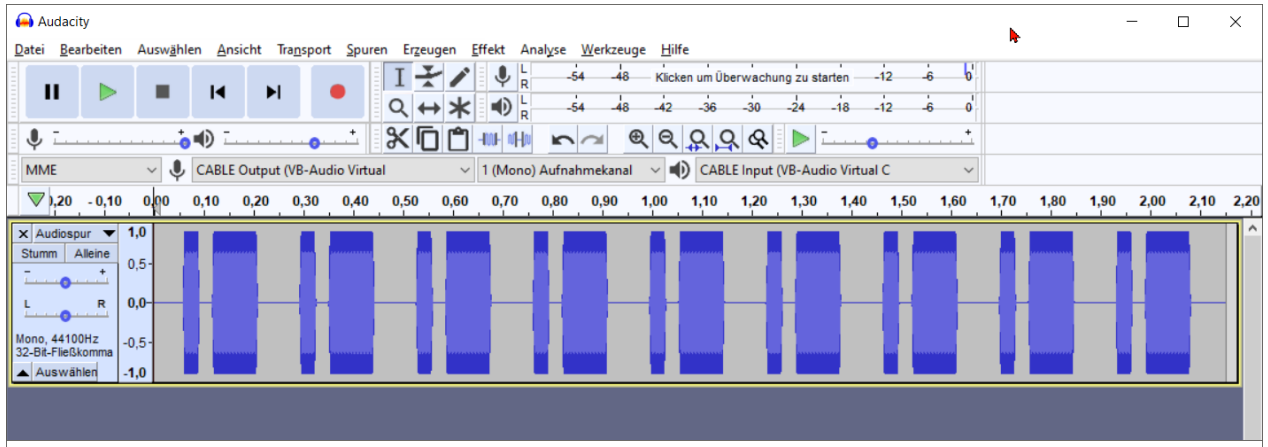
5. den TRX richtig aussteuern

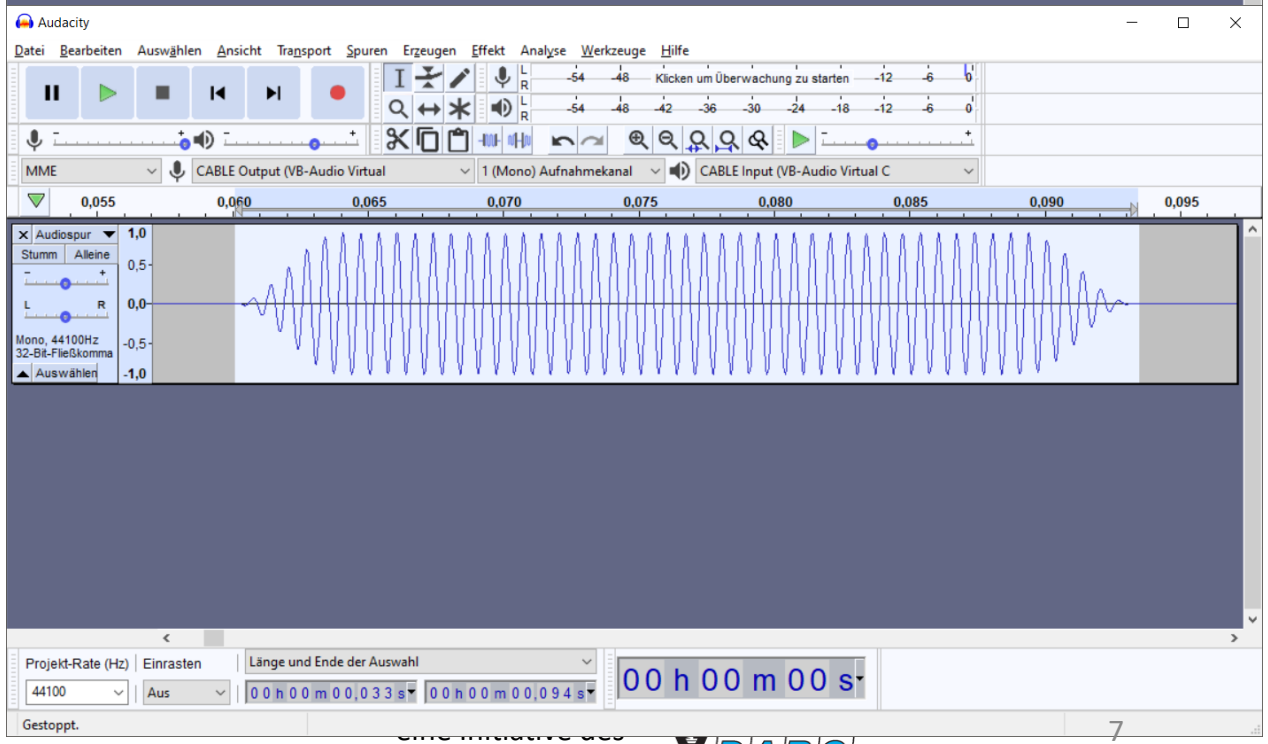
Versuchsaufbau 1:

- FLDIGI
- Virtuelles Audio-Kabel
  https://vb-audio.com/Cable/
- Audacity

- zuerst „CW“: „aaaaaaaaa“

- alles klar: „smoothe“ Einton-Tastung, konstante Amplitude
- **Vollaussteuerung bei Key-Down**



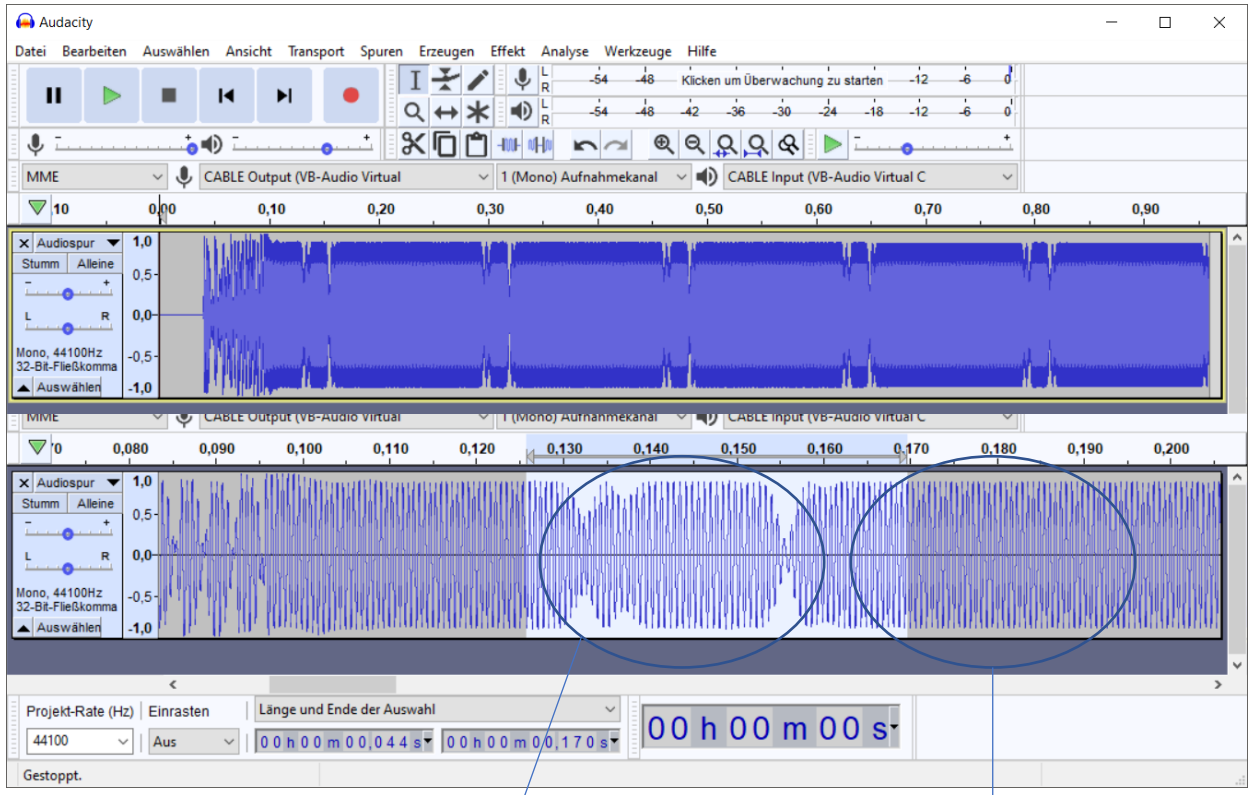


5. den TRX richtig aussteuern

Versuchsaufbau 2:

- FLDIGI
- Virtuelles Audio-Kabel
  https://vb-audio.com/Cable/
- Audacity

- RTTY im Leerlauf
- Wechsel zwischen zwei Frequenzen (mark, space)
- FLDIGI regelt für den Shift die Audio-Amplitude herunter (Standard-Einstellung, konfigurierbar)
- Konsequenz: keine konstante Sendeleistung, **gewollter Effekt**



1416 Hz | Δ ~170Hz | 1587 Hz

- Mode ☺

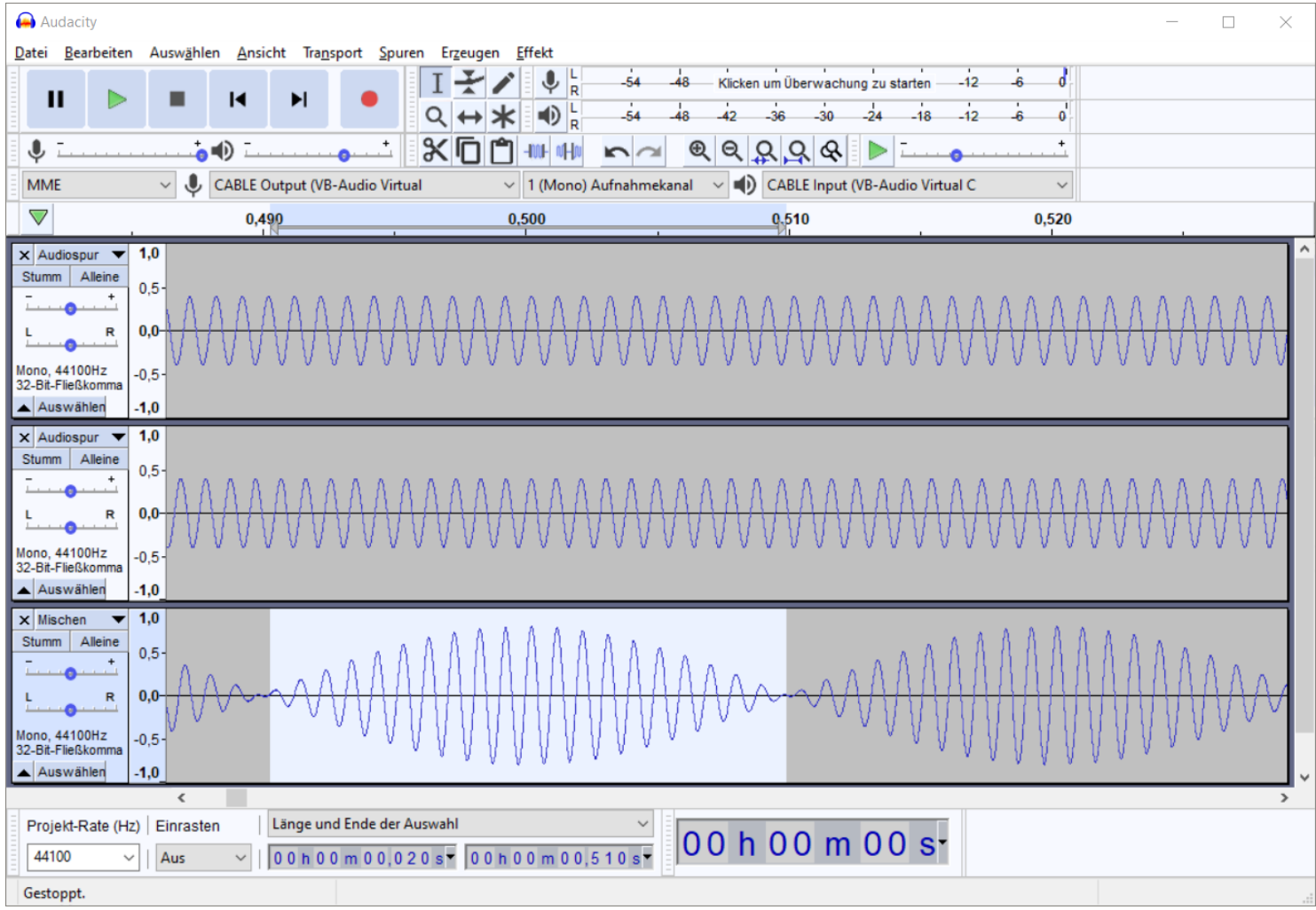
5. den TRX richtig aussteuern

Versuchsaufbau 3:

- Audacity
- Spur1: 1000Hz, Amplitude 0.4
  Spur2: 1050Hz, Amplitude 0.4
  Zweiträger-Signal
- Spur3: Mix, Mischen und Rendern
  Spur1 + Spur2

Ergebnis:

- Maximal-Amplitude 0.8
- Minimal-Amplitude 0



5. den TRX richtig aussteuern

Versuchsaufbau 3:

- Audacity
- Spur1: 1000Hz, Amplitude 0.4
  Spur2: 1050Hz, Amplitude 0.4
  Zweiträger-Signal
- Spur3: Mix, Mischen
  Spur1 + Spur2

- Maximal-Amplitude 0.8
- Minimal-Amplitude 0

700C	14 carrier coherent QPSK	-
700C	17 carrier coherent OFDM/QPSK	-
700C	21 carrier coherent OFDM/QPSK	-

- in FreeDV haben wir ≥ 14 Einzelträger, die sich überlagern
- obwohl „PSK“ eigentlich eine konstante Amplitude hat…
- die Amplitude des FreeDV Summen-signals ist natürlich **gar nicht konstant**!

- fluktuiert ständig
- effektive, gemittelte Sendeleistung ist klein, um „Luft“ für eine maximale TX- Aussteuerung zu haben

“Adjust your transmit levels so the ALC is just being nudged. For a 100W PEP radio, your average power should be 20W. More power is not better with FreeDV. An overdriven signal will have poor SNR at the receiver.”
https://sources.debian.org/src/freedv/1.4.3~1gdc71a1c-1/USER_MANUAL.md/

5. den TRX richtig aussteuern

"Adjust your transmit levels so the ALC is just being nudged. For a 100W PEP radio, your average power should be 20W. More power is not better with FreeDV. An overdriven signal will have poor SNR at the receiver."

Verbesserungs-Idee:
separater **Mic-Pegel** Einsteller in FreeDV; vermeidet ein Verstellen der Pegel für andere Digital-Modi

- dazu passende Beobachtung vom letzten Sonntag
- Sked Teilnehmer aus München mit gutem Signalpegel auf 40m in Kiel

→ aber gar nicht decodierbar weil Pegel zu hoch (ALC @50%); nach reduction viel besser…!

https://sources.debian.org/src/freedv/1.4.3~1gdc71a1c-1/USER_MANUAL.md/

4. Heute Abend…

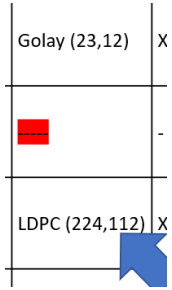
5. den TRX richtig aussteuern

	Date	Codec		Modem	Pilot	Diversity	RF BW	Raw bits/s	Fehler-Korrektur	Text	SNR min	Multipath
1600	2012	Codec2	1300	14 DQPSK	DBPSK	-	1125	1600	Golay (23,12)	X	4	poor
700C	2017	Codec2	700C	14 carrier coherent QPSK	-	yes	1500	1400	-----	-	2	good
700D	2018	Codec2	700C	17 carrier coherent OFDM/QPSK	-	-	1000	1900	LDPC (224,112)	X	-2	fair
700E	2020	Codec2	700C	21 carrier coherent OFDM/QPSK	-	-	1500	3000				
2020	2019	LPCNet 1733		31 carrier coherent OFDM/QPSK			1600	3000	LDPC (504,396)	X22.2	2	poor

6. Fehler korrigieren

7. bit/s vs. Baud

6. Fehler korrigieren



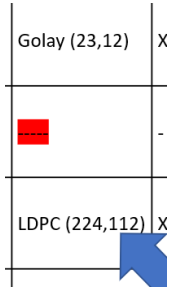
- LDPC (Low-Density-Parity-Check-Codes) als “moderner Standard” (DVB-S, FT8, 5G Mobilfunk, …)
- teilweise mit sehr komplexer Mathematik
- kann viele Fehler erkennen, und einige Fehler korrigieren

- Grund-Konzept: **Redundanz hinzufügen**

einfachstes Beispiel: **Parity Bit**

- nur geeignet bei ganz seltenen Fehlern, kann nicht korrigieren und nur **einen Fehler** erkennen
- bei zwei Fehlern sieht alles wieder gut aus…

6. Fehler korrigieren



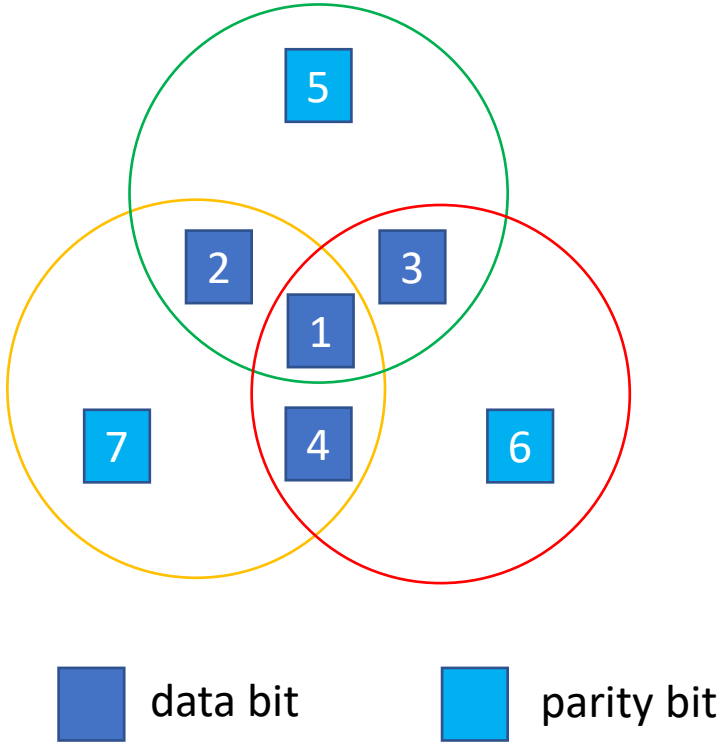
- LDPC (Low-Density-Parity-Check-Codes) als “moderner Standard” (DVB-S, FT8, 5G Mobilfunk, …)
- teilweise mit sehr komplexer Mathematik
- kann viele Fehler erkennen, und einige Fehler korrigieren

- Grund-Konzept: **Redundanz hinzufügen**

einfachstes Beispiel: **Parity Bit**

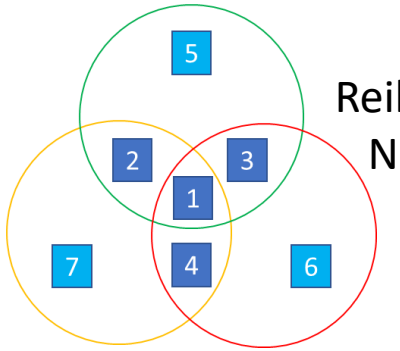
- nur geeignet bei ganz seltenen Fehlern, kann nicht korrigieren und nur **einen Fehler** erkennen
- bei zwei Fehlern sieht alles wider gut aus…

einfacher, ähnlich und nachvollziehbar: Hamming-Code (7,4)



s. auch: http://cmrr-star.ucsd.edu/static/presentations/ldpc_tutorial.pdf

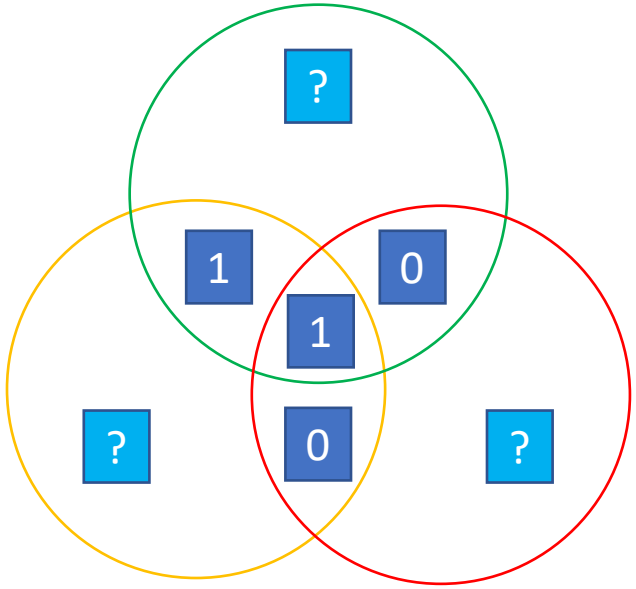
6. Fehler korrigieren



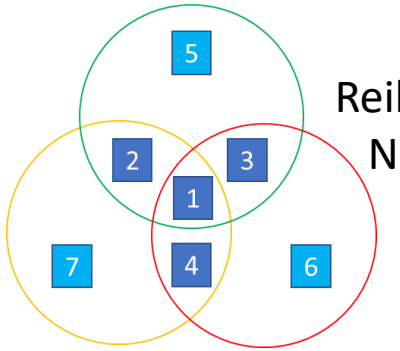
Reihenfolge, Nummerierung

Hamming-Code (7,4)

angenommene Nutzdaten:
„**1 1 0 0**“



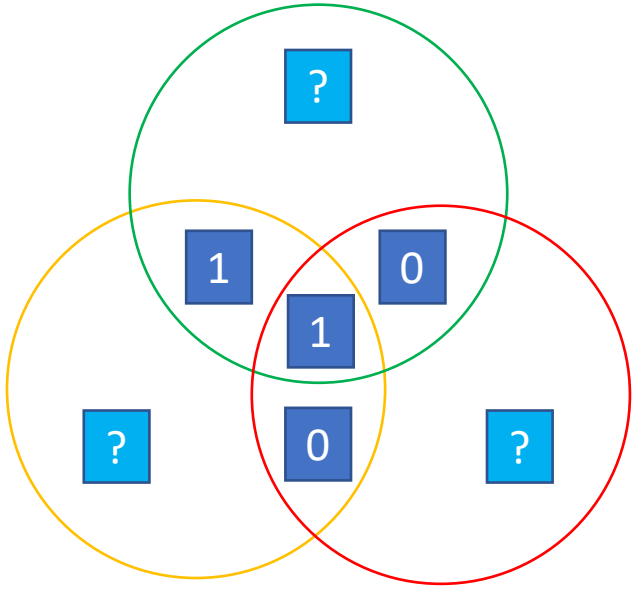
6. Fehler korrigieren



Reihenfolge, Nummerierung

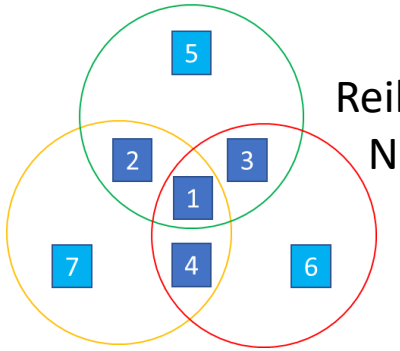
Hamming-Code (7,4)

angenommene Nutzdaten:
„**1 1 0 0**“



Regel: jeder Kreis muss eine gerade Anzahl von „Einsen“ enthalten; das müssen die **Parity Bits** sicherstellen!

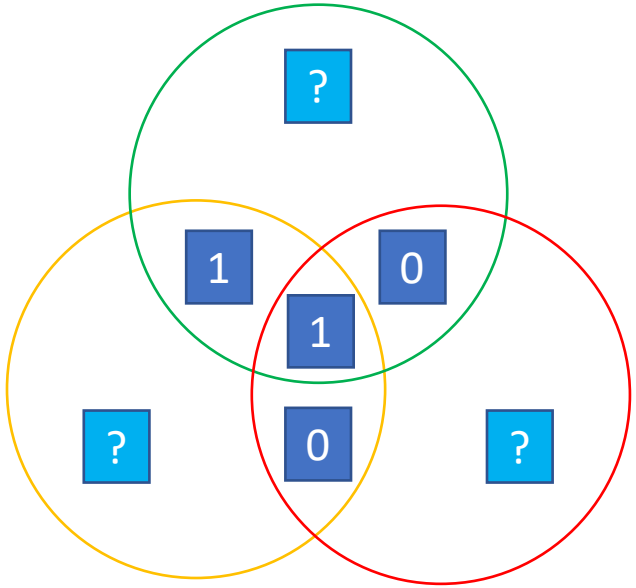
6. Fehler korrigieren



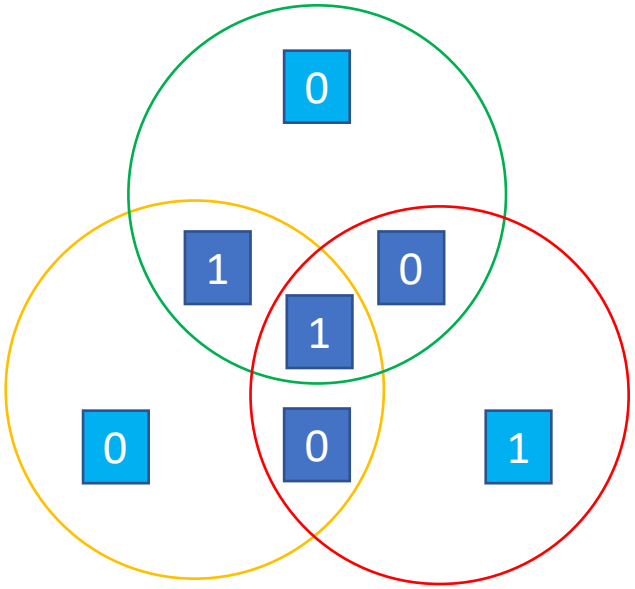
Reihenfolge, Nummerierung

Hamming-Code (7,4)
insgesamt 7bit lang,
davon 4bit Nutzdaten

angenommene Nutzdaten:
„**1 1 0 0**“

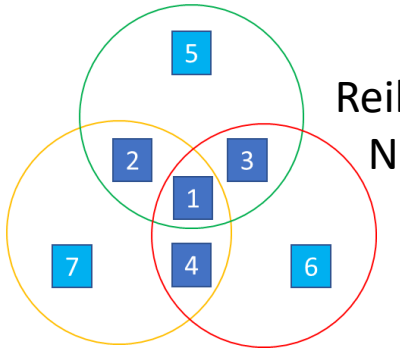


Regel: jeder Kreis muss eine gerade Anzahl von „Einsen“ enthalten; das müssen die **Parity Bits** sicherstellen!



resultierende Übertragungs-Daten mit 3 Bit Redundanz:
„**1 1 0 0 0 1 1**“

6. Fehler korrigieren



Reihenfolge,
Nummerierung

HAM group FreeDV

Hamming-Code (7,4)
insgesamt 7bit lang,
davon 4bit Nutzdaten

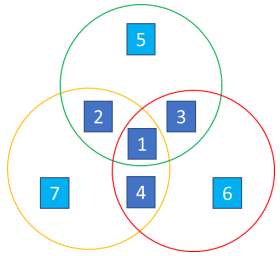
Hamming-Code (7,4) Encoder

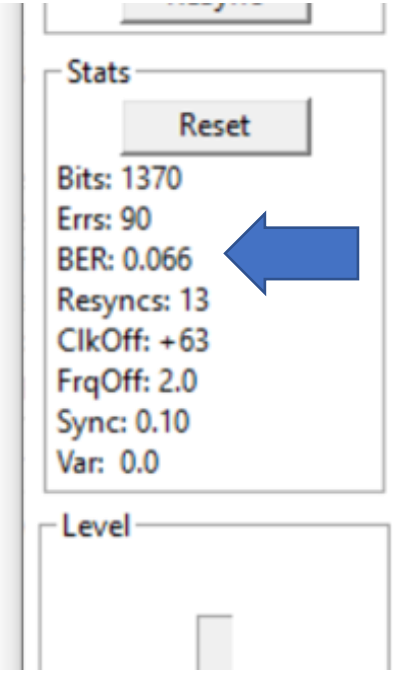
	Auswahl-Spalte	Nutzdaten-Bits				Redundanz-Bits		
0		0	0	0	0	0	0	0
1		0	0	0	1	0	1	1
2		0	0	1	0	1	1	0
3		0	0	1	1	1	0	1
4		0	1	0	0	1	0	1
5		0	1	0	1	1	1	0
6		0	1	1	0	0	1	1
7		0	1	1	1	0	0	0
8		1	0	0	0	1	1	1
9		1	0	0	1	1	0	0
10		1	0	1	0	0	0	1
11		1	0	1	1	0	1	0
12	x	1	1	0	0	0	1	0
13		1	1	0	1	0	0	1
14		1	1	1	0	1	0	0
15		1	1	1	1	1	1	1
	---->	1	1	0	0	0	1	0

ausgew

	Bit-Error								Bit-Error
		↓	↓	↓	↓	↓	↓	↓	
		1	1	0	0	0	1	0	Quell-Symbol mit Fehler
Hamming-Code (7,4) Entscheider									
0		1	1	0	0	0	1	0	3
1		1	1	0	1	0	0	1	4
2		1	1	1	0	1	0	0	4
3		1	1	1	1	1	1	1	7
4		1	0	0	0	1	1	1	4
5		1	0	0	1	1	0	0	3
6		1	0	1	0	0	0	1	3
7		1	0	1	1	0	1	0	4
8		0	1	0	0	1	0	1	3
9		0	1	0	1	1	1	0	4
10		0	1	1	0	0	1	1	4
11		0	1	1	1	0	0	0	3
12		0	0	0	0	0	0	0	0
13		0	0	0	1	0	1	1	3
14		0	0	1	0	1	1	0	3
15		0	0	1	1	1	0	1	4
									0
									minimale Abweichung

6. Fehler korrigieren





wichtige Erkenntnis:

- eine maximale Anzahl von Fehlern kann erkannt werden
- abhängig von den Eigenschaften der FEC (forward error correction) können Fehler auch korrigiert werden

- bis zu einem gewissen Grad, danach muss das Ergebnis verworfen werden

und was dann?

- alte, fehlerfrei empfangene Sprachdaten wiederholen
- Ausgang stumm schalten

FreeDV Treffen vor Weihnachten

1. Begrüßung (Andreas, DM4AB)
2. Planung weitere **Funktermine**, nächste **Plenartermine** 2022 (gemeinsam)
3. **Stimmungsbild** in der Gruppe (jeder)
4. Konzept-**Landkarte** FreeDV
5. den TRX richtig **aussteuern**
6. **Übertragungsfehler** korrigieren
7. Symbole: **bit/s** vs. **Baud**
8. Fragen, Ergänzungen, Wünsche (alle)

Ende der Veranstaltung, 22:15 Uhr