

Digitale Medien

Übung

Heute

- Wiederholung Zahlensysteme
- LZW
- Digitalisierung

Dezimal nach Binär (Komma)

Multipliziere die Zahl mit 2.

Die Zahl vor dem Komma ist die nächste Zahl des Ergebnisses.

Entferne die Zahl vor dem Komma.

Wiederhole das Verfahren, bis

nichts mehr rechts vom Komma steht oder sich die Ergebnisse

Wiederholen oder die gewünschte Genauigkeit erreicht ist.

Beispiel:

0,3125 nach Binär

$$0,3125 \times 2 = \mathbf{0,625}$$

$$0,625 \times 2 = \mathbf{1,25}$$

$$0,25 \times 2 = \mathbf{0,5}$$

$$0,5 \times 2 = \mathbf{1}$$

=> 0,0101

Idee:

Nicht einzelne Zeichen werden günstig kodiert, sondern ganze Zeichenketten (*Wörterbuch-Kompression*).

abcdabcdabcdabcd

Huffman: [a][b][c][d][a][b][c][d][a][b][c][d][a][b][c][d][d][d]

=> 18 Symbole

Wörterbuch: [abcd][abcd][abcd][abcd][dd]

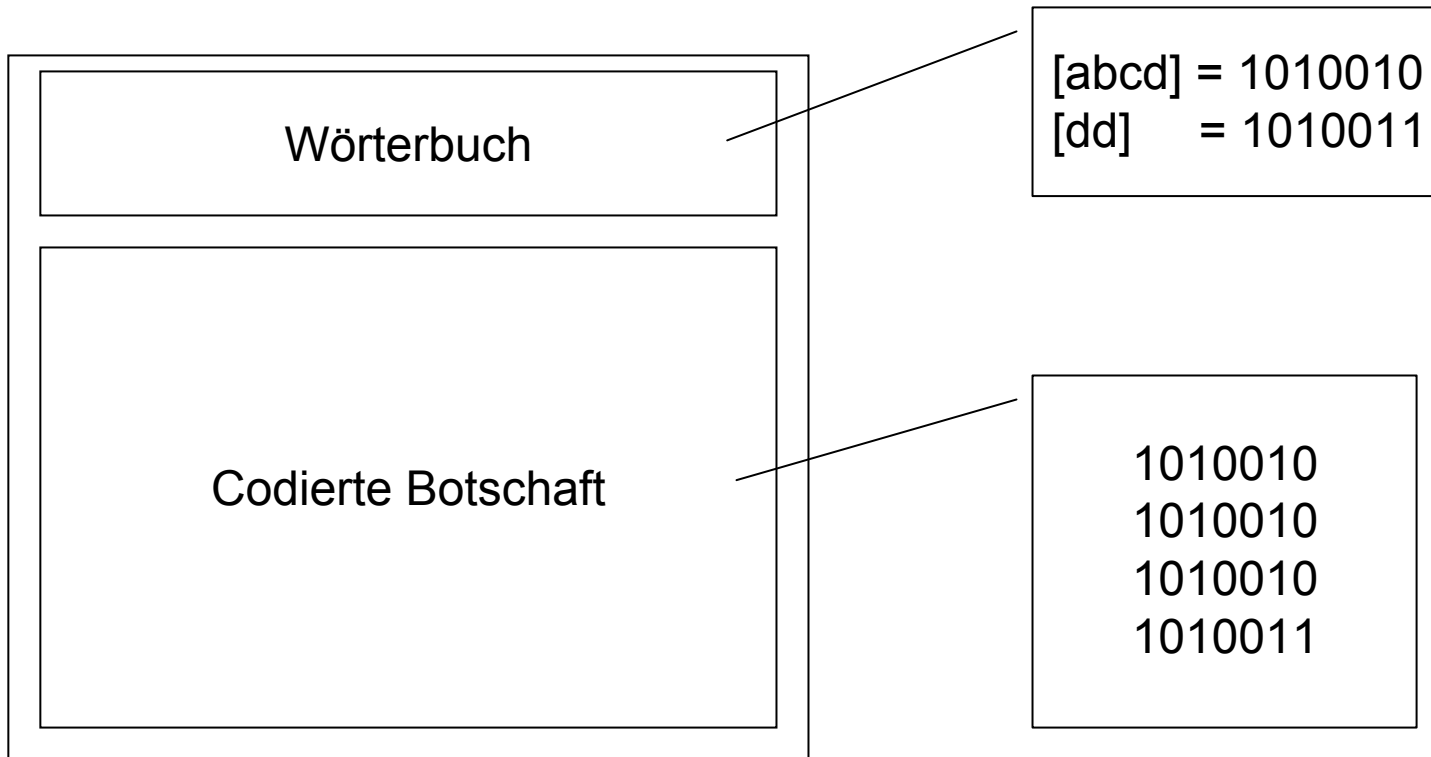
=> 5 Symbole

Nachteil: Mehr Symbole nötig (jeder Eintrag ins Wörterbuch ist ein Symbol)

Beispiele: lateinische Schrift: 26 Symbole

chinesische Schrift: ca. 10.000 Symbole

Weiterer Nachteil: Empfänger muss Wörterbuch kennen, um die Nachricht dekodieren zu können.



Aber: Muss das Wörterbuch wirklich mit übertragen werden?

Bei der [LZW-Komprimierung](#) wird das Wörterbuch *während* der Dekodierung aufgebaut!

Ausgegangen wird von einem Grundwörterbuch (z.B. ASCII-Code)

Algorithmus:

SeqChar p = < Nächstes Eingabezeichen >;

Char k = Nächstes Eingabezeichen;

Wiederhole:

Falls p & < k > in Tabelle enthalten

dann p = p & < k >

sonst trage p & <k> neu in Tabelle ein

 (und erzeuge neuen Index dafür);

 Schreibe Tabellenindex von p auf Ausgabe;

 p = < k >;

Ende Fallunterscheidung;

 k = Nächstes Eingabezeichen;

solange bis Eingabeende

Schreibe Tabellenindex von p auf Ausgabe;

a	97	h	104	o	111	v	118
b	98	i	105	p	112	w	119
c	99	j	106	q	113	x	120
d	100	k	107	r	114	y	121
e	101	l	108	s	115	z	122
f	102	m	109	t	116		
g	103	n	110	u	117		

labala

Lesen (k)	Codetabelle schreiben (p & <k>)	Ausgabe	Puffer füllen (p)
			< >
a	<la>, 256	108 (l)	<a>
b	<ab>, 257	97 (a)	
a	<ba>, 258	98 (b)	<a>
l	<al>, 259	97 (a)	< >
a			<la>
		256 (la)	<>

SeqChar p = < Nächstes Eingabezeichen >;

Char k = Nächstes Eingabezeichen;

Wiederhole:

 Falls p & <k> in Tabelle enthalten

 dann p = p & <k>

 sonst trage p & <k> neu in Tabelle ein

 (und erzeuge neuen Index dafür);

 Schreibe Tabellenindex von p auf Ausgabe;

 p = <k>;

 Ende Fallunterscheidung;

 k = Nächstes Eingabezeichen;

solange bis Eingabeende

Schreibe Tabellenindex von p auf Ausgabe;

ballaballala

Lesen (k)	Codetabelle schreiben (p & <k>)	Ausgabe	Puffer füllen (p)

ballaballala

Lesen (k)	Codetabelle schreiben (p & <k>)	Ausgabe	Puffer füllen (p)
			
a	<ba>, 256	98 (b)	<a>
l	<al>, 257	97 (a)	<l>
l	<ll>, 258	108 (l)	<l>
a	<la>, 259	108 (l)	<a>
b	<ab>, 260	97 (a)	
a			<ba>
l	<bal>, 261	256 (ba)	<l>
l			<ll>
a	<lla>, 262	258 (ll)	<a>
l			<al>
a	<ala>, 263	257 (al)	<a>
		97 (a)	<>

Ergebnis: 98 – 97 – 108 – 108 – 97 – 256 – 258 – 257 - 97

Algorithmus zur Dekomprimierung:

```

SeqChar p := <>;
int k = NächsteEingabezahl;
Schreibe Zeichenreihe mit Tabellenindex k auf Ausgabe;
int old = k;
Wiederhole solange Eingabe nicht leer:
    k = NächsteEingabezahl;
    SeqChar akt = Zeichenreihe mit Tabellenindex k;
    p = Zeichenreihe mit Tabellenindex old (letztes Teilwort);
    Falls Index k in Tabelle enthalten
        dann Char q = erstes Zeichen von akt;
            Schreibe Zeichenreihe akt auf Ausgabe;
        sonst Char q = erstes Zeichen von p;
            Schreibe Zeichenreihe p & <q> auf Ausgabe;
    Ende Fallunterscheidung;
    Trage p & <q> in Tabelle ein
    (und erzeuge neuen Index dafür);
    old = k;
Ende Wiederholung;

```

a	97	h	104	o	111	v	118
b	98	i	105	p	112	w	119
c	99	j	106	q	113	x	120
d	100	k	107	r	114	y	121
e	101	l	108	s	115	z	122
f	102	m	109	t	116		
g	103	n	110	u	117		

Lesen (k)	Ausgabe (akt) (q ist unterstrichen)	Puffer füllen (p)	Codetabelle (p & <q>)	Merken (old)
108	l			108
97	<u>a</u>	l	<la>, 256	97
98	<u>b</u>	a	<ab>, 257	98
97	<u>a</u>	b	<ba>, 258	97
256	<u>l</u> a	a	<al>, 259	256

SeqChar p := <>;

int k = NächsteEingabezahl ;

Schreibe Zeichenreihe mit Tabellenindex k auf Ausgabe;

int old = k;

Wiederhole solange Eingabe nicht leer:

 k = NächsteEingabezahl ;

 SeqChar akt = Zeichenreihe mit Tabellenindex k;

 p = Zeichenreihe mit Tabellenindex old (letztes Teilwort);

 Falls Index k in Tabelle enthalten

 dann Char q = erstes Zeichen von akt;

 Schreibe Zeichenreihe akt auf Ausgabe;

 sonst Char q = erstes Zeichen von p;

 Schreibe Zeichenreihe p & <q> auf Ausgabe;

 Ende Fallunterscheidung;

 Trage p & <q> in Tabelle ein

 (und erzeuge neuen Index dafür);

 old = k;

Ende Wiederholung;

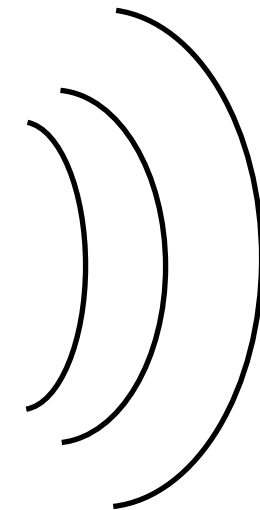
98 – 97 – 108 – 108 – 97 – 256 – 258 – 257 - 97

Lesen (k)	Ausgabe (akt) (q ist unterstrichen)	Puffer füllen (p)	Codetabelle (p & <q>)	Merken (old)

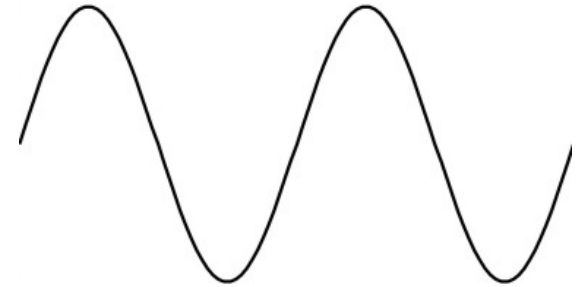
98 – 97 – 108 – 108 – 97 – 256 – 258 – 257 - 97

Lesen (k)	Ausgabe (akt) (q ist unterstrichen)	Puffer füllen (p)	Codetabelle (p & <q>)	Merken (old)
98	b			98
97	<u>a</u>	b	<ba>, 256	97
108	<u>l</u>	a	<al>, 257	108
108	<u>l</u>	l	<ll>, 258	108
97	<u>a</u>	l	<la>, 259	97
256	<u>ba</u>	a	<ab>, 260	256
258	<u>ll</u>	ba	<bal>, 261	258
257	<u>al</u>	ll	<lla>, 262	257
97	<u>a</u>	al	<ala>, 263	97

physikalische Signale



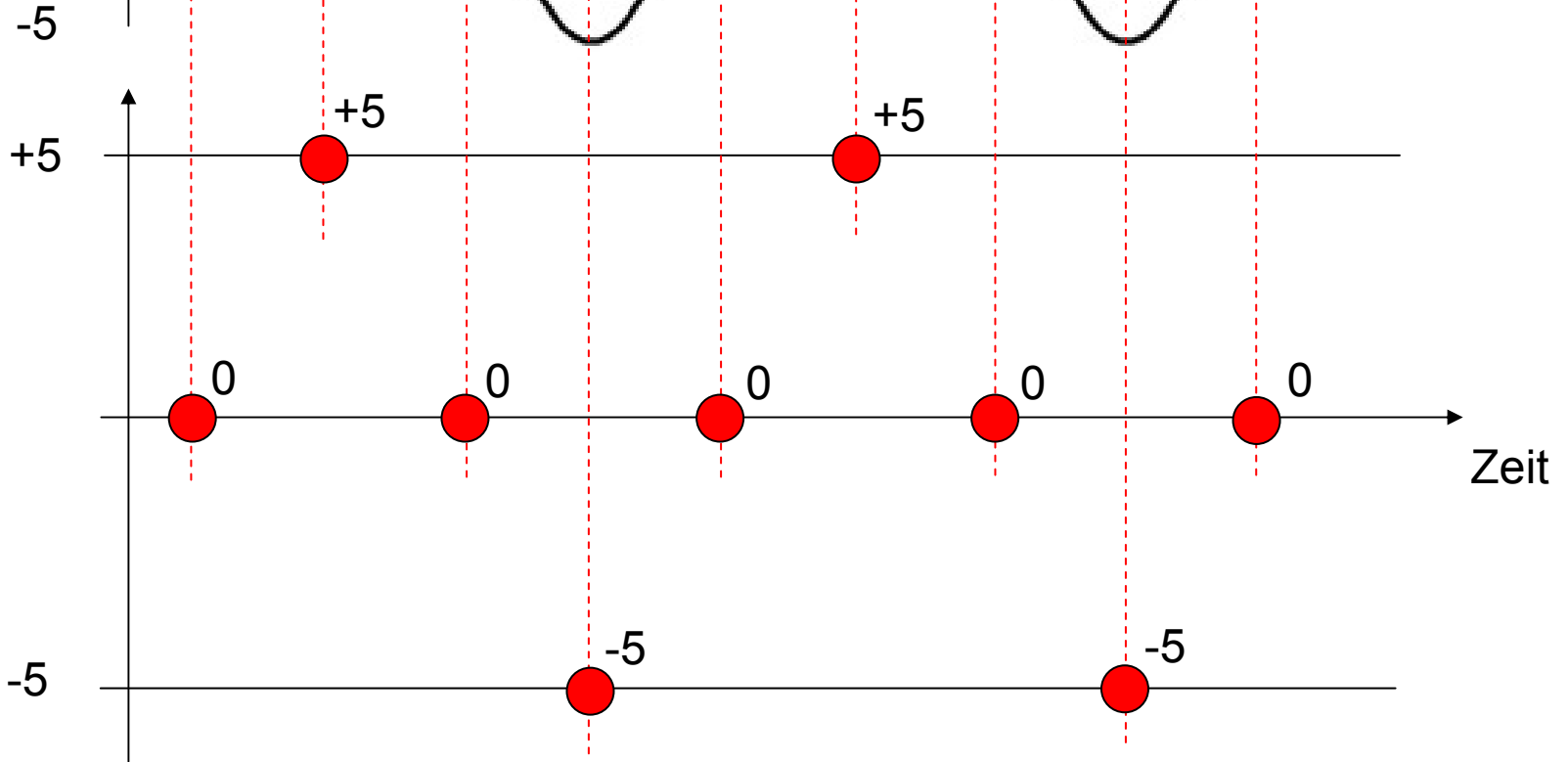
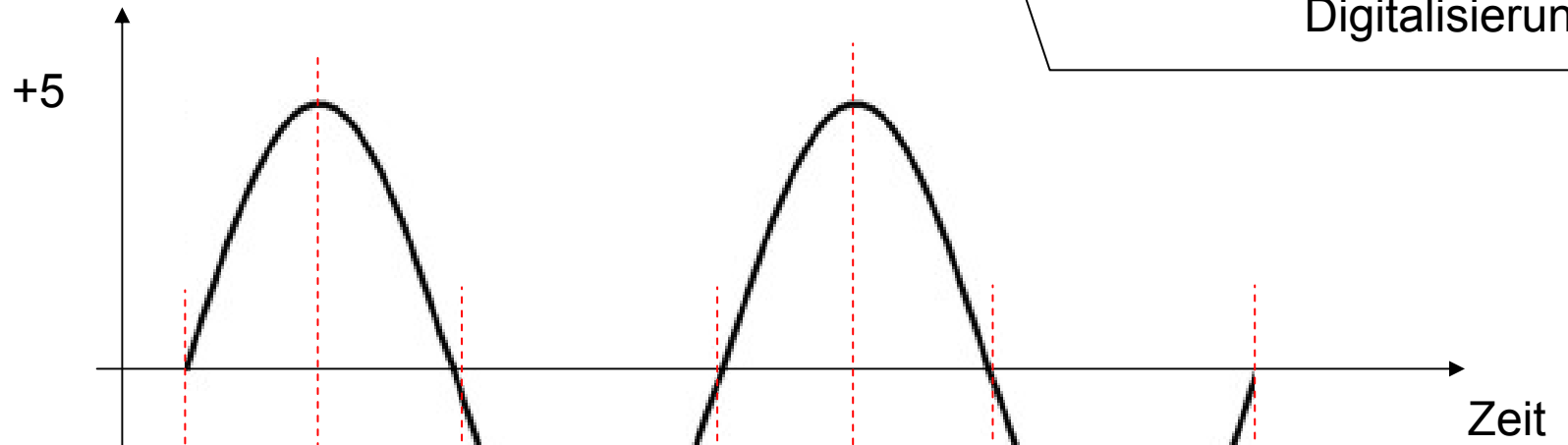
elektrische Signale



digitale Signale



10100100101111010111101



Wellen

Beispiele: Schallwellen, Wasserwellen, Licht, Radiowellen

Eine Welle wird beschrieben durch:

- A Amplitude (z.B. Lautstärke dB)
- c Ausbreitungsgeschwindigkeit
- f Frequenz
bzw.
- λ Wellenlänge (c / f)



Frequenz f wird in Hertz (Hz) gemessen.

1 Hz = 1 Schwingung / Sekunde

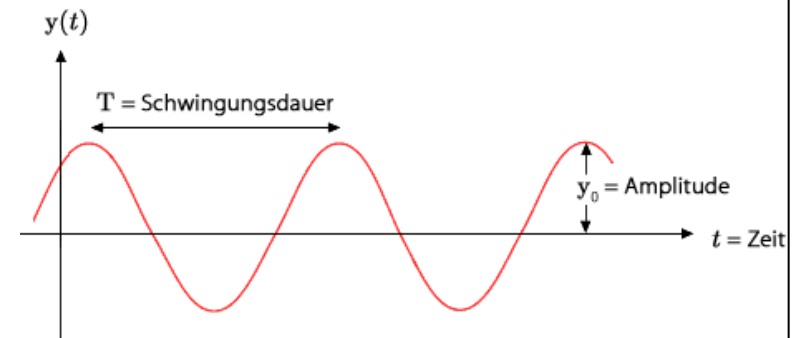
1 Kilohertz (kHz) = 1.000 Schwingungen / Sekunde

1 Megahertz (MHz) = 1.000.000 Schwingungen / Sekunde

...

Schwingungsdauer T ist die Zeit bis sich das Wellenmuster wiederholt.

Wenn eine periodische Welle einmal pro Sekunde schwingt ($T = 1$ sec), hat sie eine Frequenz von 1 Hz, bei 1.000 Schwingungen ($T = 0,001$ sec) 1 kHz, etc.



Allgemein: $f = 1 / T$ (mit T Schwingungsdauer in Sekunden und f Frequenz)

Frequenz Eselsbrücke:

Das Herz eines erwachsenen schlägt ca. 60 mal pro Minute.

D.h. ca. 1 mal pro Sekunde.

Damit hat es eine Frequenz von 1 Hz.

Umrechnung von Schwingungsdauer und Frequenz

Welle mit Schwingungsdauer $T = 2 \text{ ms}$. Wie hoch ist die Frequenz f ?

$$T = 2 \text{ ms} = 2 * 10^{-3} \text{ s} = 0,002 \text{ s}$$

$$f = 1 / T = 1 / 0,002 = 500 \text{ Hz}$$

Welle mit Frequenz $f = 500 \text{ MHz}$. Bestimme Schwingungsdauer T .

$$f = 500 \text{ MHz} = 500.000.000 \text{ Hz}$$

$$f = 1 / T \quad \Rightarrow \quad T = 1 / f$$

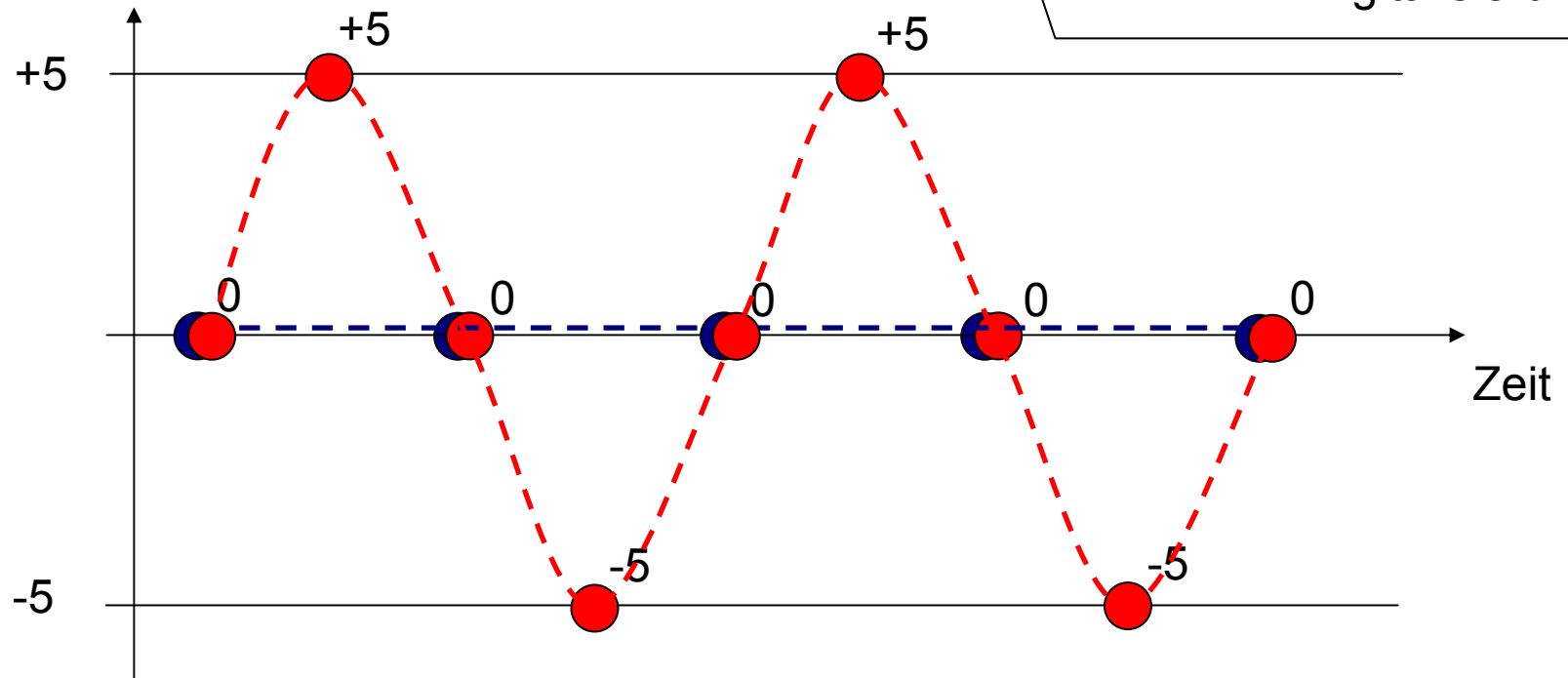
$$T = 1 / 500.000.000 \text{ Hz} = 0,000000002 \text{ s} = 0,000002 \text{ ms} = 0,002 \text{ } \mu\text{s}$$

1 Millisekunde (ms) =

$$1 * 10^{-3} \text{ Sekunden} = 0,001 \text{ Sekunden}$$

1 Mikrosekunde (μs) =

$$1 * 10^{-6} \text{ Sekunden} = 0,000001 \text{ Sekunden}$$



Aus den Samples wird versucht, das Originalsignal zu rekonstruieren.
Je höher die Abtastrate desto besser die Rekonstruktion.

Doppelte Frequenz der Welle (blau) reicht nicht!
Abtastrate f_A muss größer als die doppelte Frequenz f sein:

$$f_A > 2 * f \quad (\text{Nyquist-Theorem})$$

z.B. Audio-CD: Frequenz f maximal 20 kHz, Abtastrate 44,1 kHz

Nächste Woche

- HTML