Research Institute for Future Media Computing Institute of Computer Vision
未来媒体技术与研究所 计算机视觉研究所

多媒体系统导论
Fundamentals of Multimedia System

授课教师：文嘉俊
邮箱：wenjiajun@szu.edu.cn
2024年春季课程

Outline of Lecture 05

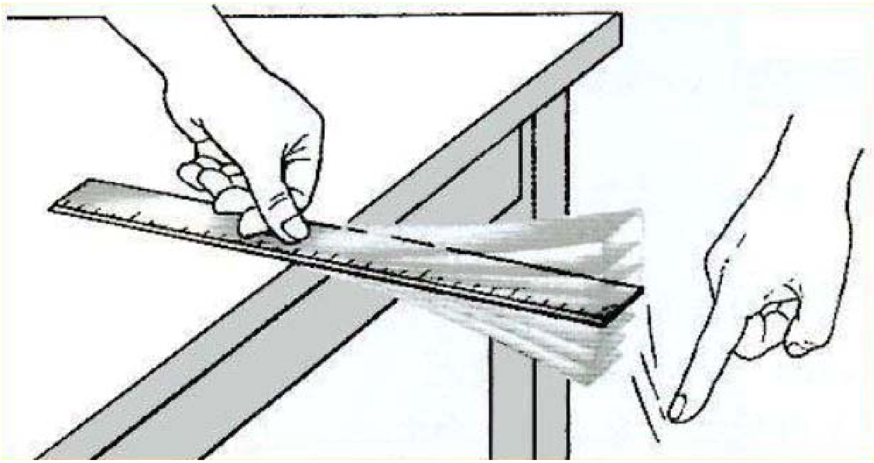
- Digitization of Sound 声音数字化
  - Sound and Digitization 声音和数字化
  - Nyquist Theorem 奈奎斯特理论
  - Signal to Noise Ratio (SNR) 信噪比
  - Linear and Non-linear Quantization 线性和非线性量化
  - Audio Quality and Synthetic Sounds 音频质量和合成声音
- MIDI: Musical Instrument Digital Interface 乐器数字接口
- Quantization and Transmission of Audio 音频的量化和传输
- Experiments 实验

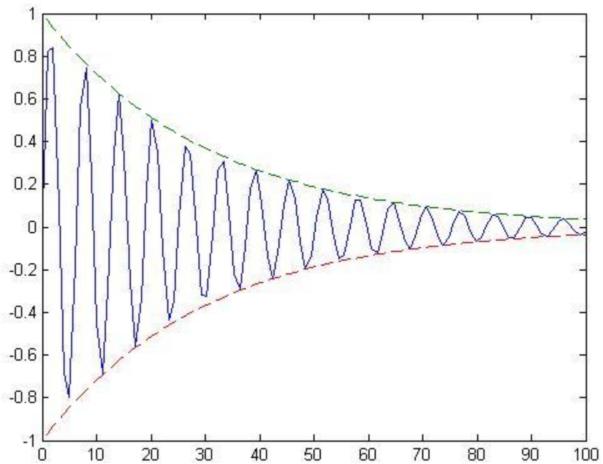
Digitization of Sound

- What is Sound? 什么是声音？
  - Sound is a wave phenomenon like light, but is macroscopic and involves molecules of air being compressed and expanded under the action of some physical device.波动现象，由空气分子压缩膨胀产生

a) For example, a speaker in an audio system vibrates back and forth and produces a longitudinal pressure wave that we perceive as sound.扩音器前后振动产生径向压力波

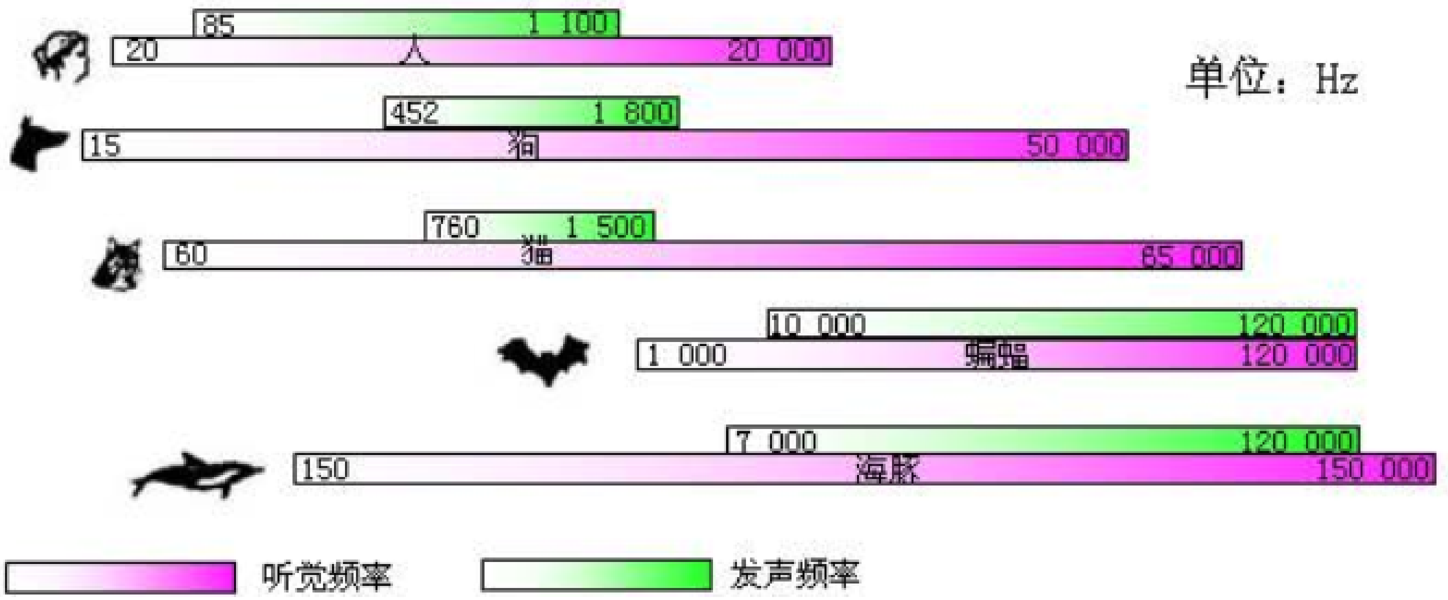
b) Since sound is a pressure wave, it takes on **continuous values**, as opposed to digitized ones. 声波取值连续





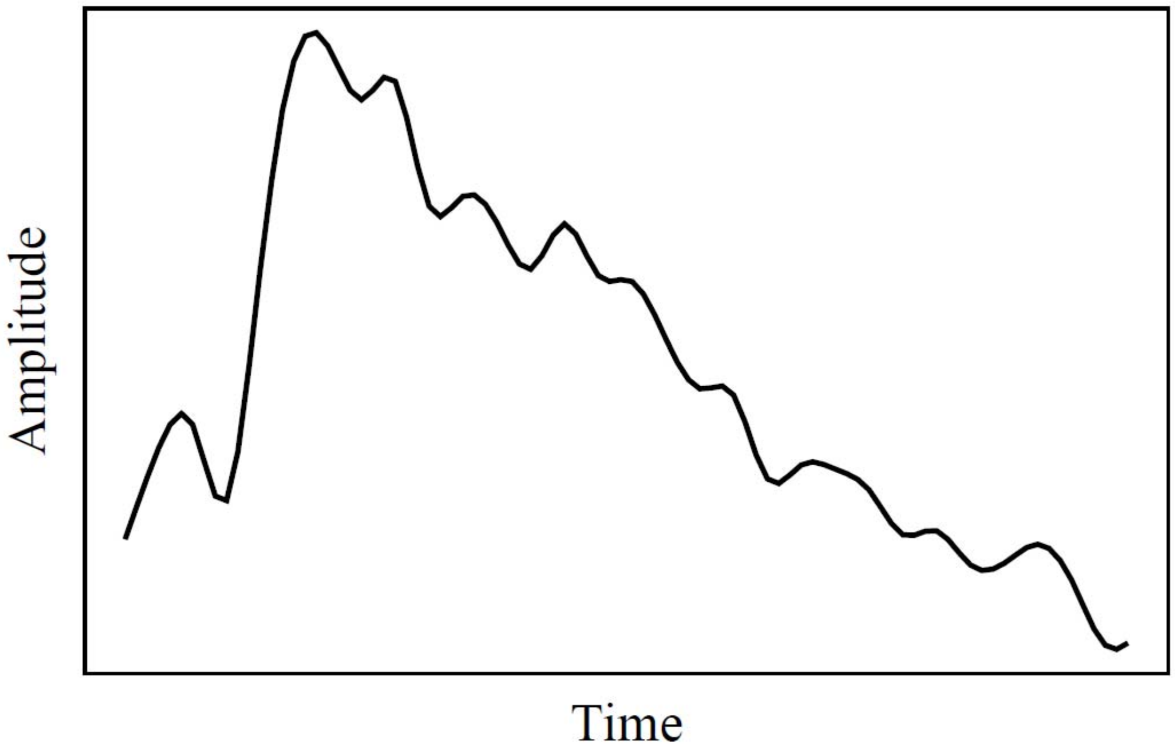
Digitization of Sound

- What is Sound? 声音是什么？
  - Frequency ranges of human and animals. 人和动物听觉频率范围



Digitization of Sound

- Digitization 声音数字化
  - Digitization means conversion to *a stream of numbers*, and preferably these numbers should be **integers** for efficiency. 声音信号可用离散的整形数字来表示



An analog signal: continuous measurement of pressure wave.

Digitization of Sound

- Digitization 声音数字化
  - Digitization means conversion to *a stream of numbers*, and preferably these numbers should be **integers** for efficiency. 声音信号可用离散的整形数字来表示
  - The previous sound signal has to be made digital in both time and amplitude. To digitize, the signal must be sampled in each dimension: in **time**, and in **amplitude**. 信号在两个维度上离散化：时间和幅值

a) **Sampling** means measuring the quantity we are interested in, usually at evenly-spaced intervals. 采样即等间距数值测量

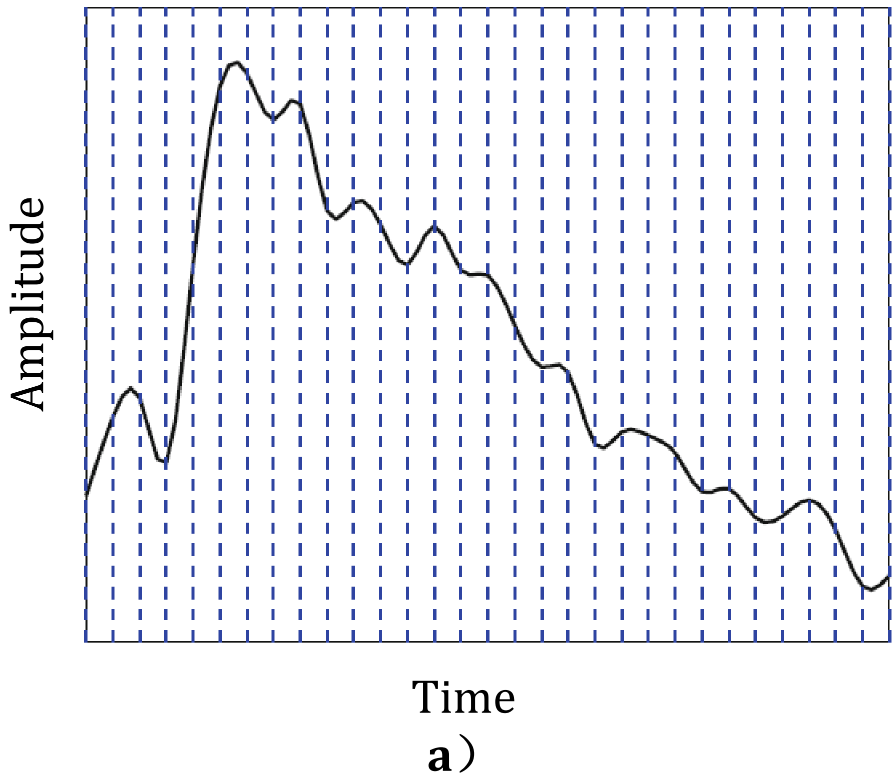
b) The first kind of sampling, using measurements only at evenly spaced time intervals, is simply called, sampling**.** The rate at which it is performed is called the ***sampling frequency***. 采样速度即采样频率

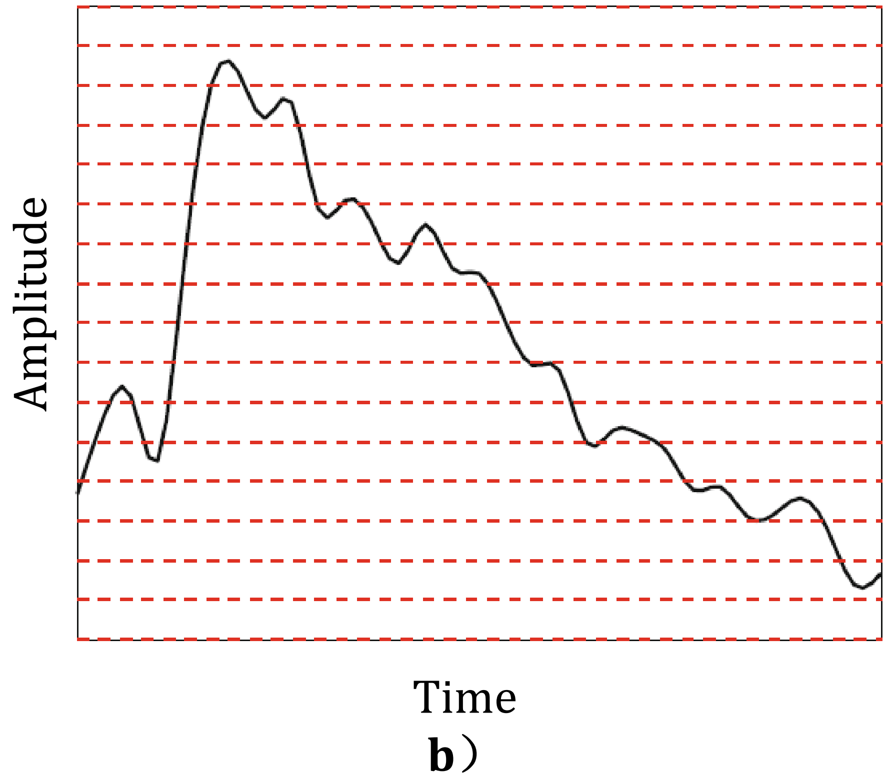
c) For audio, typical sampling rates are from 8 kHz (8,000 samples per second) to 48 kHz. This range is determined by **Nyquist theorem** (discussed later). 采样频率由奈奎斯特理论决定

d) Sampling in the amplitude or voltage dimension is called ***quantization***. 在幅值上的采样称为量化

Digitization of Sound

◆ Digitization 声音数字化





Sampling and quantization: **a)** sampling the analog signal in the time dimension; **b)** quantization is sampling the analog signal in the amplitude dimension.
采样和量化，采样沿着时间方向进行，量化沿着纵向方向进行

Digitization of Sound

- Digitization 声音数字化
  - Thus to decide how to digitize audio data we need to answer the following questions: 声音数字化两个关键问题

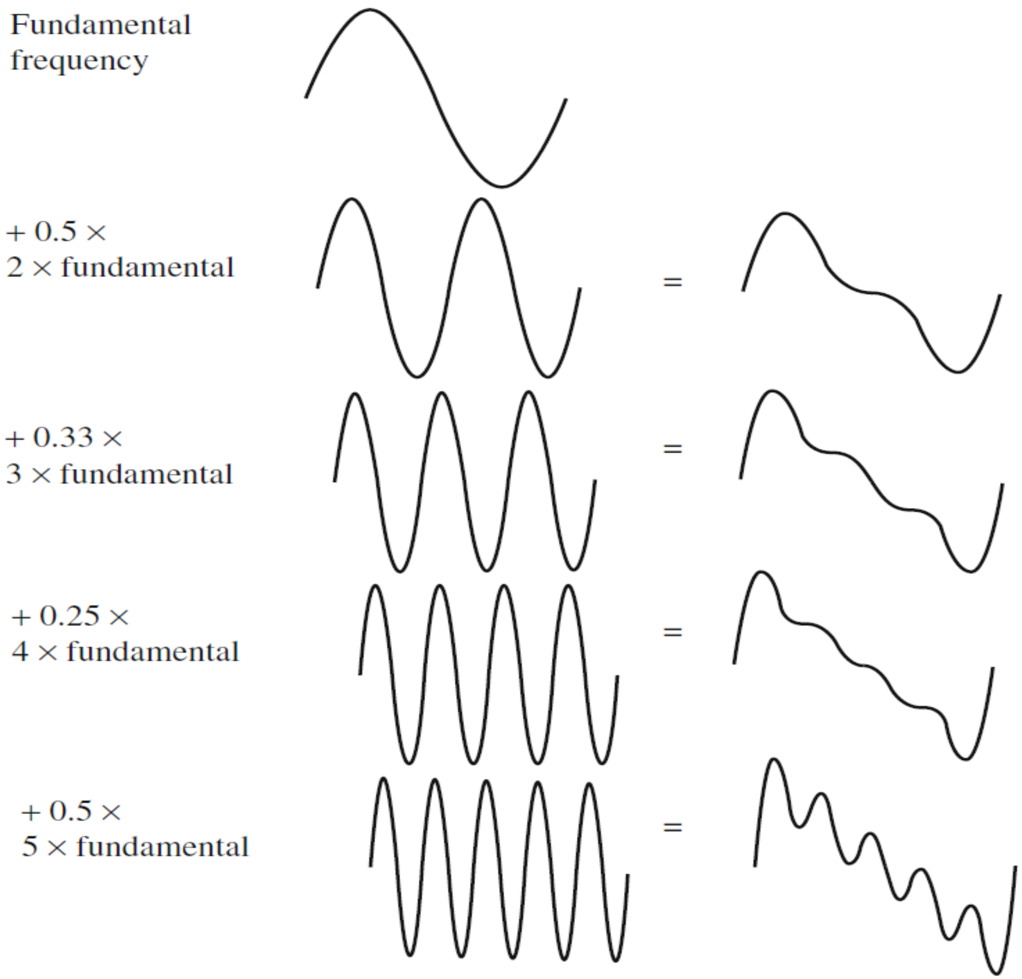
a) What is the sampling rate? 什么是采样率

b) How finely is the data to be quantized, and is quantization uniform? 声音数据需要经过怎样的量化处理?

c) How is audio data formatted? (file format) 声音数据的结构是怎样?

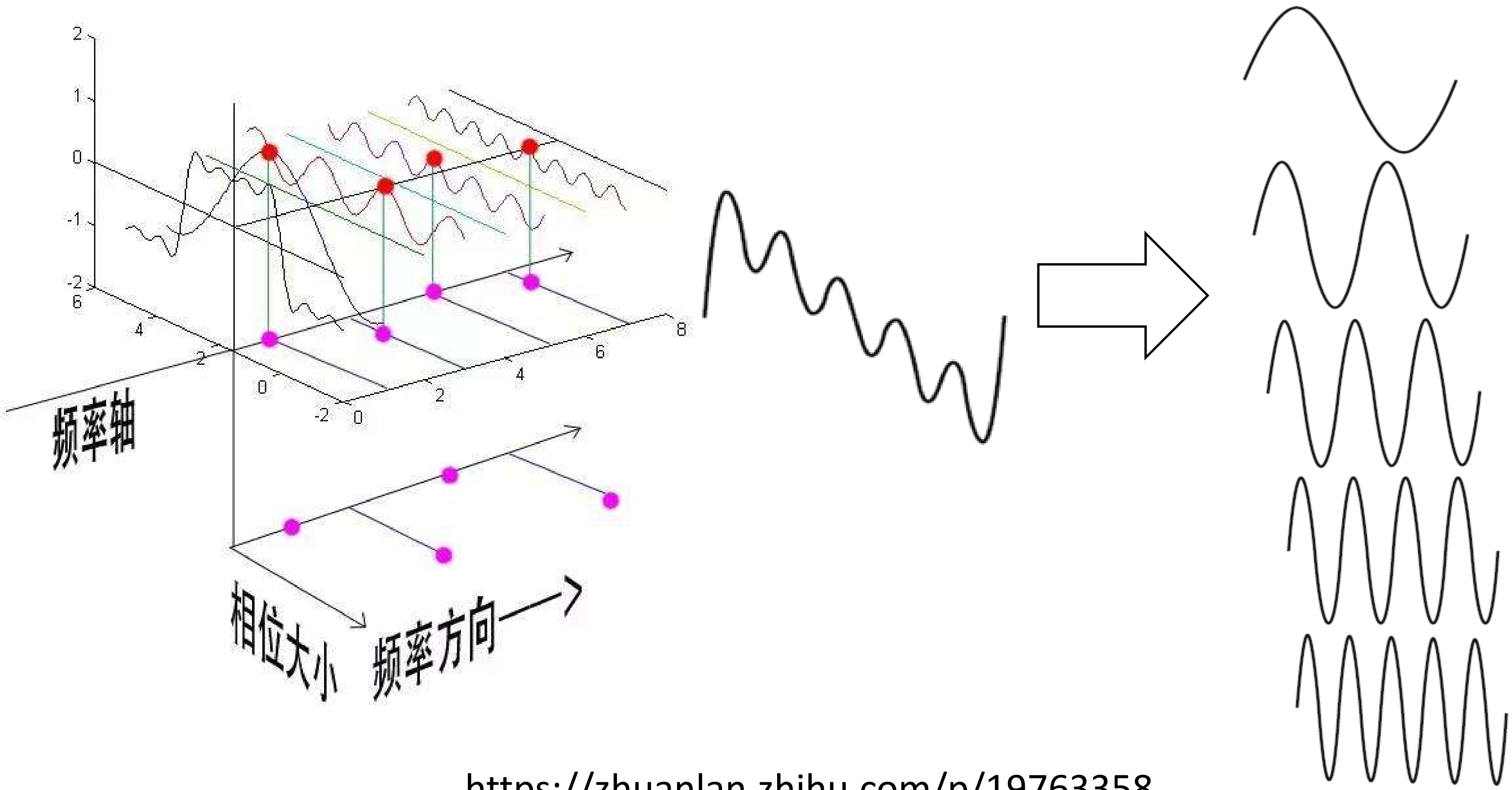
Digitization of Sound

- Digitization 声音数字化
  - Complex signal by superposing sinusoids 通过正弦信号叠加构造复杂信号



Digitization of Sound

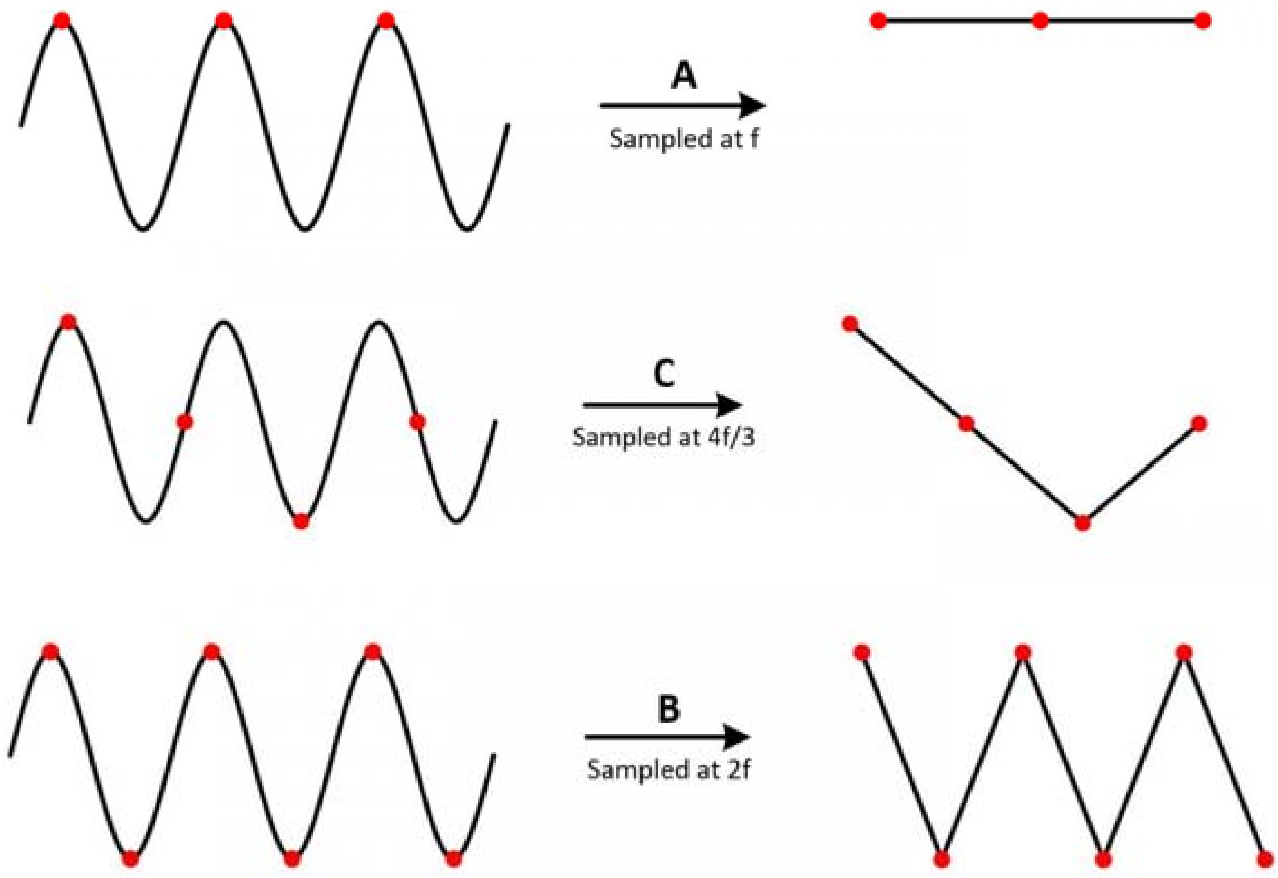
- Digitization 声音数字化
  - Fourier transformation: Extract information of each components 利用傅里叶变换提取声音信号成分



https://zhuanlan.zhihu.com/p/19763358

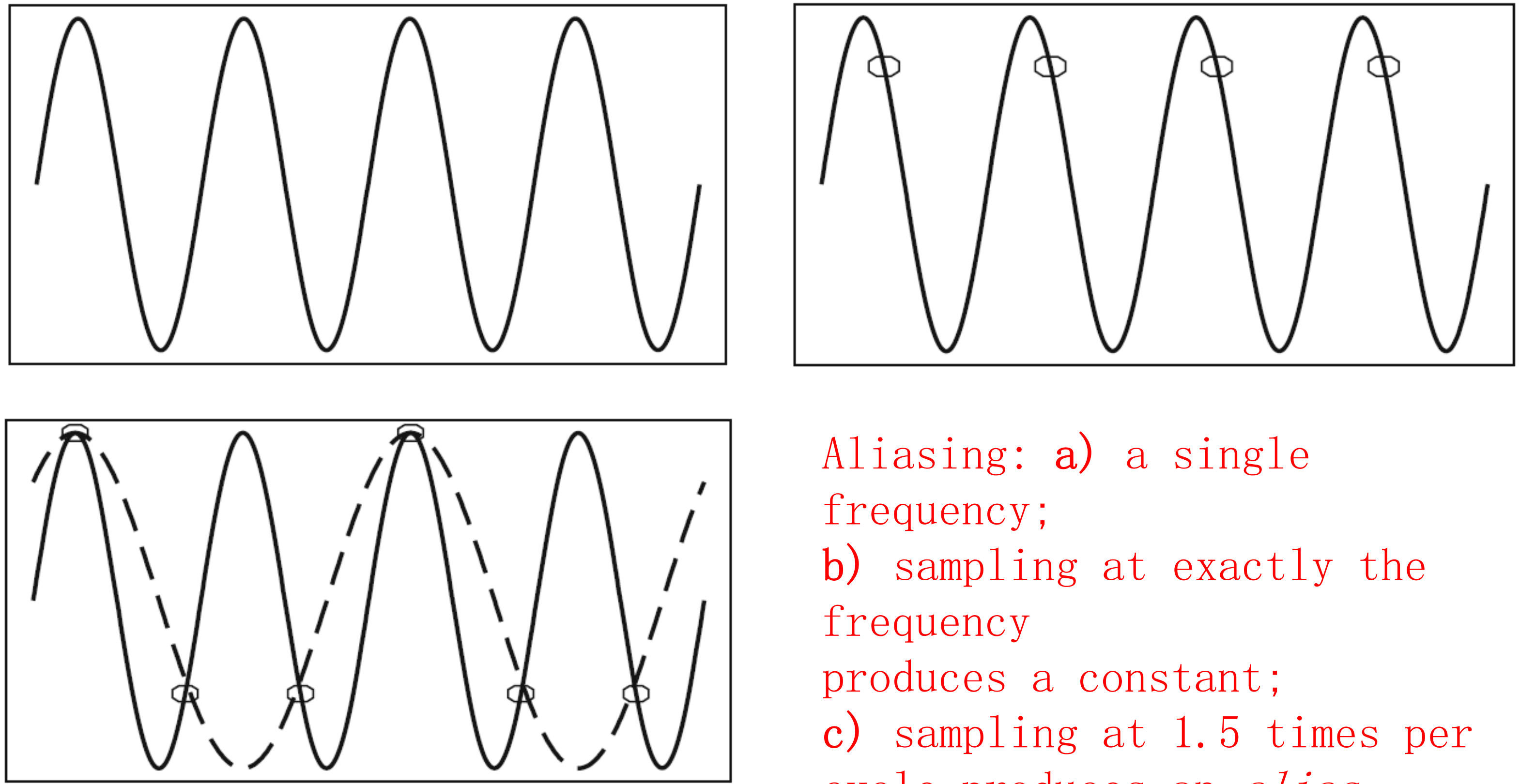
Digitization of Sound

- Nyquist Theorem 奈奎斯特理论
  - The Nyquist theorem states **how frequently** we must sample in time to be able to recover the original sound. 该理论告诉我们要对声音进行恢复应该在频率上如何进行采样



Digitization of Sound

- Nyquist Theorem 奈奎斯特理论
  - The Nyquist theorem states **how frequently** we must sample in time to be able to recover the original sound.该理论告诉我们要对声音进行恢复应该在频率上如何进行采样



Aliasing: a) a single frequency;
b) sampling at exactly the frequency produces a constant;
c) sampling at 1.5 times per cycle produces an *alias* frequency that is perceived

Digitization of Sound

- Nyquist Theorem 奈奎斯特理论
  - The Nyquist theorem states **how frequently** we must sample in time to be able to recover the original sound.该理论告诉我们要对声音进行恢复应该在频率上如何进行采样
  - For correct sampling we must use a sampling rate equal to at least *twice the maximum frequency* content in the signal. This rate is called the **Nyquist rate**. 至少使用信号最大频率两倍以上的采样率进行采样
  - **Nyquist Theorem**: If a signal is band-limited, i.e., there is a lower limit f_1 and an upper limit f_2 of frequency components in the signal, then the sampling rate should be at least $2(f_2 - f_1)$. 奈奎斯特理论
  - **Nyquist frequency**: half of the Nyquist rate. 奈奎斯特频率
  - The relationship among the Sampling Frequency, True Frequency, and the Alias Frequency is as follows:

$$(f_{alias} = f_{sampling} - f_{ture}, \ \textbf{for } \boldsymbol{f_{true} < f_{sampling} < 2 \times f_{ture}})$$

Digitization of Sound

- Signal to Noise Ratio (SNR)
  - The ratio of the power of the correct signal and the noise is called the *signal to noise ratio* (**SNR**) - a measure of the quality of the signal.
  - The SNR is usually measured in decibels (**dB**), where 1 dB is a tenth of a **bel**. The SNR value, in units of dB, is defined in terms of base-10 logarithms of squared voltages, as follows:

$$SNR = 10\log_{10}\frac{V_{signal}^2}{V_{noise}^2} = 20\log_{10}\frac{V_{signal}}{V_{noise}}$$

For example, if the signal voltage V_{signal} is 10 times the noise, then the SNR is $20\log_{10}(10)=20$dB.

Digitization of Sound

- Signal to Noise Ratio (SNR)
  - The usual levels of sound we hear around us are described in terms of decibels, as a ratio to the quietest sound we are capable of hearing.

Threshold of hearing	0
Rustle of leaves	10
Very quiet room	20
Average room	40
Conversation	60
Busy street	70
Loud radio	80
Train through station	90
Riveter	100
Threshold of discomfort	120
Threshold of pain	140
Damage to eardrum	160

Magnitude levels of common sounds, in decibels

Digitization of Sound

◆ Signal to Noise Ratio (SNR)

Digitization of Sound

- Signal to Quantization Noise Ratio (SQNR)

  - Aside from any noise that may have been present in the original analog signal, there is also an additional error that results from quantization.

  a) If voltages are actually in 0 to 1 but we have only 8 bits in which to store values, then effectively we force all continuous values of voltage into only 256 different values.

  b) This introduces a roundoff error. It is not really "noise". Nevertheless it is called **quantization noise** (or quantization error).

Digitization of Sound

- Signal to Quantization Noise Ratio (SQNR)
  - The quality of the quantization is characterized by the Signal to Quantization Noise Ratio (SQNR).

a) **Quantization noise**: the *difference* between the actual value of the analog signal, for the particular sampling time, and the nearest quantization interval value.

b) At most, this error can be as much as half of the interval.

c) For a quantization accuracy of N bits per sample, the SQNR can be simply expressed:

$$SQNR = 20\log_{10}\frac{V_{\text{signal}}}{V_{\text{quan\_noise}}} = 20\log_{10}\frac{2^{N-1}}{\frac{1}{2}}$$
$$= 20 \times N \times \log 2 = 6.02N(\text{dB})$$

- We map the maximum signal to $2^{N-1} - 1$ ($\approx 2^{N-1}$) and the most negative signal to -2^{N-1}.
- This is the Peak signal-to-noise ratio, PSQNR: peak signal and peak noise.

Digitization of Sound

- Signal to Quantization Noise Ratio (SQNR)

  - **6:02N** is the worst case. If the input signal is sinusoidal, the quantization error is statistically independent, and its magnitude is ***uniformly distributed*** between 0 and half of the interval, then it can be shown that the expression for the SQNR becomes:

$$SQNR = 6.02N + 1.76(dB)$$

The greater the N is , the larger the SQNR will be and the better quality of audio system can be provided.

Digitization of Sound

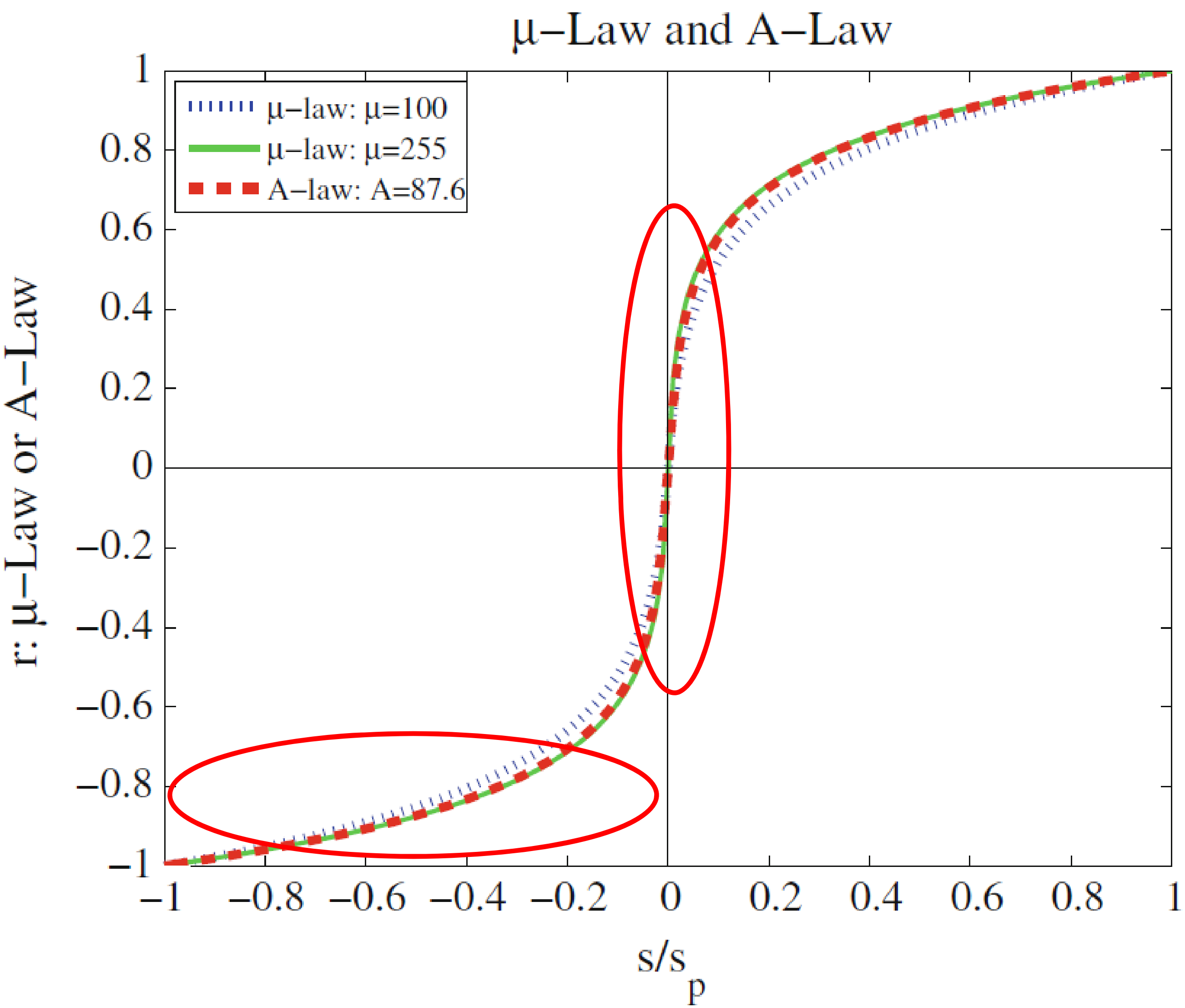
- Linear and Non-linear Quantization
  - **Linear format**: samples are typically stored as uniformly quantized values.
  - **Non-uniform quantization**: set up more finely-spaced levels where humans hear with the most acuity.
  - Weber's Law stated formally says that equally perceived differences have values proportional to absolute levels:

$$\Delta\text{Response} \propto \Delta\text{Stimulus}/\text{Stimulus}$$

  - for example, if we can feel an increase in weight from 10 to 11 pounds, then if instead we start at 20 pounds, it would take **22 pounds** for us

Digitization of Sound

- Linear and Non-linear Quantization



Digitization of Sound

- Linear and Non-linear Quantization
  - Nonlinear quantization works by first transforming an analog signal from the raw *s* space into the theoretical *r* space, and then uniformly quantizing the resulting values.
  - Such a law for audio is called ***μ*-law** encoding, (or **u-law**). A very similar rule, called **A-law**, is used in telephony in Europe.

μ-law:

$$r = \frac{\operatorname{sign}(s)}{\ln(1+\mu)} \ln\left\{1 + \mu\left|\frac{s}{s_p}\right|\right\}, \qquad \left|\frac{s}{s_p}\right| \le 1$$

A-law:

$$r = \begin{cases} \frac{A}{1+\ln A}\left(\frac{s}{s_p}\right), & \left|\frac{s}{s_p}\right| \le \frac{1}{A} \\ \frac{\operatorname{sign}(s)}{1+\ln A}\left[1 + \ln A\left|\frac{s}{s_p}\right|\right], & \frac{1}{A} \le \left|\frac{s}{s_p}\right| \le 1 \end{cases}$$

$$\text{where } \operatorname{sign}(s) = \begin{cases} 1 & \text{if } s > 0, \\ -1 & \text{otherwise} \end{cases}$$

Digitization of Sound

- Audio Filtering

  - Prior to sampling and AD conversion, the audio signal is also usually *filtered* to remove **unwanted frequencies**. The frequencies kept depend on the application:

a) For speech, typically from *50Hz to 10kHz* is retained, and other frequencies are blocked by the use of **a band-pass filter** that screens out lower and higher frequencies.

b) An audio music signal will typically contain from about *20Hz up to 20kHz*.

c) At the DA converter end, high frequencies may reappear in the output-- because of sampling and then quantization, smooth input signal is replaced by a series of step functions containing all possible frequencies.

d) So at the decoder side, a **lowpass** filter is used after the DA circuit.

Digitization of Sound

- Audio Quality *vs*. Data Rate
  - The uncompressed data rate increases as more bits are used for quantization. Stereo: double the bandwidth to transmit a digital audio signal.

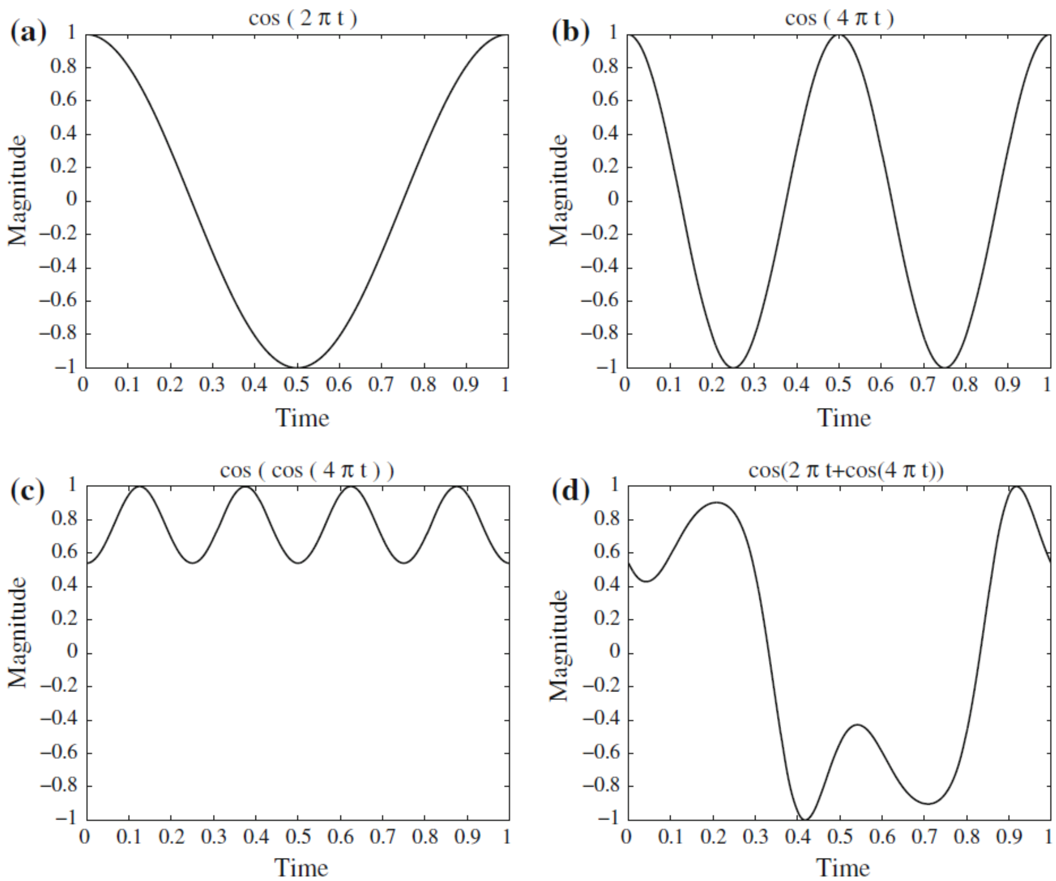
Quality	Sampling rate (kHz)	Bits per sample	Mono/ Stereo	Bitrate (if uncompressed) (kB/s)	Signal bandwidth (Hz)
Telephone	8	8	Mono	8	200–3,400
AM radio	11.025	8	Mono	11.0	100–5,500
FM radio	22.05	16	Stereo	88.2	20–11,000
CD	44.1	16	Stereo	176.4	5–20,000
DVD audio	192 (max)	24 (max)	Up to 6 channels	1,200.0 (max)	0–96,000 (max)

Data rate and bandwidth in sample audio applications

Digitization of Sound

◆ Synthetic Sounds

- **FM (Frequency Modulation):** one approach to generating synthetic sound:
- **Wave Table synthesis:** A more accurate way of generating sounds from digital signals. Also known, simply, as sampling.



Frequency Modulation. (a): A single frequency. (b): Twice the frequency. (c): Usually, FM is carried out using a sinusoid argument to a sinusoid. (d): A more complex form arises from a carrier frequency, $2\pi t$ and a modulating frequency $4\pi t$ cosine inside the sinusoid.

Outline of Lecture 05

- Digitization of Sound
  - Sound and Digitization
  - Nyquist Theorem
  - Signal to Noise Ratio (SNR)
  - Linear and Non-linear Quantization
  - Audio Quality and Synthetic Sounds
- MIDI: Musical Instrument Digital Interface
- Quantization and Transmission of Audio
- Experiments

MIDI: Musical Instrument Digital Interface

- **MIDI (乐器数字接口) Overview**
  - Use the sound card's defaults for sounds
  - Use a simple scripting language and hardware setup called **MIDI**.

a) MIDI is a **scripting language**--it codes "events" that stand for the production of sounds. E.g., a MIDI event might include values for the pitch of a single note, its duration, and its volume.

b) MIDI is a standard adopted by the **electronic music industry** for controlling devices, such as synthesizers and sound cards, that produce music.

c) Computers must have a special MIDI interface, but this is incorporated into most sound cards. The sound card must also have both D/A and A/D converters.

MIDI: Musical Instrument Digital Interface

◆ MIDI Concepts

- MIDI **channels** are used to separate messages.

a) There are 16 channels numbered from 0 to 15. The channel forms the last 4 bits (the least significant bits) of the message.

b) Usually a channel is associated with a particular instrument: e.g., channel 1 is the piano, channel 10 is the drums, etc.

c) Nevertheless, one can switch instruments midstream, if desired, and associate another instrument with any channel.

MIDI: Musical Instrument Digital Interface

- **System messages**

a) Several other types of messages, e.g. a general message for all instruments indicating a change in tuning(基调) or timing(节拍).

b) If the first 4 bits are all 1s, then the message is interpreted as a system common message.

- The way a synthetic musical instrument responds to a MIDI message is usually by simply ignoring any **play sound** message that is not for its channel.

a) If several messages are for its channel, then the instrument responds, provided it is **multi-voice**, i.e., can play more than a single note at once.

MIDI: Musical Instrument Digital Interface

- It is easy to confuse the term **voice(声部)** with the term **timbre(音色)** -- the latter is MIDI terminology for just what instrument that is trying to be emulated, e.g. a piano as opposed to a violin: it is the quality of the sound.

a) An instrument (or sound card) that is **multi-timbral** is one that is capable of playing many different sounds at the same time, e.g., piano, brass, drums, etc.

b) On the other hand, the term voice(声部), while sometimes used by musicians to mean the same thing as timbre(音色), is used in MIDI to mean every different timbre(音色) and pitch(音调) that the tone module(声音模块) can produce at the same time.

- Different timbres(音色) are produced digitally by using a **patch(编配程序)**-- the set of control settings that dene a particular timbre(音色). Patches(编配程序) are often organized into databases, called **banks(音色库)**.

MIDI: Musical Instrument Digital Interface

- **General MIDI:**
  - A standard mapping specifying what instruments (what patches) will be associated with what channels.

a) In General MIDI, channel 10 is reserved for percussion instruments(敲击乐器),

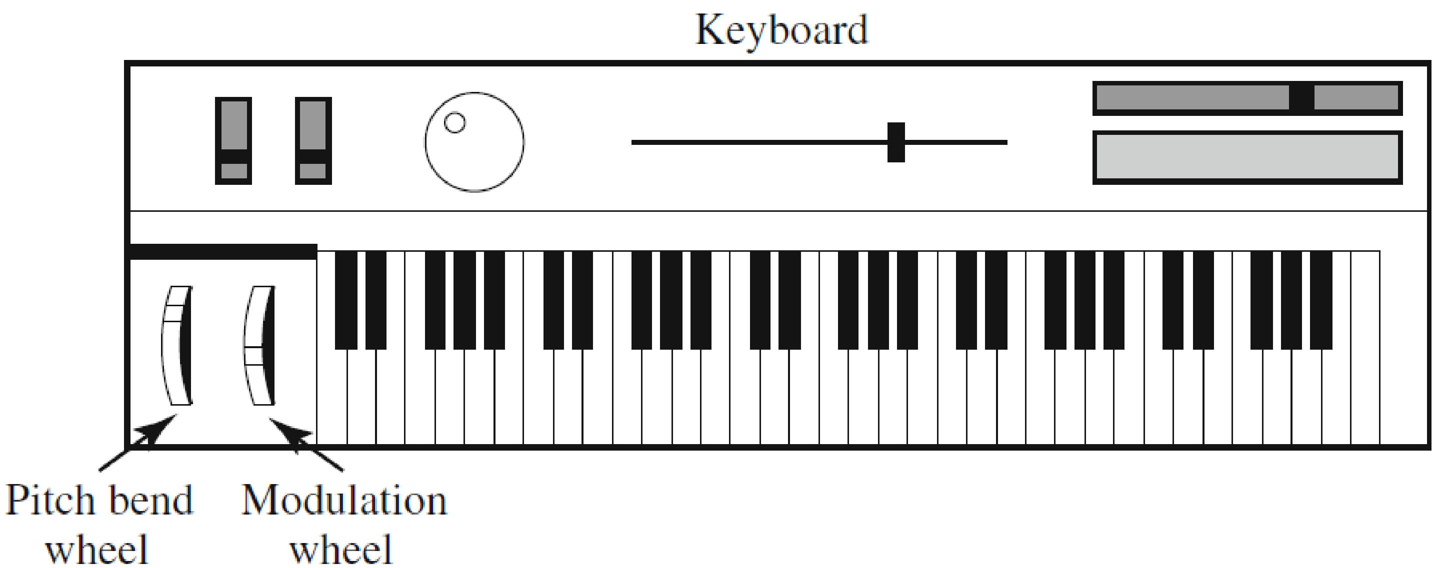
b) For most instruments, a typical message might be a Note On message (meaning, e.g., a keypress and release), consisting of what channel(通道编号), what pitch(音调), and what “velocity” (i.e., volume)(音量).

c) For percussion instruments, however, the pitch(音调) data means which kind of drum.

d) A Note On message consists of “status” byte--which channel, what pitch--followed by two data bytes. It is followed by a Note Off message, which also has a pitch(音调数据) (which note to turn off) and a velocity(音量数据) (often set to zero).

MIDI: Musical Instrument Digital Interface

- **Hardware Aspects of MIDI:**
  - The MIDI hardware setup consists of a 31.25 kbps serial connection. Usually, MIDI-capable units are either Input devices or Output devices, not both.
  - A traditional synthesizer is shown in Figure below:



A MIDI synthesizer

For more information, refers to
https://baike.baidu.com/item/MIDI/217824?fr=aladdin

MIDI: Musical Instrument Digital Interface

◆ MIDI to WAV Conversion

- Some programs, such as early versions of Premiere, cannot include *.mid* files-- instead, they insist on *.wav* format files.

a) Various shareware programs exist for approximating a reasonable conversion between MIDI and WAV formats.

b) These programs essentially consist of large lookup files that try to substitute pre-defined or shifted WAV output for MIDI messages, with inconsistent success.

Outline of Lecture 05

- Digitization of Sound
  - Sound and Digitization
  - Nyquist Theorem
  - Signal to Noise Ratio (SNR)
  - Linear and Non-linear Quantization
  - Audio Quality and Synthetic Sounds
- MIDI: Musical Instrument Digital Interface
- Quantization and Transmission of Audio
- Experiments

Quantization and Transmission of Audio

◆ Coding of Audio

- Quantization and transformation of data are collectively known as **coding** of the data.

a) For audio, the μ-law technique for companding audio signals(扩展压缩音频信号) is usually combined with an algorithm that exploits the temporal redundancy present in audio signals.

b) *Differences* in signals between the present and a past time can reduce the size of signal values and also concentrate the histogram of pixel values (differences, now) into a much smaller range.

Quantization and Transmission of Audio

◆ Coding of Audio

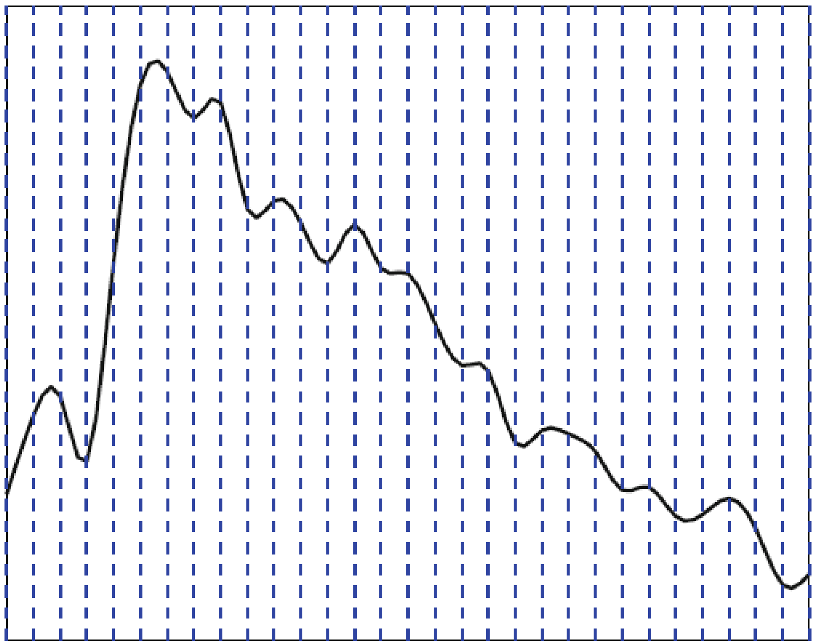
a) The result of reducing the variance of values is that lossless compression methods produce a bitstream with shorter bit lengths for more likely values.

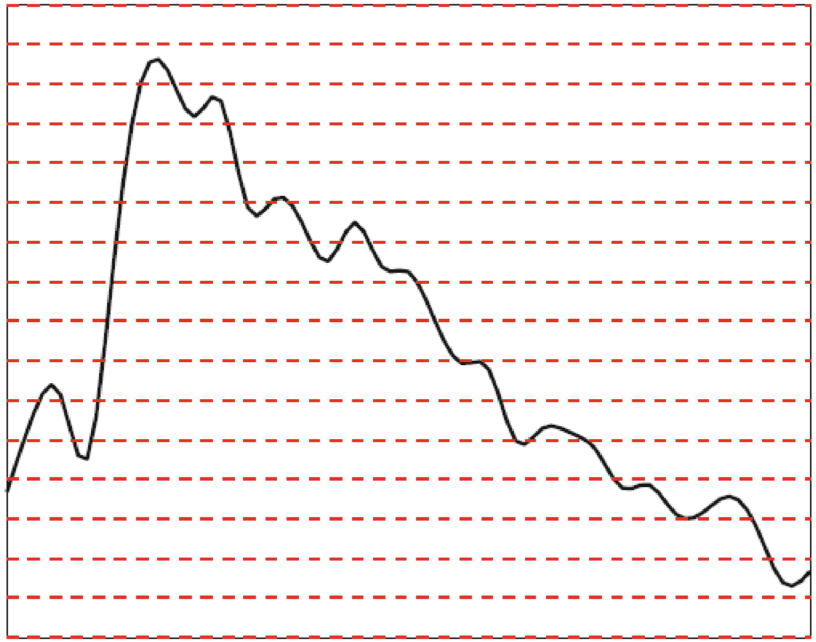
- In general, producing quantized sampled output for audio is called **PCM** (Pulse Code Modulation). The differences version is called **DPCM** (and a crude but efficient variant is called **DM**). The adaptive version is called **ADPCM**.

Quantization and Transmission of Audio

◆ Pulse Code Modulation

- The basic techniques for creating digital signals from analog signals are **sampling** and **quantization**.
- Quantization consists of selecting breakpoints in magnitude, and then re-mapping any value within an interval to one of the representative output levels.





Sampling and Quantization.

Quantization and Transmission of Audio

◆ Pulse Code Modulation

a) The set of interval boundaries are called **decision boundaries**, and the representative values are called **reconstruction levels(重构层)**.

b) The boundaries for quantizer input intervals that will all be mapped into the same output level form a **coder mapping**.

c) The representative values that are the output values from a quantizer are a **decoder mapping**.

d) Finally, we may wish to **compress** the data, by assigning a bit stream that uses fewer bits for the most prevalent signal values

Quantization and Transmission of Audio

◆ Pulse Code Modulation

- Every compression scheme has three stages:

a) The input data is **transformed** to a new representation that is easier or more efficient to compress.

b) We may introduce **loss** of information. *Quantization* is the main lossy step -- we use a limited number of reconstruction levels, fewer than in the original signal.

c) Coding. Assign a codeword (thus forming a binary bitstream) to each output level or symbol. This could be axed-length code, or a variable length code such as Huffman coding (Chapter 7).

d) PCM leads to Lossless Predictive Coding and the **DPCM** scheme -- *differential coding*. As well, the adaptive version, **ADPCM**, which can provide better compression.

Quantization and Transmission of Audio

◆ PCM in Speech Compression

- Assuming a bandwidth for speech from about 50 Hz to about 10 kHz, the Nyquist rate would dictate a sampling rate of 20 kHz.

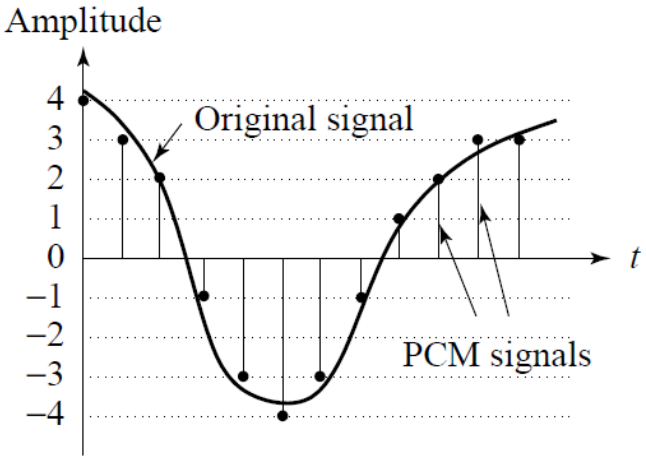
a) Using uniform quantization without companding(压缩扩展), the minimum sample size we could get away with would likely be about *12 bits*. Hence for mono speech transmission the bit-rate would be **240 kbps**.

b) With companding(压缩扩展), we can reduce the sample size down to about *8 bits* with the same perceived level of quality, and thus reduce the bit-rate to **160 kbps**.

c) However, the standard approach to telephony assumes that the highest-frequency audio signal is only *about 4 kHz*. Therefore the sampling rate is only 8 kHz, and the companded bit-rate thus reduces this to **64 kbps**.

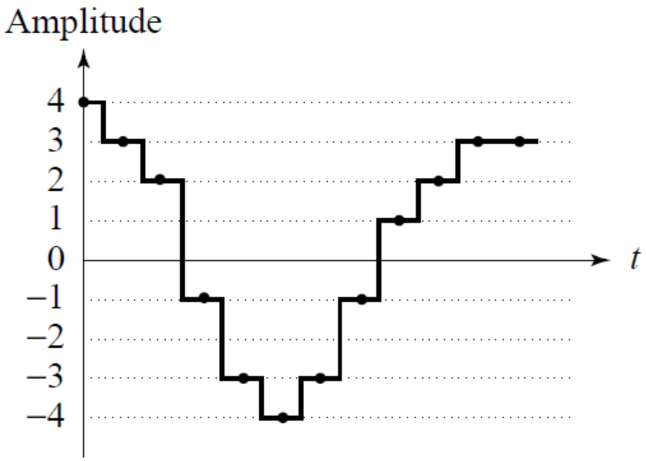
Quantization and Transmission of Audio

◆ PCM in Speech Compression

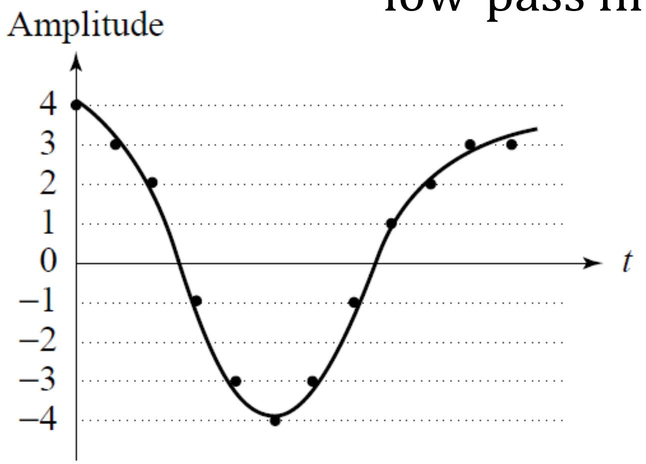


(a)

Pulse Code Modulation (PCM).
(a) Original analog signal and its corresponding PCM signals.
(b) Decoded staircase signal.
(c) Reconstructed signal after low-pass filtering.



(b)

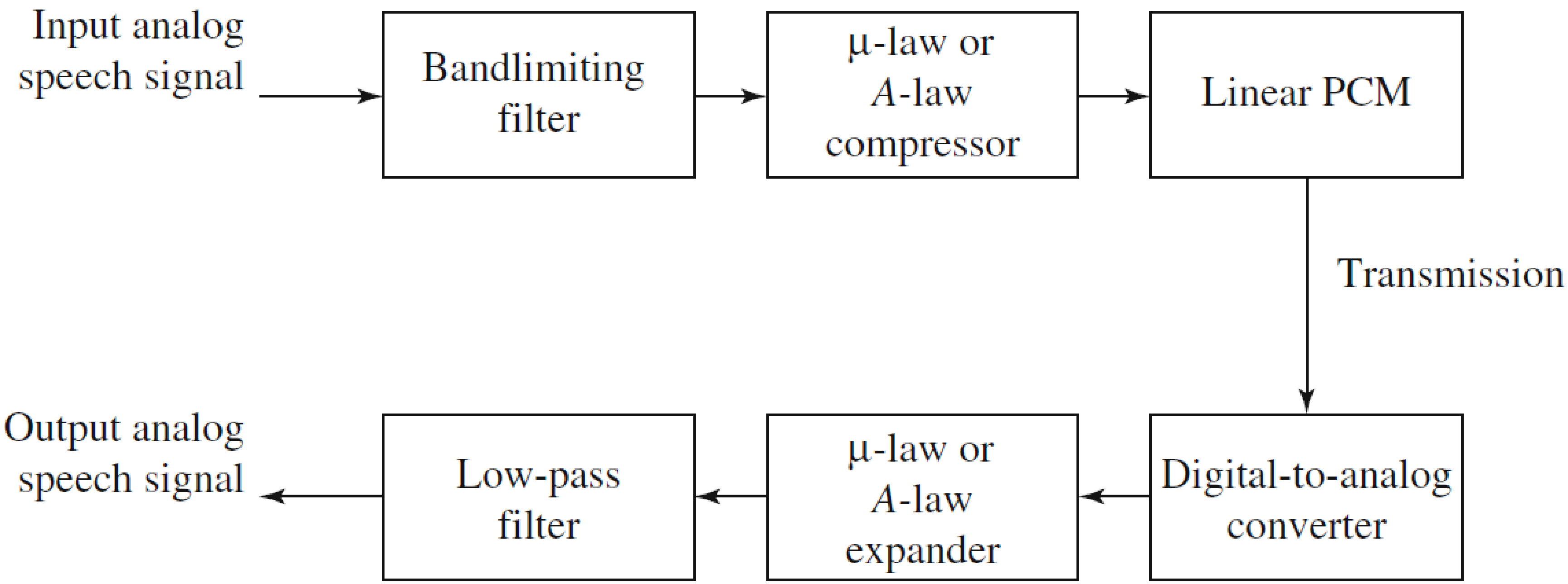


(c)

Quantization and Transmission of Audio

◆ PCM in Speech Compression

- The complete scheme for encoding and decoding telephony signals is shown as a schematic in Figure below.



PCM signal encoding and decoding

Quantization and Transmission of Audio

◆ Differential Coding of Audio

- Audio is often stored not in simple PCM but instead in a form that exploits *differences* -- which are generally smaller numbers, so offer the possibility of using fewer bits to store.

a) If a time-dependent signal has some consistency over time ("temporal redundancy"), the difference signal, subtracting the current sample from the previous one, will have a more *peaked histogram*, with a maximum around zero.

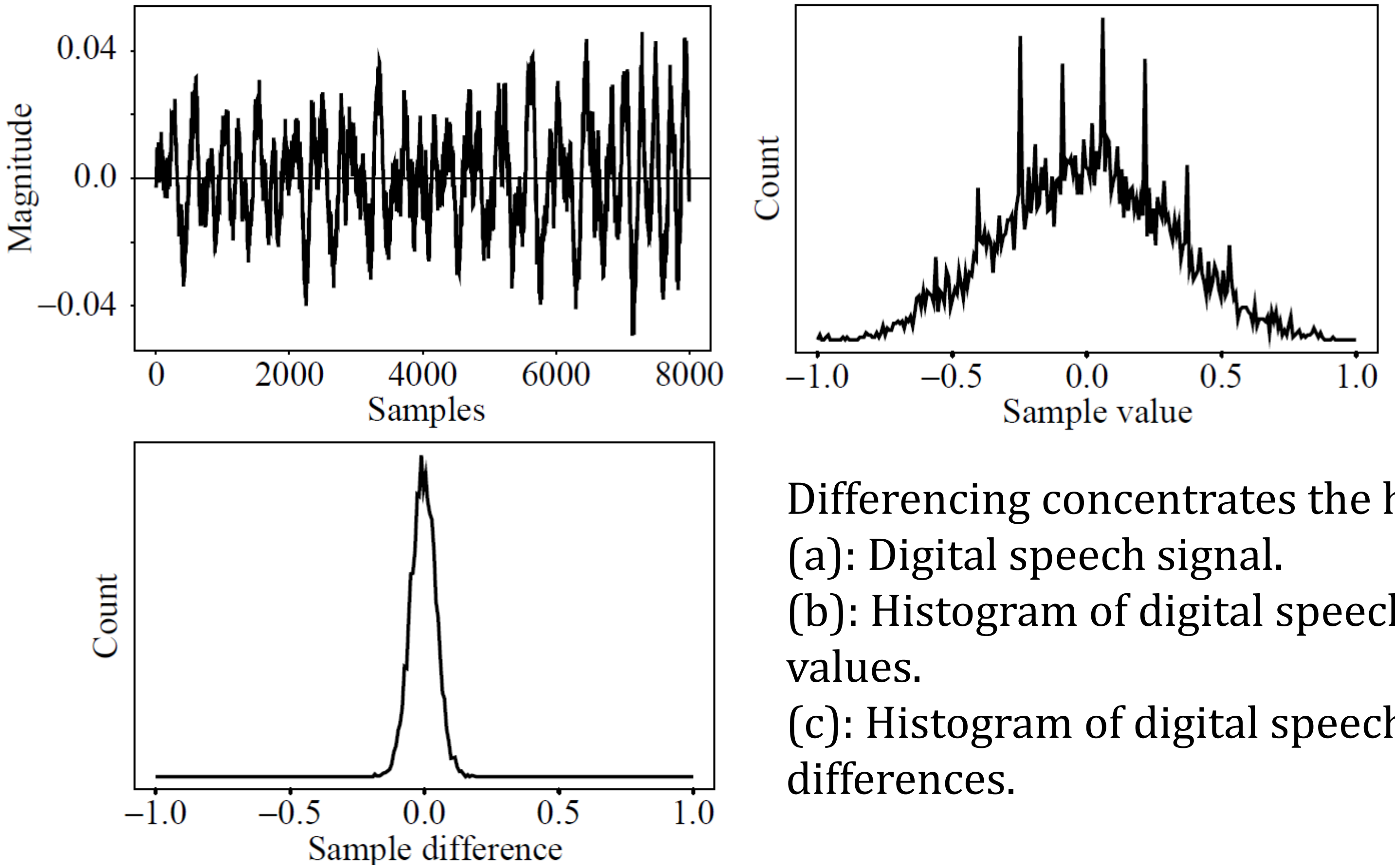
b) For example, as an extreme case the histogram for a linear ramp signal that has constant slope is flat, whereas the histogram for the derivative of the signal (i.e., the differences, from sampling point to sampling point) consists of a spike at the slope value.

c) So if we then go on to assign bit-string codewords to differences, we can assign *short codes* to prevalent values and *long codewords* to rarely occurring ones.

Quantization and Transmission of Audio

◆ Lossless Predictive Coding

- The idea of forming differences is to make the histogram of sample values more peaked.
- Coding scheme: assigns short codewords to frequently occurring symbols.



Differencing concentrates the histogram.
(a): Digital speech signal.
(b): Histogram of digital speech signal values.
(c): Histogram of digital speech signal differences.

Quantization and Transmission of Audio

◆ Lossless Predictive Coding

- **Predictive coding**: simply means transmitting differences -- predict the next sample as being equal to the current sample; send not the sample itself but the difference between previous and next.

a) Predictive coding consists of finding differences, and transmitting these using a PCM system.

b) Note that differences of integers will be integers.

Input signal: f_n

Predict values: $\widehat{f_n} = f_{n-1}$

Error:$e_n = f_n - \widehat{f_n}$

c) Some function of a few of the previous values provides a better prediction.

$$\widehat{f_n} = \sum_{k=1}^{2\sim 4} a_{n-k} f_{n-k}$$

Quantization and Transmission of Audio

◆ Lossless Predictive Coding

- One problem: suppose our integer sample values are in the range 0..255. Then differences could be as much as -255..255 -- we've increased our dynamic range (ratio of maximum to minimum) by a factor of two -- need more bits to transmit some differences.

a) A clever solution for this: define two new codes, denoted ***SU*** and ***SD***, standing for *Shift-Up* and *Shift-Down*.

b) Differences which lie in the limited range can be coded as is, but with the extra two values for ***SU (32)***, ***SD (32)***, a value outside the range.

c) For example, in range -15~16, the difference value 100 is transmitted as: ***SU***, ***SU***, ***SU***, 4 (4 is in the range -15~16), where (the codes for) ***SU*** and for 4 are what are transmitted (or stored).

问题：-65和80的编码分别是多少？ -65: SD, SD, -1； 80：SU,SU, 16

Quantization and Transmission of Audio

◆ Lossless Predictive Coding

- Simple example: the sequence f_1, f_2, f_3, f_4, f_5= 21, 22, 27, 25, 22.

$$\widehat{f_n} = \left\lfloor \frac{1}{2}(f_{n-1} + f_{n-2}) \right\rfloor$$
$$e_n = f_n - \widehat{f_n}$$

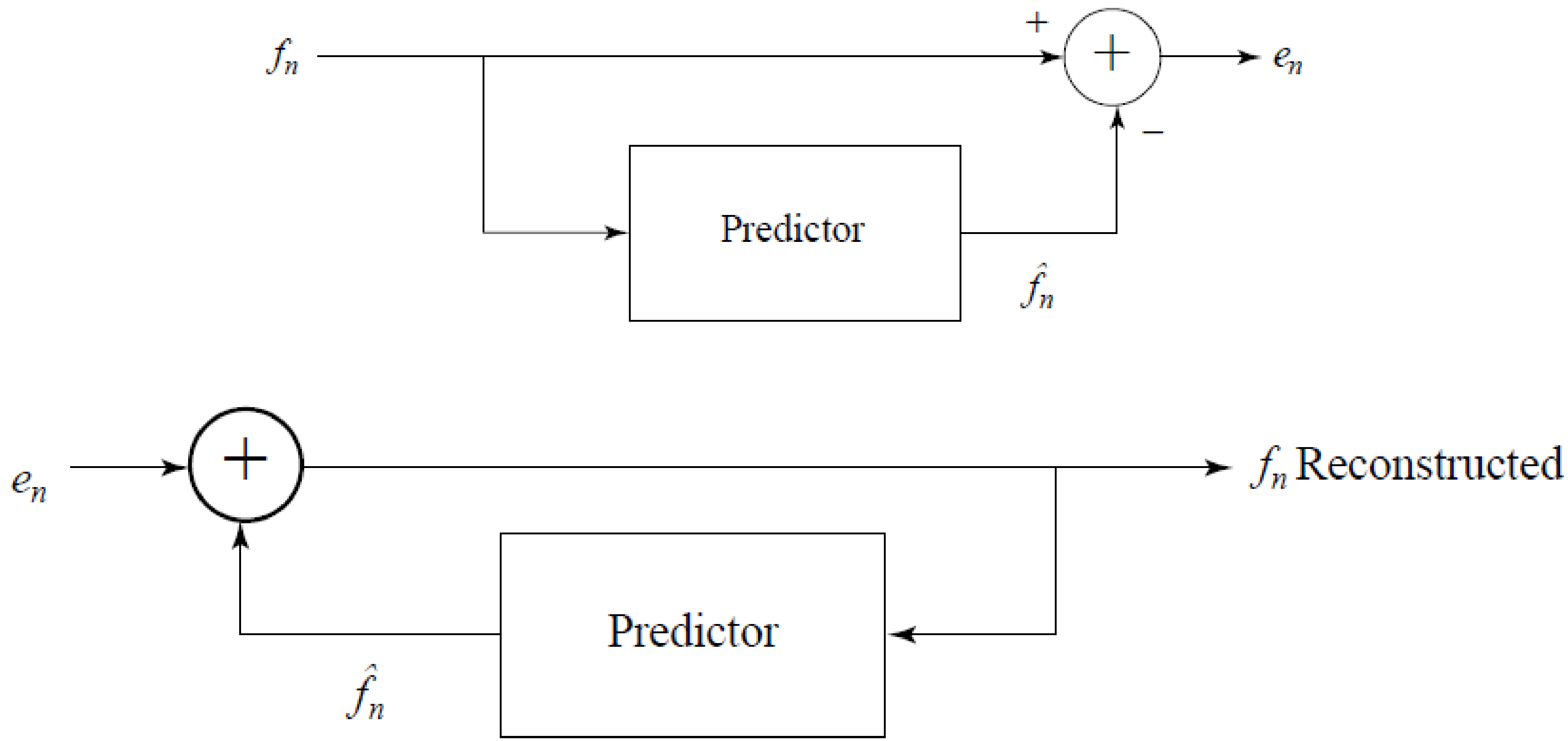
$$\hat{f}_2 = 21, \quad e_2 = 22 - 21 = 1$$

$$\hat{f}_3 = \lfloor \frac{1}{2}(f_2 + f_1) \rfloor = \lfloor \frac{1}{2}(22 + 21) \rfloor = 21$$
$$e_3 = 27 - 21 = 6$$
$$\hat{f}_4 = \lfloor \frac{1}{2}(f_3 + f_2) \rfloor = \lfloor \frac{1}{2}(27 + 22) \rfloor = 24$$
$$e_4 = 25 - 24 = 1$$
$$\hat{f}_5 = \lfloor \frac{1}{2}(f_4 + f_3) \rfloor = \lfloor \frac{1}{2}(25 + 27) \rfloor = 26$$
$$e_5 = 22 - 26 = -4$$

The error does center around zero, we see, and coding will be efficient.

Quantization and Transmission of Audio

◆ **Lossless Predictive Coding**



Schematic diagram for Predictive Coding encoder and decoder.

Quantization and Transmission of Audio

◆ DPCM

- Differential PCM is exactly the same as Predictive Coding, except that it incorporates a **quantizer** step.

a) One scheme for analytically determining the best set of quantizer steps, for a non-uniform quantizer, is the **Lloyd-Max** quantizer, which is based on a least-squares minimization of the error term.

b) Our nomenclature: **signal values**: f_n - the original signal, $\widehat{f}_n$ - the predicted signal, and $\widetilde{f}_n$ - the quantized, reconstructed signal; **Error values**: e_n - an error by subtracting the prediction from the actual signal, $\tilde{e}_n$ - an error for quantizing the original error.

Quantization and Transmission of Audio

◆ DPCM

- The set of equations that describe DPCM are as follows

$$\hat{f}_n = \text{function\_of}\ (\tilde{f}_{n-1}, \tilde{f}_{n-2}, \tilde{f}_{n-3}, \ldots)$$

$$e_n = f_n - \hat{f}_n$$

$$\tilde{e}_n = Q[e_n]$$

$$\text{transmit codeword}(\tilde{e}_n)$$

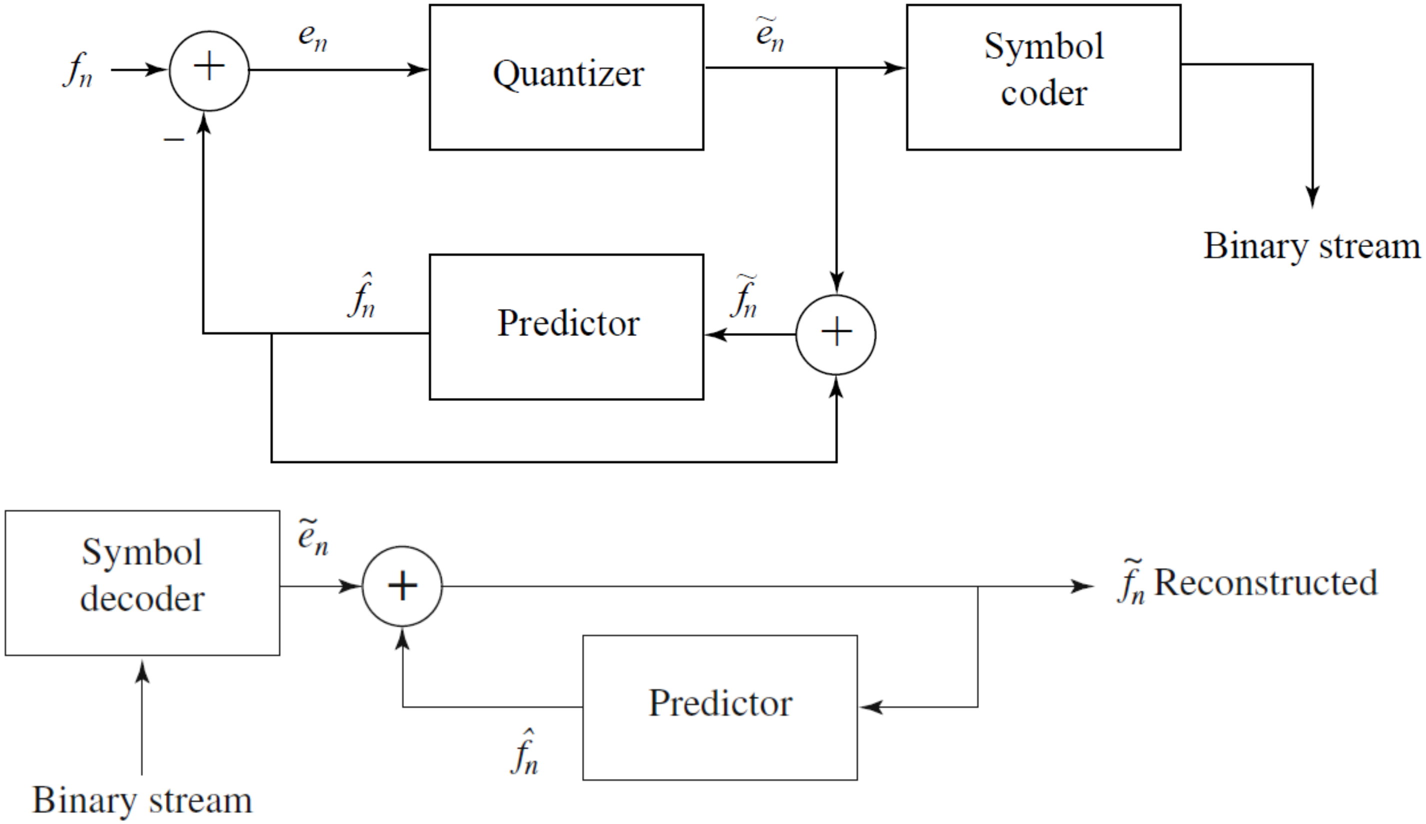
$$\text{reconstruct: } \tilde{f}_n = \hat{f}_n + \tilde{e}_n$$

- *codewords* for quantized error values $\tilde{e}_n$ are produced using entropy coding, e.g. Huffman coding (Chapter 7).

Quantization and Transmission of Audio

◆ DPCM

- Schematic diagram for DPCM:



- Notice that the quantization noise, $f_n - \widetilde{f_n}$, is equal to the quantization effect on the error term, $e_n - \tilde{e}_n$.

Because , $e_n = f_n - \widehat{f_n}$, $\tilde{e}_n = \widetilde{f_n} - \widehat{f_n}$.

Quantization and Transmission of Audio

◆ DPCM

- For example, suppose we adopt the particular predictor below:

$$\hat{f}_n = \text{trunc}\left[\left(\tilde{f}_{n-1} + \tilde{f}_{n-2}\right)/2\right]$$

- so that $e_n = f_n - \tilde{f}_n$ is an integer. As well, use the quantization scheme:

$$\tilde{e}_n = Q[e_n] = 16 * \text{trunc}\left[(255 + e_n)/16\right] - 256 + 8$$
$$\tilde{f}_n = \hat{f}_n + \tilde{e}_n$$

- First, we note that the error is in the range −255~255, i.e., there are **511 possible levels** for the error term. The quantizer simply divides the error range into 32 patches of about 16 levels each. It also makes the representative reconstructed value for each patch equal to the midway point for each group of 16 levels.

Quantization and Transmission of Audio

◆ DPCM

- Table below gives output values for any of the input codes: 4-bit codes are mapped to 32 reconstruction levels in a staircase fashion.

e_n in range	Quantized to value
$-255 .. -240$	-248
$-239 .. -224$	-232
$\vdots$	$\vdots$
$-31 .. -16$	-24
$-15 .. 0$	-8
$1 .. 16$	8
$17 .. 32$	24
$\vdots$	$\vdots$
$225 .. 240$	232
$241 .. 255$	248

Quantization and Transmission of Audio

◆ DPCM

- As an example stream of signal values, consider the set of values:

f_1	f_2	f_3	f_4	f_5
130	150	140	200	230

- Prepend extra values f = 130 to replicate the first value, f_1. Initialize with quantized error $\tilde{e}_1$ = 0, so that the first reconstructed value is exact: $\tilde{f}_1$ = 130. Then the rest of the values calculated are as follows (with prepended values in a box):

$$\hat{f}_1 = f$$

$$\hat{f}_n = trunc\left[\frac{\tilde{f}_{n-1} + \tilde{f}_{n-2}}{2}\right]$$

$$e_n = f_n - \hat{f}_n$$

$$\tilde{e}_n = 16 * \text{trunc}\left[\frac{255 + e_n}{16}\right] - 256 + 8$$

$$\tilde{f}_n = \hat{f}_n + \tilde{e}_n$$

$$\hat{f} = \boxed{130},\ 130,\ 142,\ 144,\ 167$$

$$e = \boxed{0},\ 20,\ -2,\ 56,\ 63$$

$$\tilde{e} = \boxed{0},\ 24,\ -8,\ 56,\ 56$$

$$\tilde{f} = \boxed{130},\ 154,\ 134,\ 200,\ 223$$

Quantization and Transmission of Audio

◆ DM

- **DM (Delta Modulation)**: simplified version of DPCM. Often used as a quick AD converter.
- Uniform-Delta DM: use only a single quantized error value, either positive or negative.

a) a 1-bit coder. Produces coded output that follows the original signal in a staircase fashion. The set of equations is:

$$\hat{f}_{\mathrm{n}} = \tilde{f}_{\mathrm{n}-1}$$

$$e_{\mathrm{n}} = f_{\mathrm{n}} - \hat{f}_{\mathrm{n}} = f_{\mathrm{n}} - \tilde{f}_{\mathrm{n}-1}$$

$$\tilde{e}_{\mathrm{n}} = \begin{cases} +k \text{ if } e_{\mathrm{n}} > 0, & \text{where } k \text{ is a constant} \\ -k \text{ otherwise}, \end{cases}$$

$$\tilde{f}_{\mathrm{n}} = \hat{f}_{n} + \tilde{e}_{\mathrm{n}}$$

Note that the prediction simply involves a **delay**.

Quantization and Transmission of Audio

- **DM**

$$\hat{f}_n = \tilde{f}_{n-1}$$
$$e_n = f_n - \hat{f}_n = f_n - \tilde{f}_{n-1}$$
$$\tilde{e}_n = \begin{cases} +k \text{ if } e_n > 0, & \text{where } k \text{ is a constant} \\ -k \text{ otherwise,} \end{cases}$$
$$\tilde{f}_n = \hat{f}_n + \tilde{e}_n$$

- Consider actual numbers: Suppose signal values are

f_1	f_2	f_3	f_4
10	11	13	15

- Define ne an exact reconstructed value $\tilde{f}_1 = f_1 = 10$.
- use step value $k = 4$:

$$\begin{array}{llll} \hat{f}_2 = 10, & e_2 = 11 - 10 = 1, & \tilde{e}_2 = 4, & \tilde{f}_2 = 10 + 4 = 14 \\ \hat{f}_3 = 14, & e_3 = 13 - 14 = -1, & \tilde{e}_3 = -4, & \tilde{f}_3 = 14 - 4 = 10 \\ \hat{f}_4 = 10, & e_4 = 15 - 10 = 5, & \tilde{e}_4 = 4, & \tilde{f}_4 = 10 + 4 = 14 \end{array}$$

- The reconstructed set of values 10, 14, 10, 14 is close to the correct set 10, 11, 13, 15.
- However, DM copes less well with **rapidly changing signals**. One approach to mitigating this problem is to simply **increase the sampling**, perhaps to many times the Nyquist rate.

Quantization and Transmission of Audio

◆ DM

- **Adaptive DM:** If the slope of the actual signal curve is high, the staircase approximation cannot keep up. For a steep curve, should *change the step size k* adaptively.
- One scheme for analytically *determining the best set of quantizer steps*, for a non-uniform quantizer, is Lloyd-Max.

Quantization and Transmission of Audio

◆ ADPCM

- **ADPCM** (Adaptive DPCM) takes the idea of adapting the coder to suit the input much farther. The two pieces that make up a DPCM coder: the *quantizer* and the *predictor*.
- In Adaptive DM, adapt the quantizer step size to suit the input. In DPCM, we can change the step size as well as decision boundaries, using a non-uniform quantizer. We can carry this out in two ways:

a) **Forward adaptive quantization**: use the properties of the *input signal*.

b) **Backward adaptive quantization**: use the properties of the *quantized output*. If quantized errors become too large, we should change the non-uniform quantizer.

Quantization and Transmission of Audio

◆ ADPCM

- We can also adapt the predictor, again using forward or backward adaptation. Making the predictor coe-fficients adaptive is called *Adaptive Predictive Coding* (APC):

a) Recall that the predictor is usually taken to be a linear function of previous reconstructed quantized values, $\tilde{f}_n$

b) The number of previous values used is called the "order" of the predictor. For example, if we use M previous values, we need M coefficients a_i; $i = 1,\ldots,M$ in a predictor

$$\hat{f}_n = \sum_{i=1}^{M} a_i \tilde{f}_{n-i}$$

However we can get into a difficult situation if we try to change the prediction coefficients, that multiply previous quantized values, because that makes a complicated set of equations to solve for these coefficients:

Quantization and Transmission of Audio

◆ ADPCM

a) Suppose we decide to use a least-squares approach to solving a minimization trying to find the best values of the a_i:

$$min \sum_{n=1}^{N} (f_n - \hat{f}_n)^2$$

b) Here we would sum over a large number of samples f_n, for the current patch of speech, say. Because $\widehat{f_n}$ depends on the quantization we have a difficult problem to solve. As well, we should really be changing the fineness of the quantization at the same time, to suit the signal's changing nature; this makes things problematical.

Quantization and Transmission of Audio

◆ ADPCM

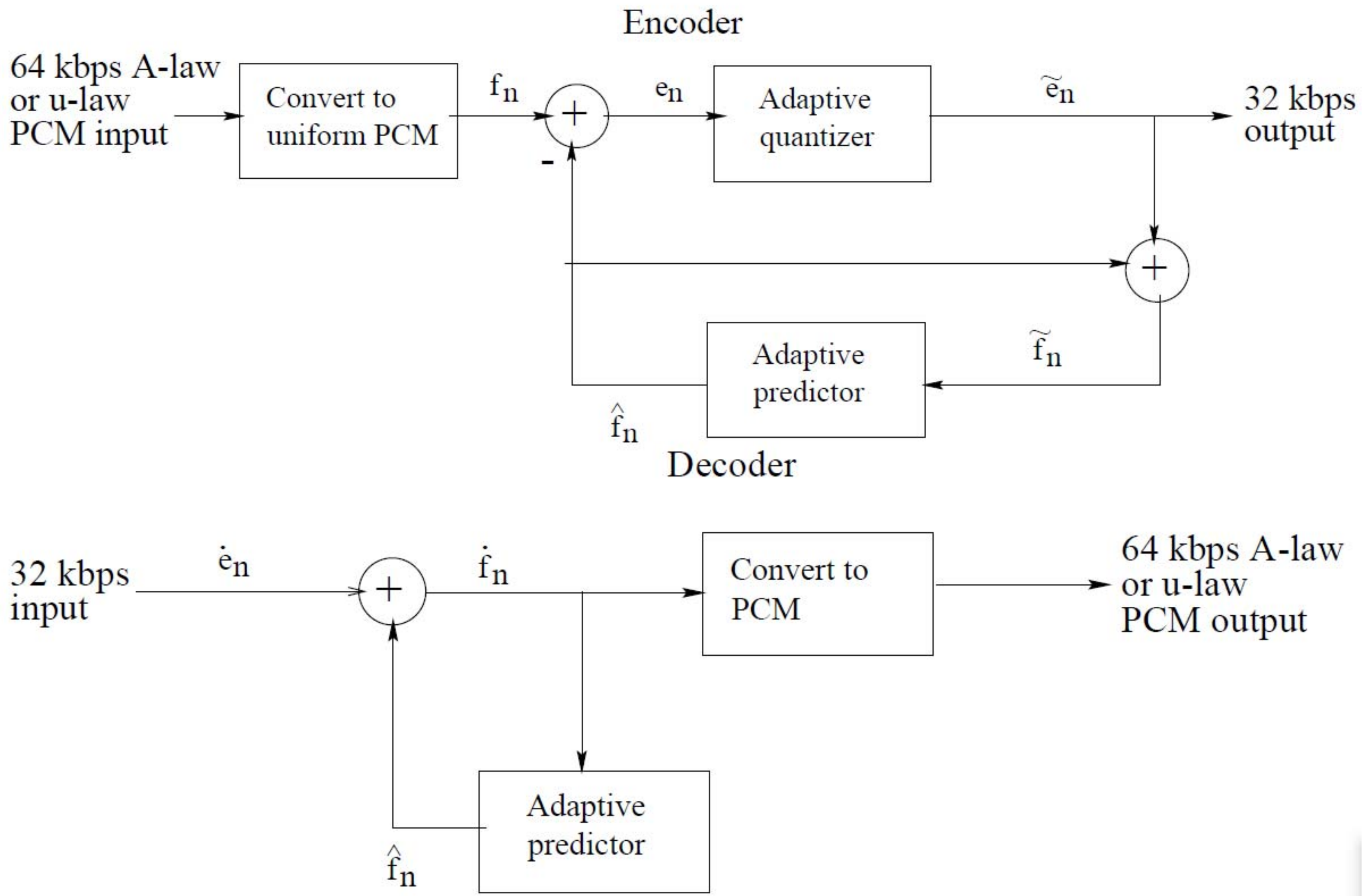
- Instead, one usually resorts to solving the simpler problem that results from using not $\tilde{f}_n$ in the prediction, but instead simply the signal f_n itself. Explicitly writing in terms of the coefficients a_i, we wish to solve:

$$min \sum_{n=1}^{N} \left(f_n - \sum_{i=1}^{M} a_i f_{n-i} \right)^2$$

- Differentiation with respect to each of the a_i, and setting to zero, produces a linear system of M equations that is easy to solve. (The set of equations is called the Wiener-Hopf equations.)

Quantization and Transmission of Audio

◆ ADPCM



Schematic diagram for ADPCM encoder and decoder

Outline of Lecture 05

- Digitization of Sound
  - Sound and Digitization
  - Nyquist Theorem
  - Signal to Noise Ratio (SNR)
  - Linear and Non-linear Quantization
  - Audio Quality and Synthetic Sounds
- MIDI: Musical Instrument Digital Interface
- Quantization and Transmission of Audio
- Experiments

Experiments

- Signal-to-Quantization-Noise Ratio (SQNR)
  - *ch06_sqnr_sinusoid.m*
- Pulse Code Modulation (PCM)
  - *ch06_PCM.m*
- Differential Pulse Code Modulation (DPCM)
  - *ch06_DPCM.m*

Class Assignments

1、什么情况下会出现假频？如真实频率为22.05kHz，采样频率33.075kHz，则假频为多少？

2、某电脑上有一块16位的声卡，这里的16位是指什么？其信号量化噪声比SQNR是多少？

3、编写代码实现声音信号的均匀增量调制（DM, Delta Modulation）.