technische universität
dortmund

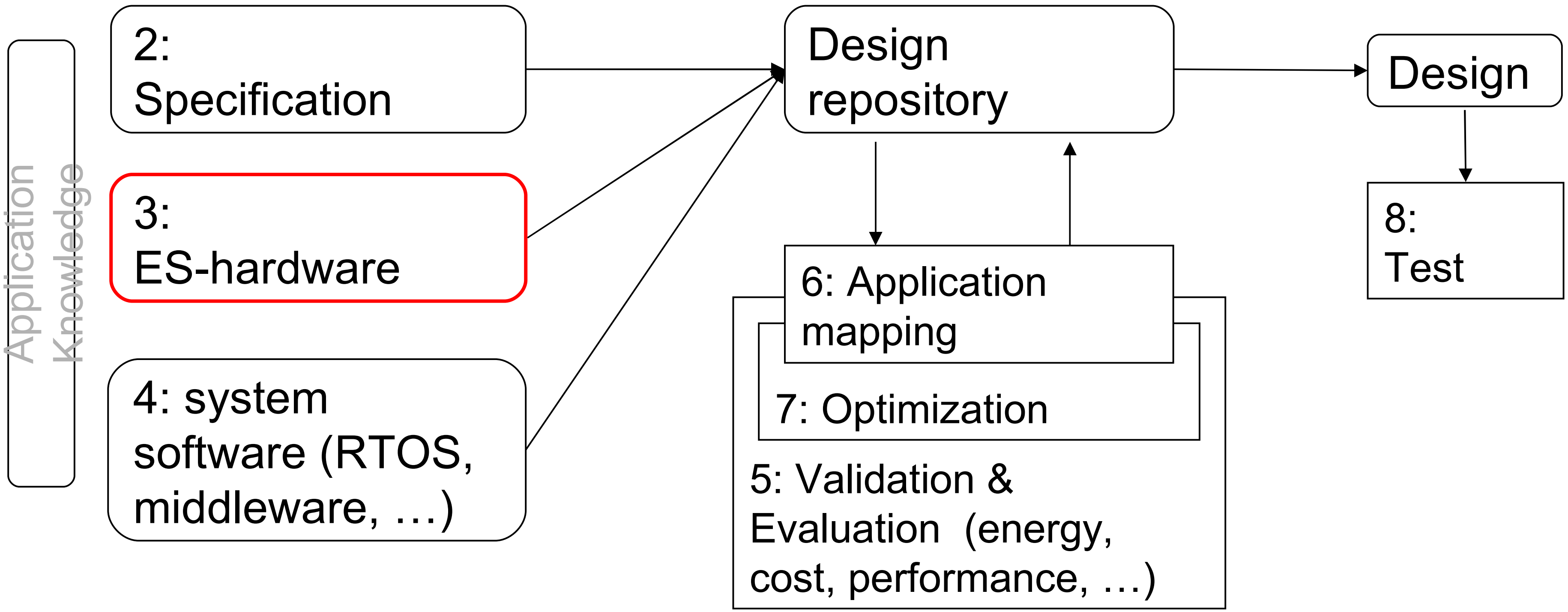
fakultät für informatik
informatik 12

Embedded System Hardware

Peter Marwedel
Informatik 12
TU Dortmund
Germany

2009/11/10

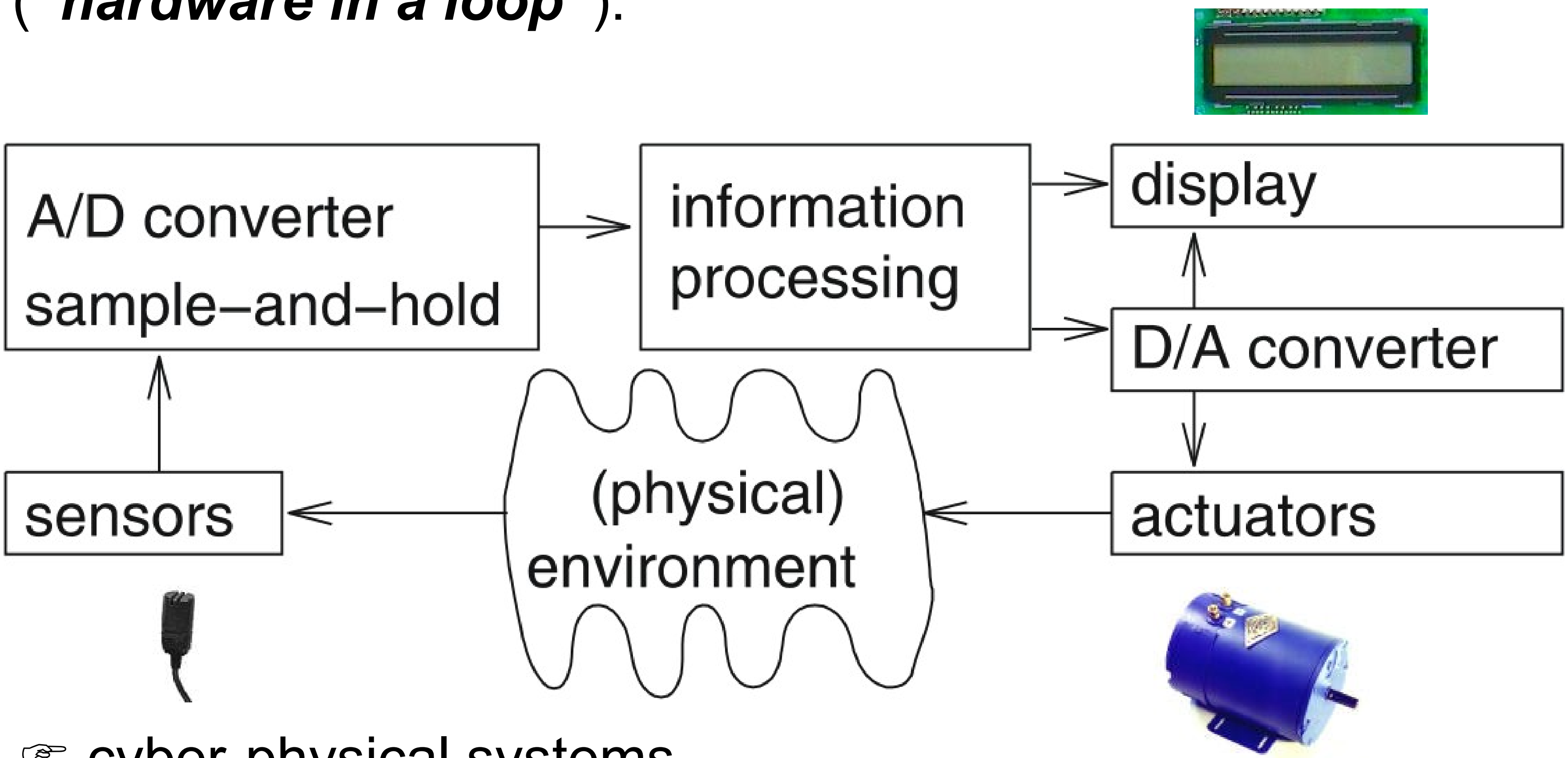
Structure of this course



Numbers denote sequence of chapters

Embedded System Hardware

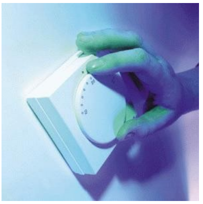
Embedded system hardware is frequently used in a loop (***"hardware in a loop"***):

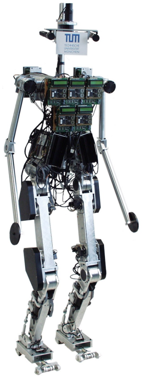


☞ cyber-physical systems

Many examples of such loops

- Heating
- Lights
- Engine control
- Power supply
- …
- Robots





Heating: www.masonsplumbing.co.uk/images/heating.jpg
Robot:: Courtesy and ©: H.Ulbrich, F. Pfeiffer, TU München

Sensors

Processing of physical data starts with capturing this data.

Sensors can be designed for virtually every physical and chemical quantity

- including weight, velocity, acceleration, electrical current, voltage, temperatures etc.
- chemical compounds.

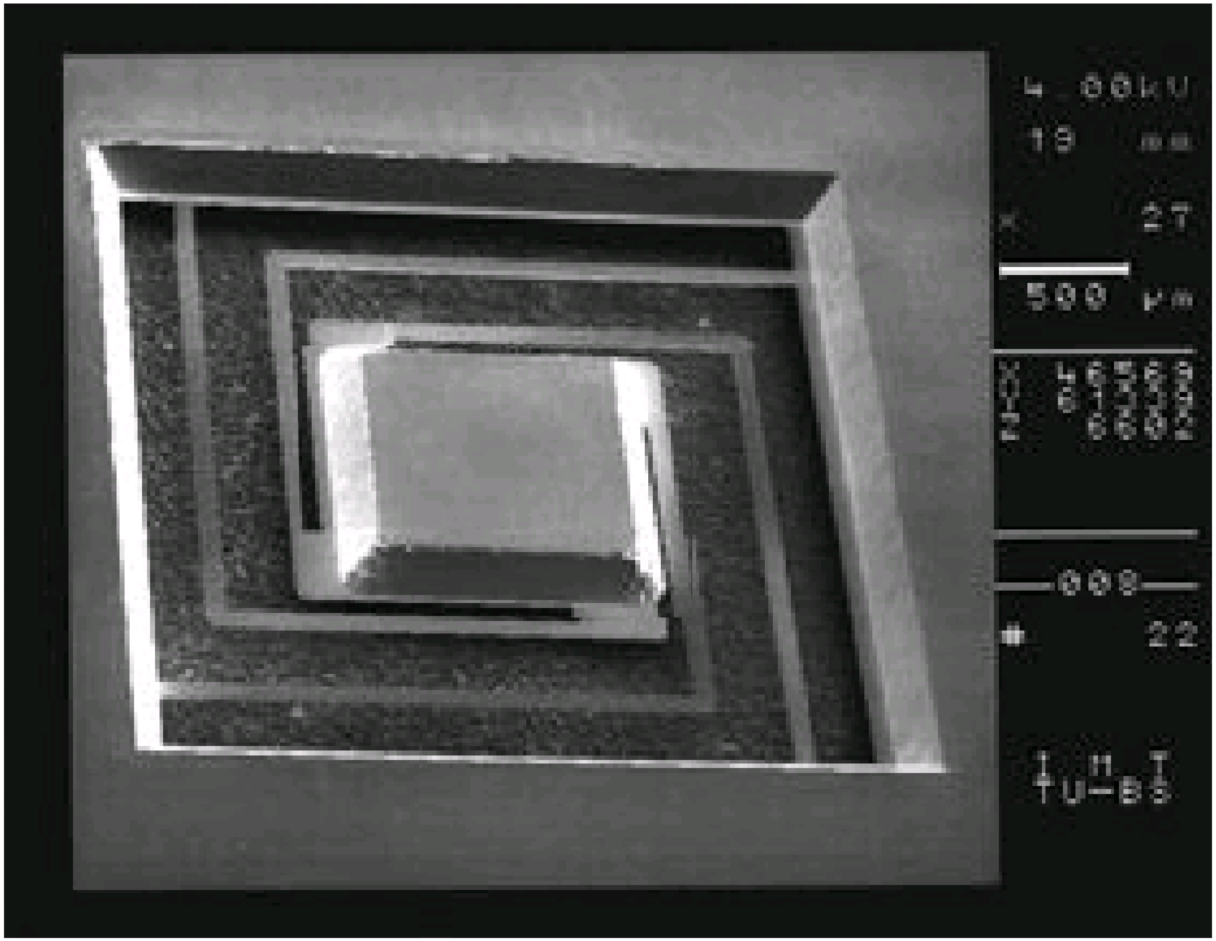
Many physical effects used for constructing sensors.

Examples:

- law of induction (generation of voltages in an electric field),
- light-electric effects.

Huge amount of sensors designed in recent years.

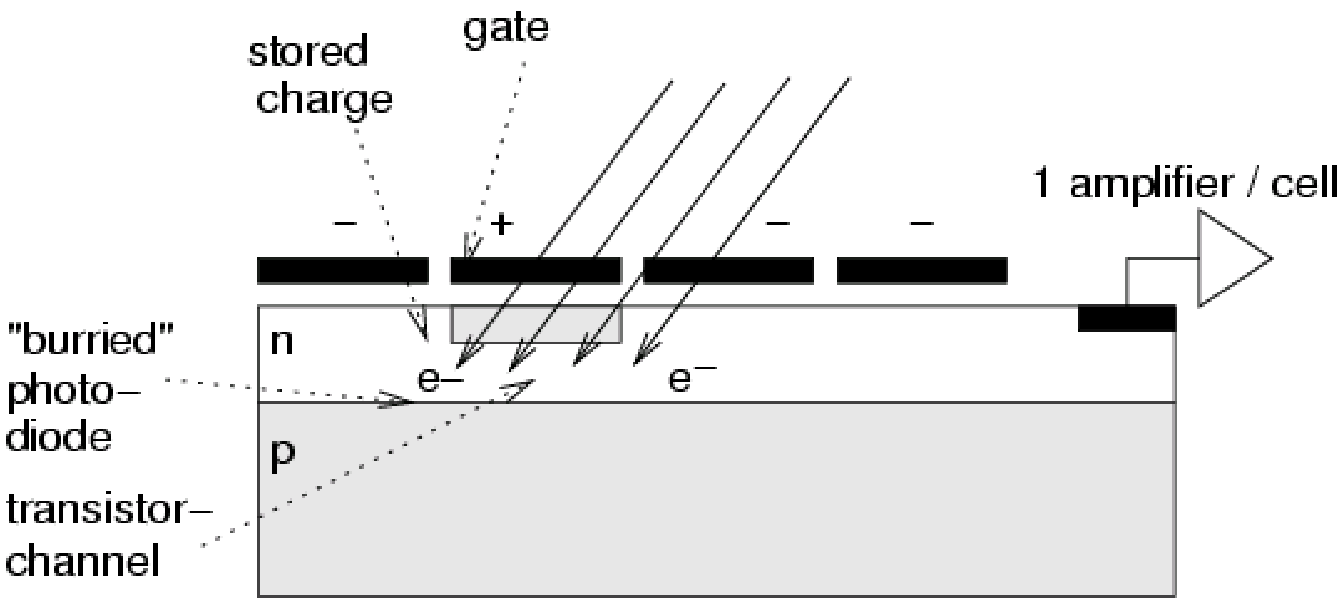
Example: Acceleration Sensor



Courtesy & ©: S. Bütgenbach, TU Braunschweig

Charge-coupled devices (CCD) image sensors

Based on charge transfer to next pixel cell

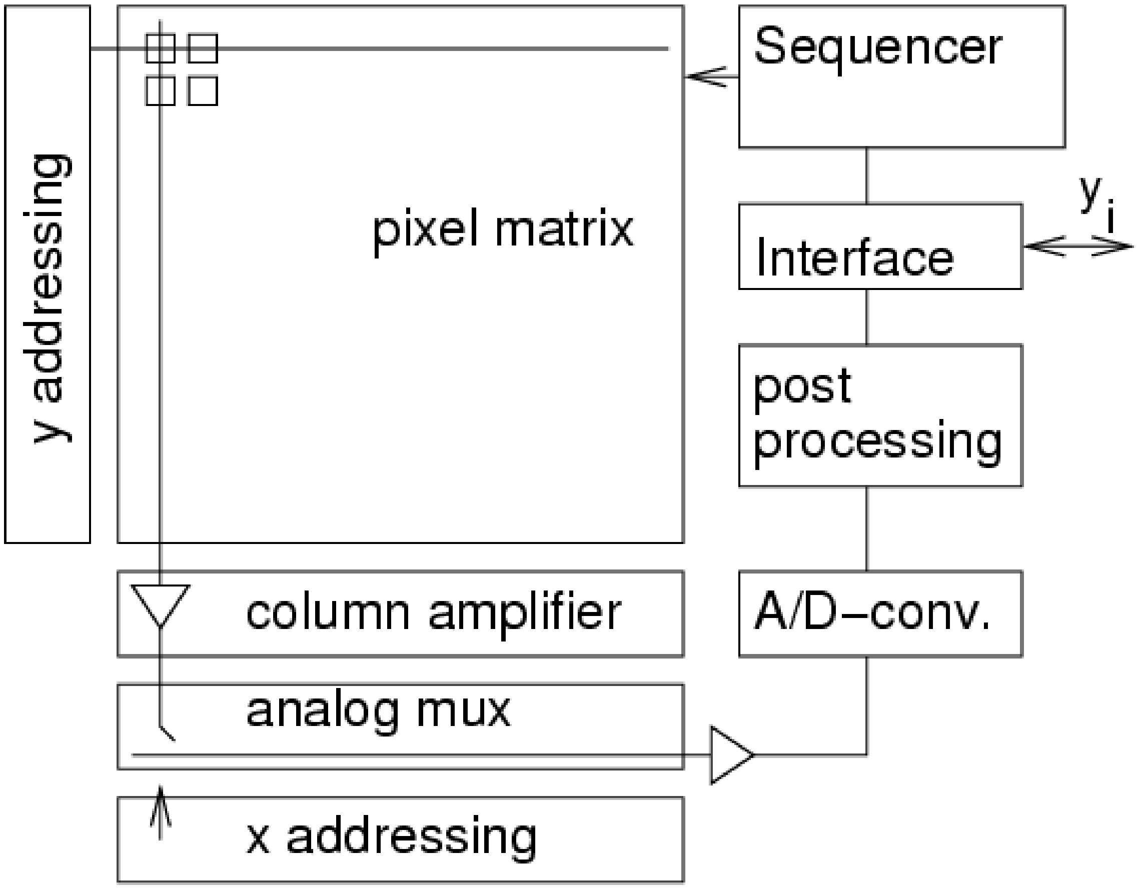


http://www.schulen.regensburg.de/hhgs/klassen/2001a/feuerwehr/kette2.jpg

Corresponding to “bucket brigade device” (German: “*Eimerkettenschaltung*”)

CMOS image sensors

Based on standard production process for CMOS chips, allows integration with other components.



technische universität dortmund

fakultät für informatik

Comparison CCD/CMOS sensors

Property	CCD	CMOS
Signal/noise ratio (SNR)	Excellent	Medium
Dark current	Very low	Medium
Technology optimized for	Optics	VLSI technology
Technology	Special	Standard
Smart sensors?	No, no logic or A/D converters on chip	Logic elements on chip
Access	Serial	Random
Interface	Complex	Simple, single VDD

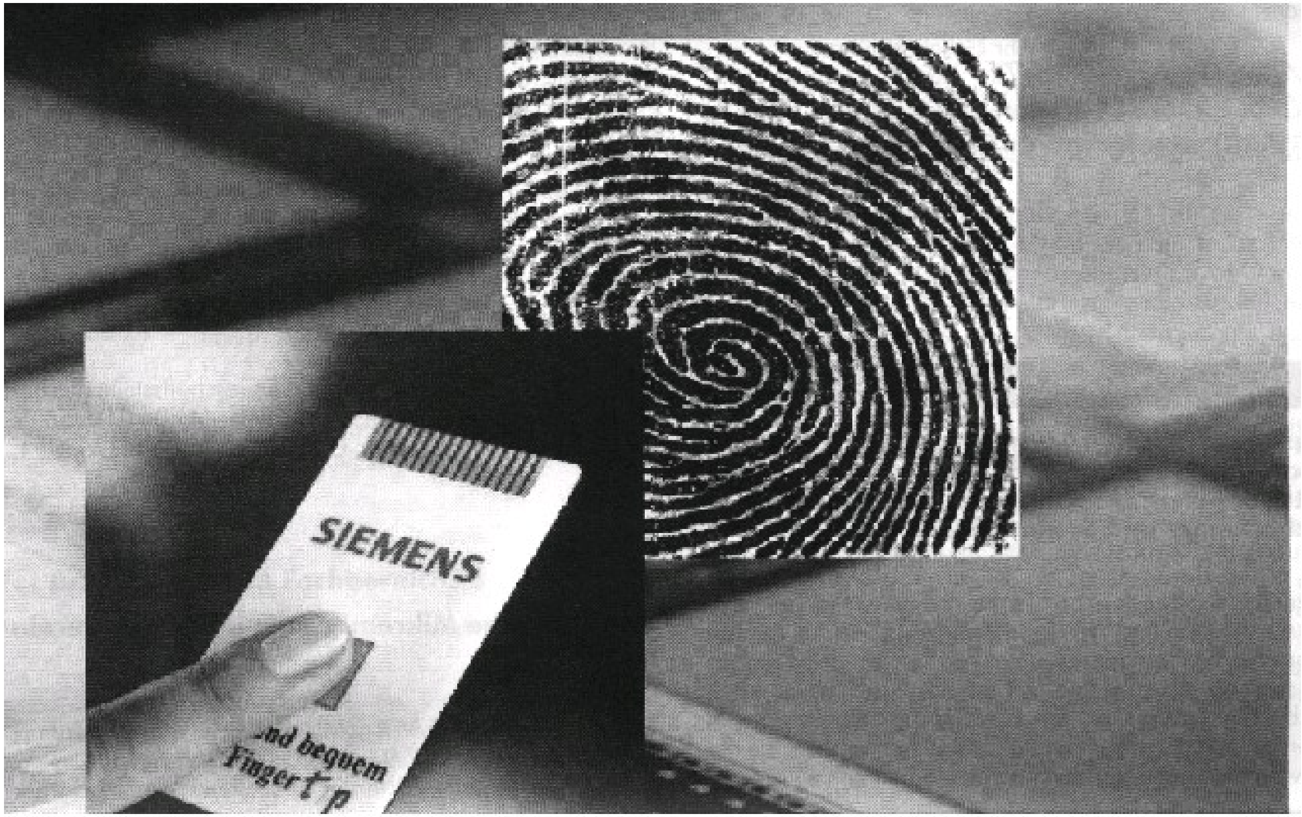
CMOS: low cost devices + digital SLR cameras (due to large size, …)

CCD: medium to high end non-SLR cameras

Source: B. Diericks: CMOS image sensor concepts. Photonics West 2000 Short course (Web)

Example: Biometrical Sensors

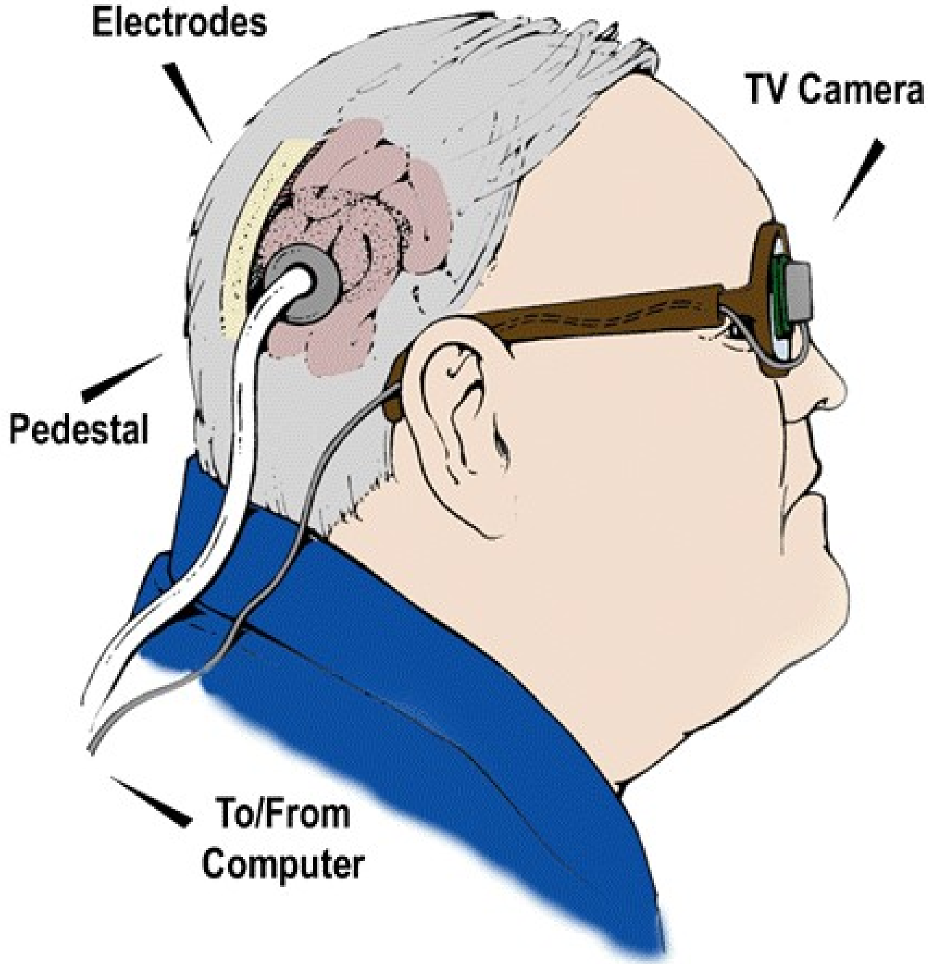
Example: Fingerprint sensor (© Siemens, VDE):



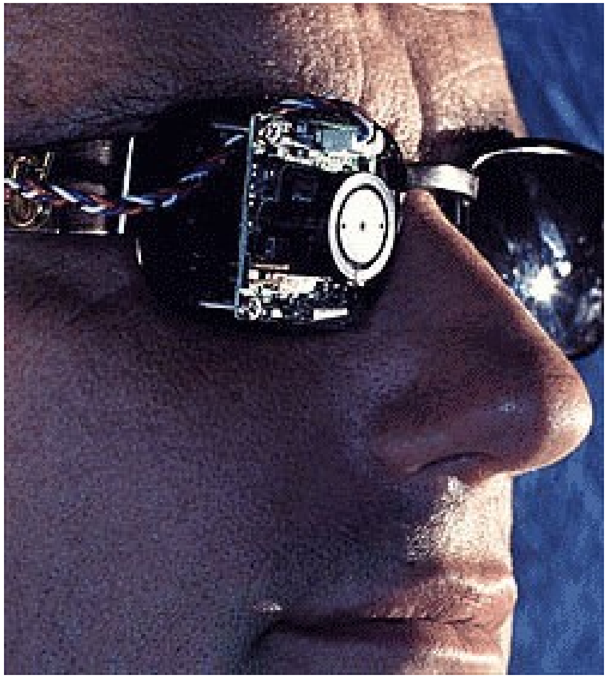
Matrix of 256 x 256 elem.
Voltage ~ distance.
Resistance also computed. No fooling by photos and wax copies.
Carbon dust?

Integrated into ID mouse.

Artificial eyes



Artificial eyes (2)



He looks hale, hearty, and healthy — except for the wires. They run from the laptops into the signal processors, then out again and across the table and up into the air, flanking his face like curtains before disappearing into holes drilled through his skull. Since his hair is dark and the wires are black, it's hard to see the actual points of entry. From a distance the wires look like long ponytails.

© Dobelle Institute

Show movie from www.dobelle.com (e.g. blind person driving a car)

Artificial eyes (3)

- Übersetzung in Schall; angeblich bessere Auflösung [http://www.seeingwithsound.com/etumble.htm]

technische universität dortmund

fakultät für informatik

Other sensors

- Rain sensors for wiper control
  (“Sensors multiply like rabbits“ [ITT automotive])
- Pressure sensors
- Proximity sensors
- Engine control sensors
- Hall effect sensors

technische universität dortmund

fakultät für informatik

Signals

Sensors generate *signals*

Definition: a **signal** *s* is a mapping from the time domain D_T to a value domain D_V:

$$s: D_T \rightarrow D_V$$

D_T: continuous or discrete time domain

D_V : continuous or discrete value domain.

Discretization

Peter Marwedel
Informatik 12
TU Dortmund
Germany

Discretization of time

Digital computers require discrete sequences of physical values

$$s: D_T \rightarrow D_V$$

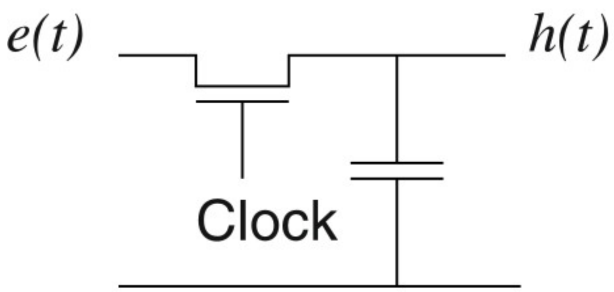
Discrete time domain

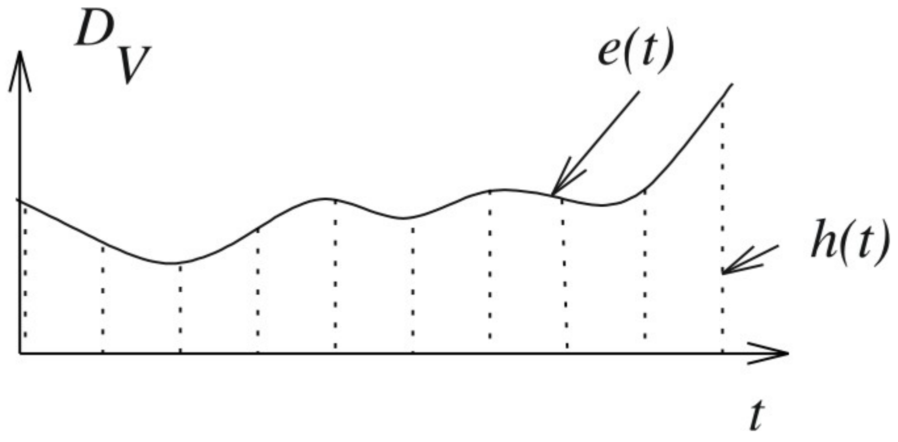
☞ Sample-and-hold circuits

Sample-and-hold circuits

Clocked transistor + capacitor;

Capacitor stores sequence values





$e(t)$ is a mapping $\mathbb{R} \rightarrow \mathbb{R}$

$h(t)$ is a **sequence** of values or a mapping $\mathbb{Z} \rightarrow \mathbb{R}$

Do we loose information due to sampling?

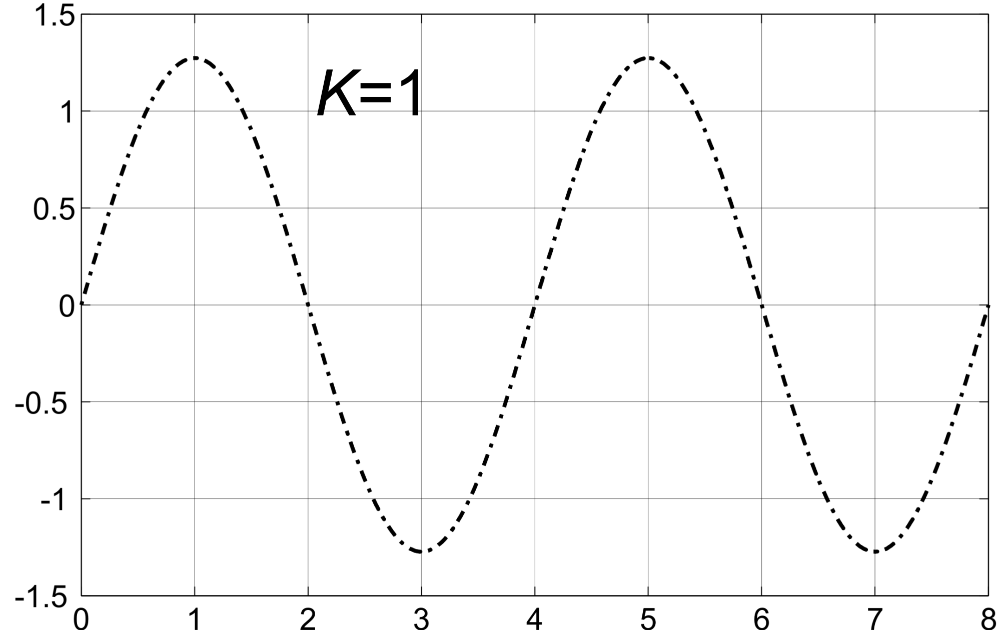
Would we be able to reconstruct input signals from the sampled signals?

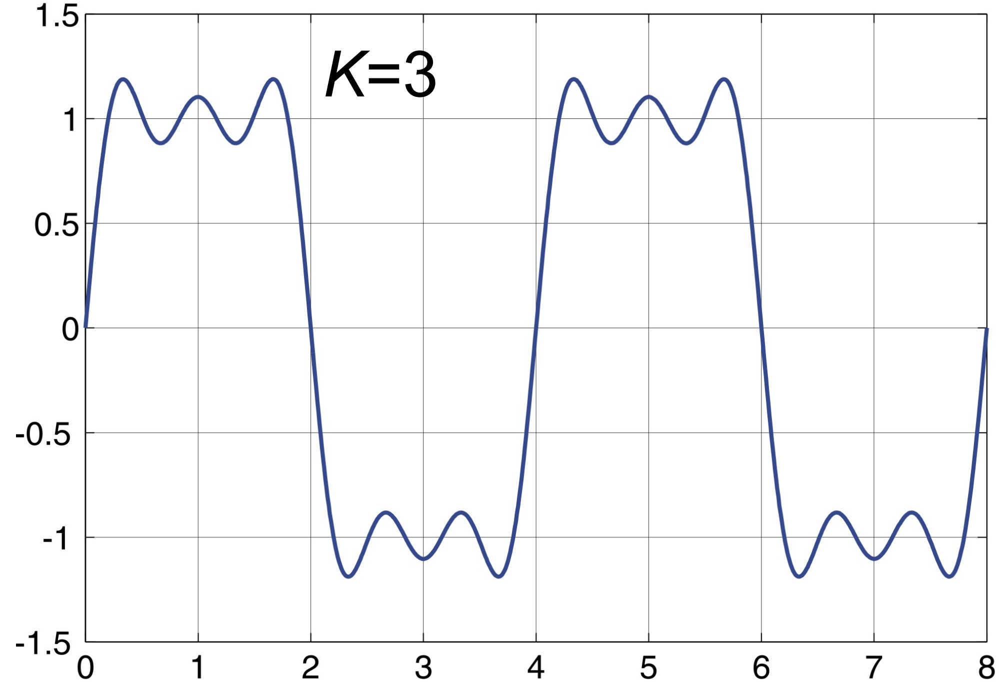
☞ approximation of signals by sine waves.

technische universität dortmund
fakultät für informatik

Approximation of a square wave (1)

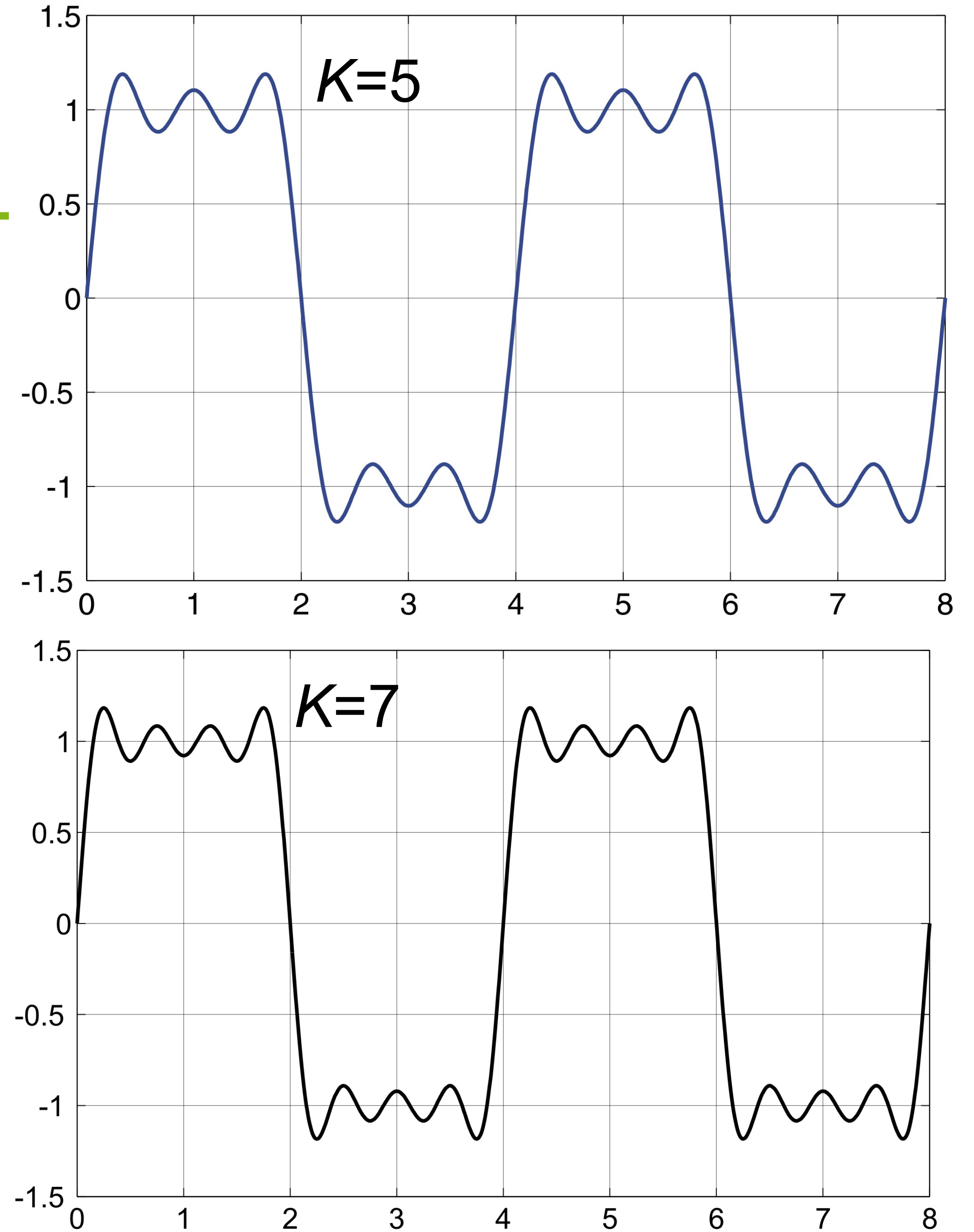
$$e'_K(t) = \sum_{k=1,3,5,..}^{K} \frac{4}{\pi k} \sin\left(\frac{2\pi k}{T}\right)$$





Approximation of a square wave (2)

$$e'_K(t) = \sum_{k=1,3,5,..}^{K} \frac{4}{\pi k} \sin\left(\frac{2\pi k}{T}\right)$$

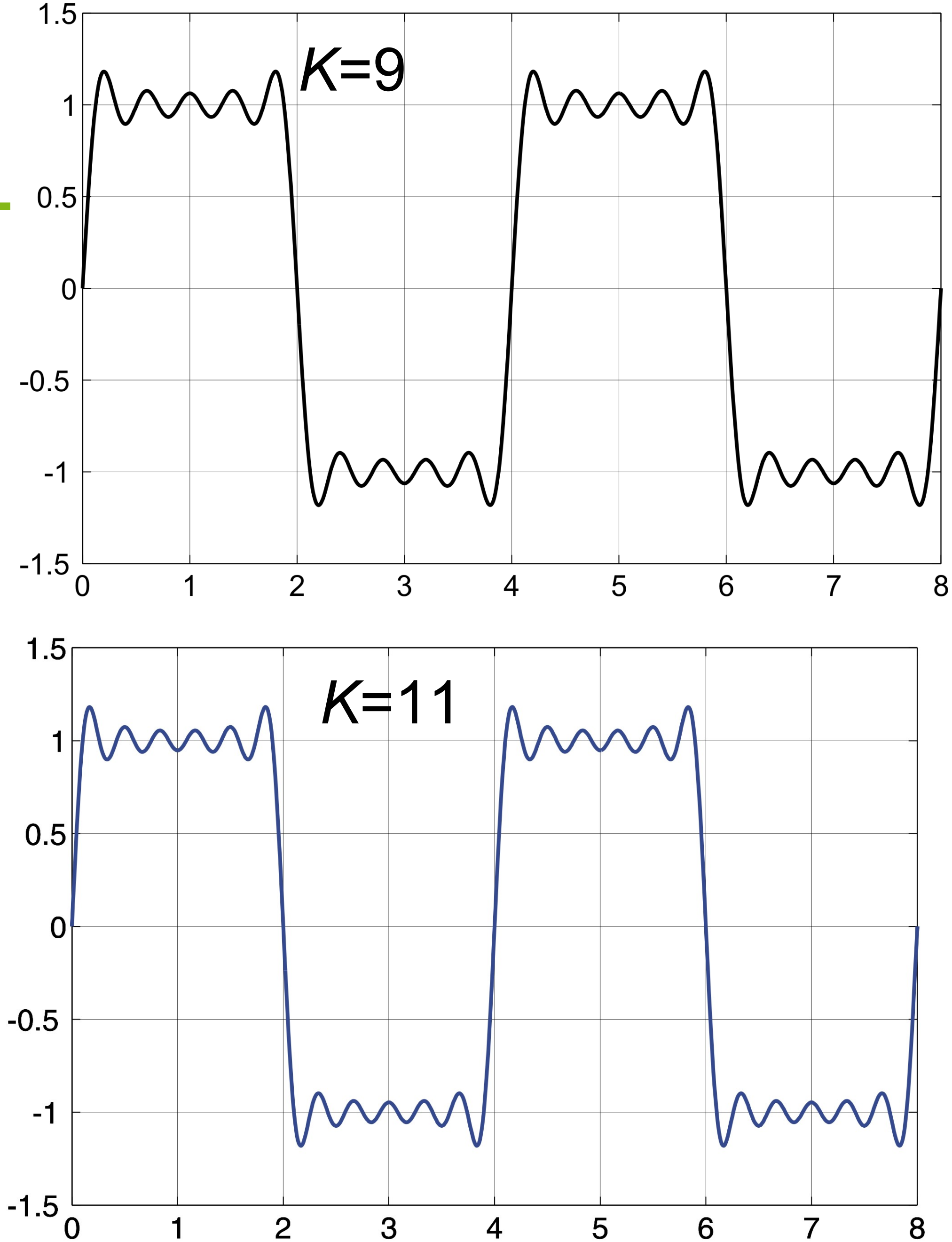


technische universität
dortmund

fakultät für
informatik

Approximation of a square wave (3)

$$e'_K(t) = \sum_{k=1,3,5,..}^{K} \frac{4}{\pi k} \sin\left(\frac{2\pi k}{T}\right)$$



Linear transformations

Let $e_1(t)$ and $e_2(t)$ be signals

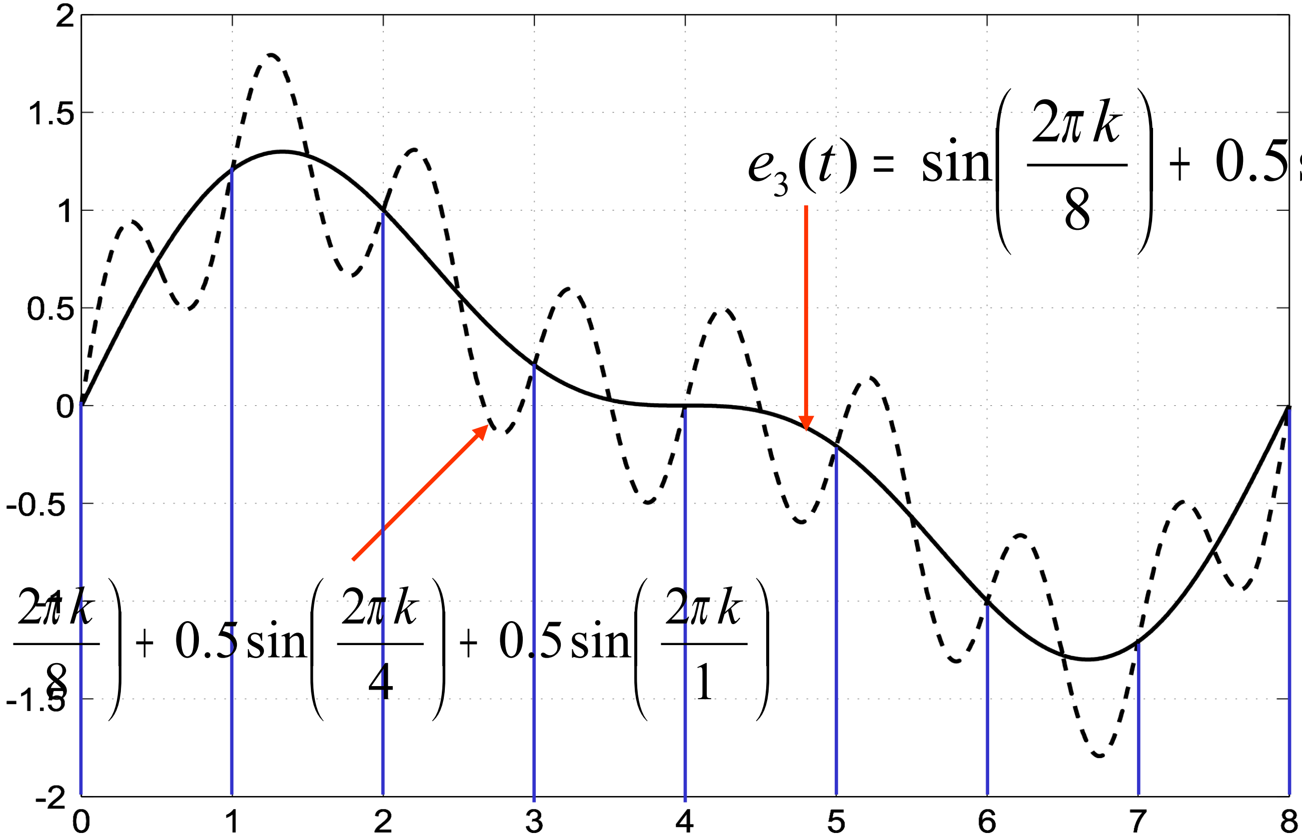
Definition: A transformation Tr of signals is linear iff

$$Tr(e_1 + e_2) = Tr(e_1) + Tr(e_2)$$

In the following, we will consider linear transformations.

☞ We consider sums of sine waves instead of the original signals.

Aliasing



$$e_3(t) = \sin\left(\frac{2\pi k}{8}\right) + 0.5\sin\left(\frac{2\pi k}{4}\right)$$

$$e_4(t) = \sin\left(\frac{2\pi k}{8}\right) + 0.5\sin\left(\frac{2\pi k}{4}\right) + 0.5\sin\left(\frac{2\pi k}{1}\right)$$

Periods of T=8,4,1

Indistinguishable if sampled at integer times T_s=1

Aliasing (2)

☞ Reconstruction impossible, if not sampling frequently enough

How frequently do we have to sample?

Nyquist criterion (sampling theory):

Aliasing can be avoided if we restrict the frequencies of the incoming signal to less than half of the sampling rate.

$T_s < ½\, T$ where T is the period of the "fastest" sine wave

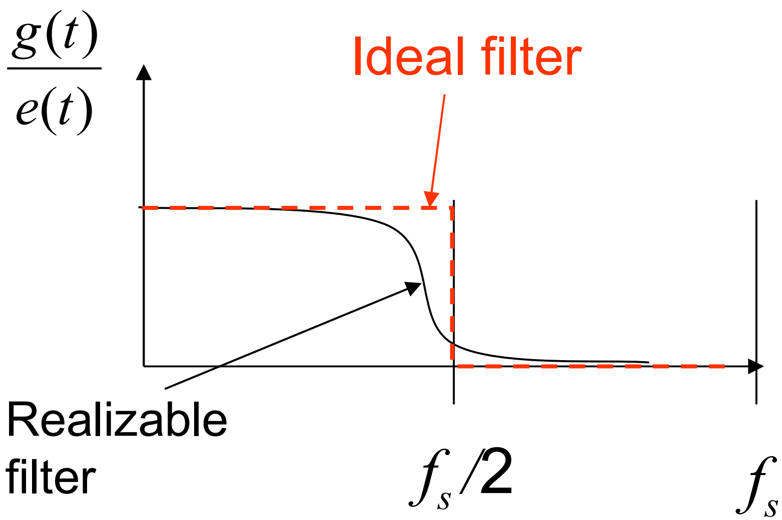
or $f_s > 2\, f$ where f is the frequency of the "fastest" sine wave

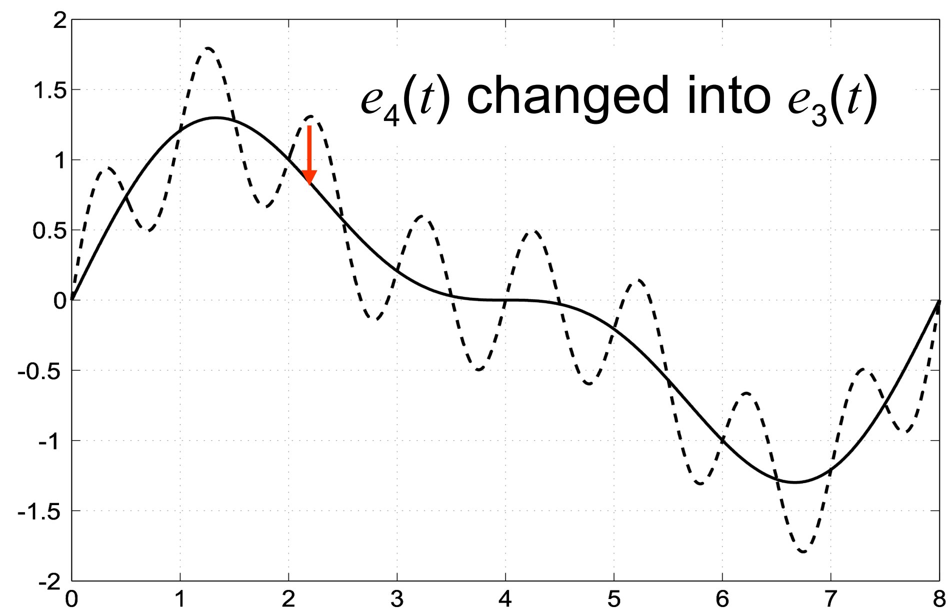
f is called the **Nyquist frequency**, f_s is the **sampling rate**.

See e.g. [Oppenheim/Schafer, 2009]

Anti-aliasing filter

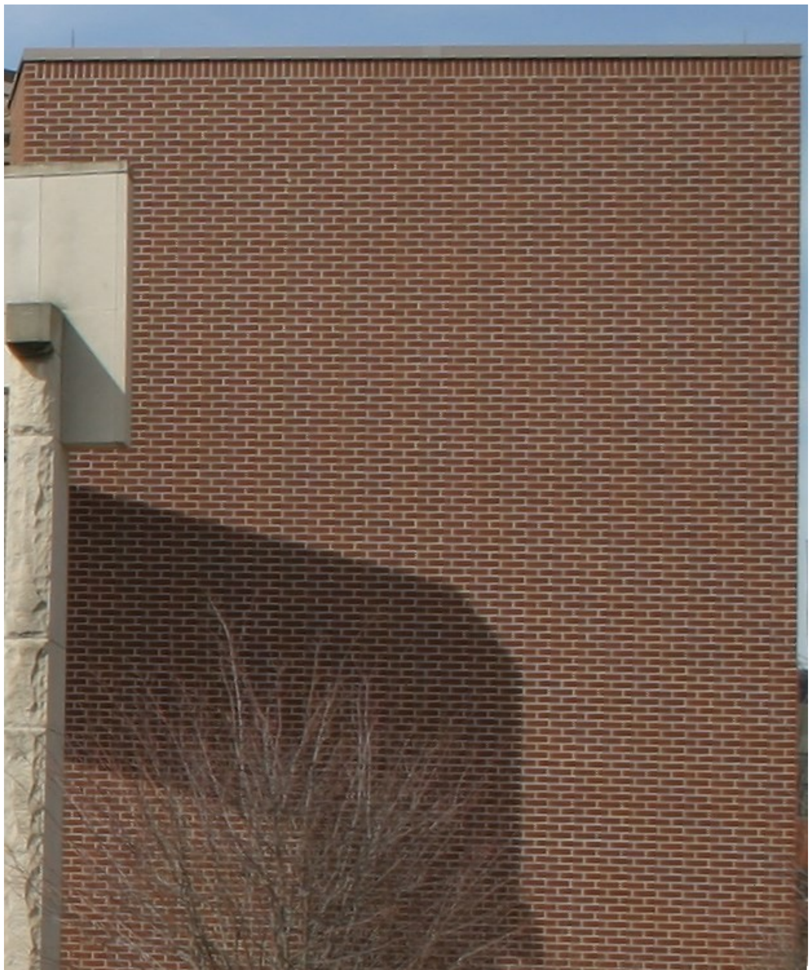
A filter is needed to remove high frequencies





Examples of Aliasing in computer graphics

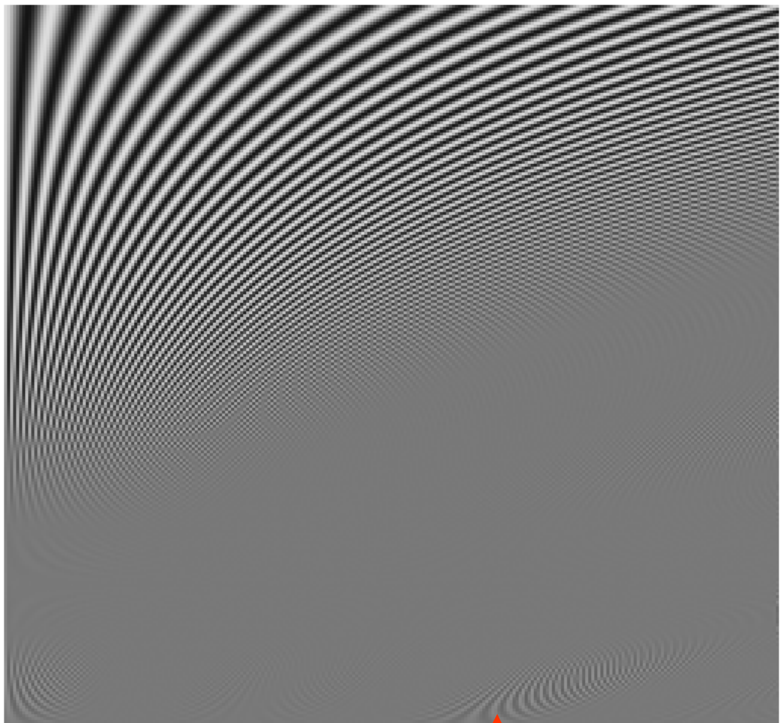
Original



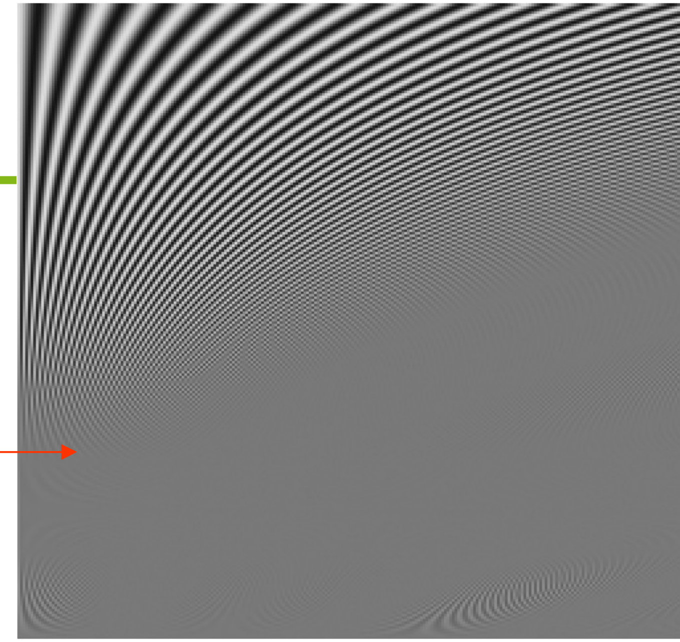
Sub-sampled, no filtering

Examples of Aliasing in computer graphics (2)

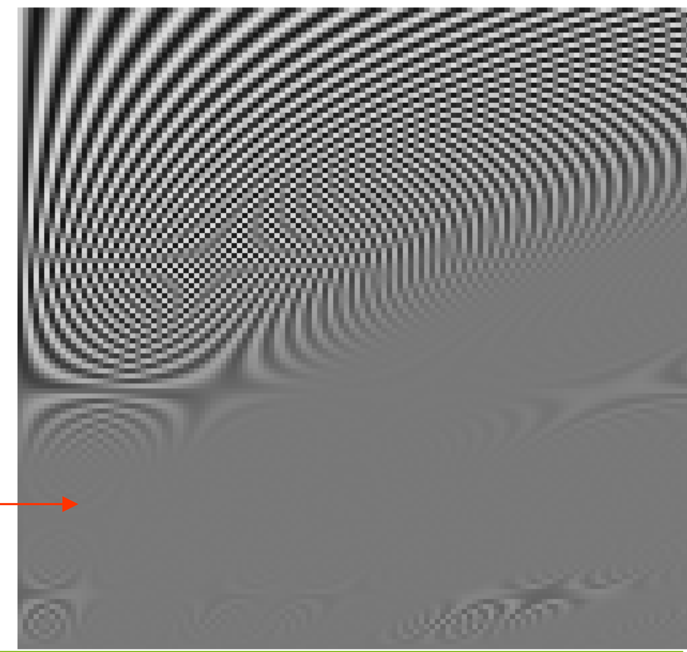
Original (pdf screen copy)



Filtered & sub-sampled



Sub-sampled, no filtering



Impact of rasterization

http://www.niirs10.com/Resources/Reference Documents/Accuracy in Digital Image Processing.pdf

Discretization of values: A/D-converters

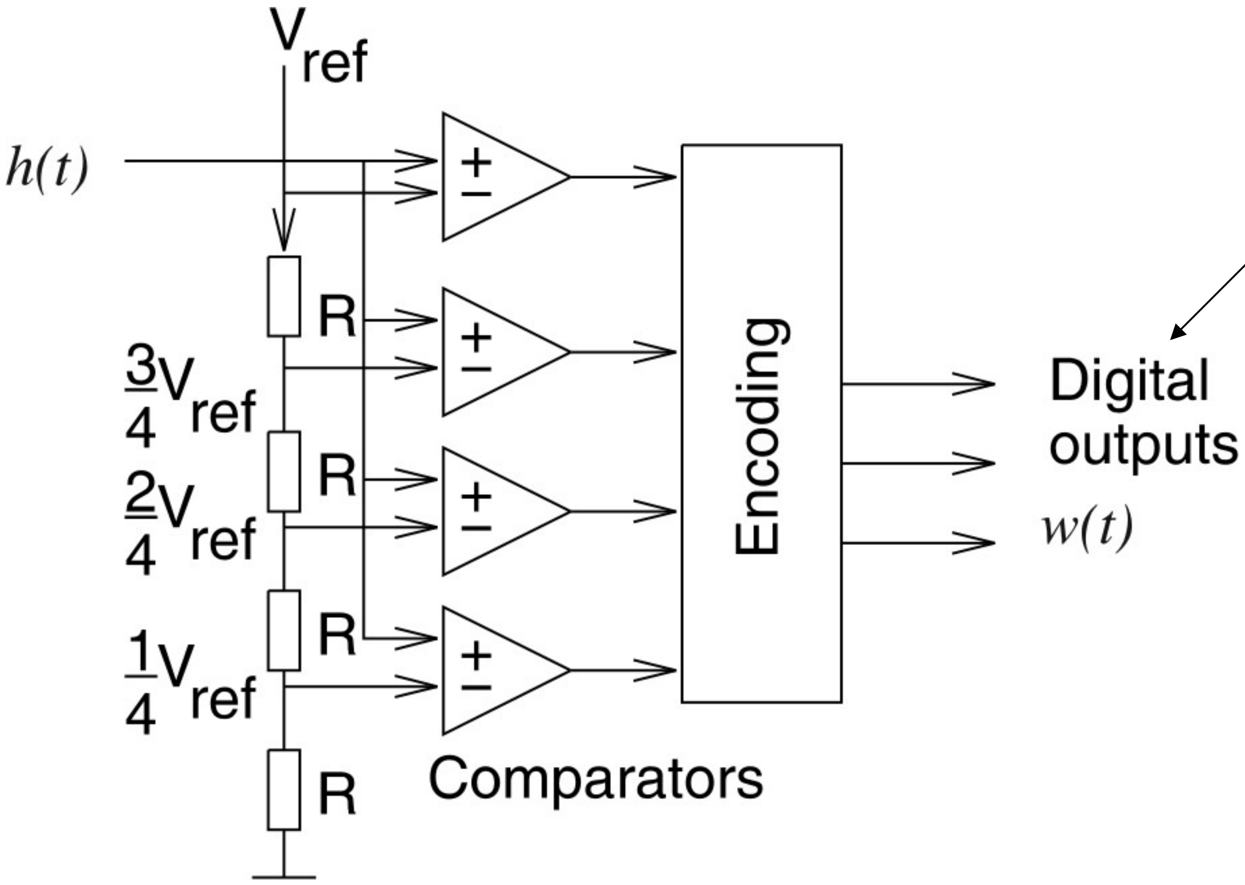
Digital computers require digital form of physical values

$$s: D_T \rightarrow D_V$$

↑ Discrete value domain

☞ A/D-conversion; many methods with different speeds.

Flash A/D converter



Encodes input number of most significant '1' as an unsigned number, e.g.
"1111" -> "100",
"0111" -> "011",
"0011" -> "010",
"0001" -> "001",
"0000" -> "000"
(Priority encoder).

Resolution

- Resolution (in bits): number of bits produced
- Resolution Q (in volts): difference between two input voltages causing the output to be incremented by 1

$$Q = \frac{V_{FSR}}{n}$$ with

Q: resolution in volts per step

V_{FSR}: difference between largest and smallest voltage

n: number of voltage intervals

Resolution and speed of Flash A/D-converter

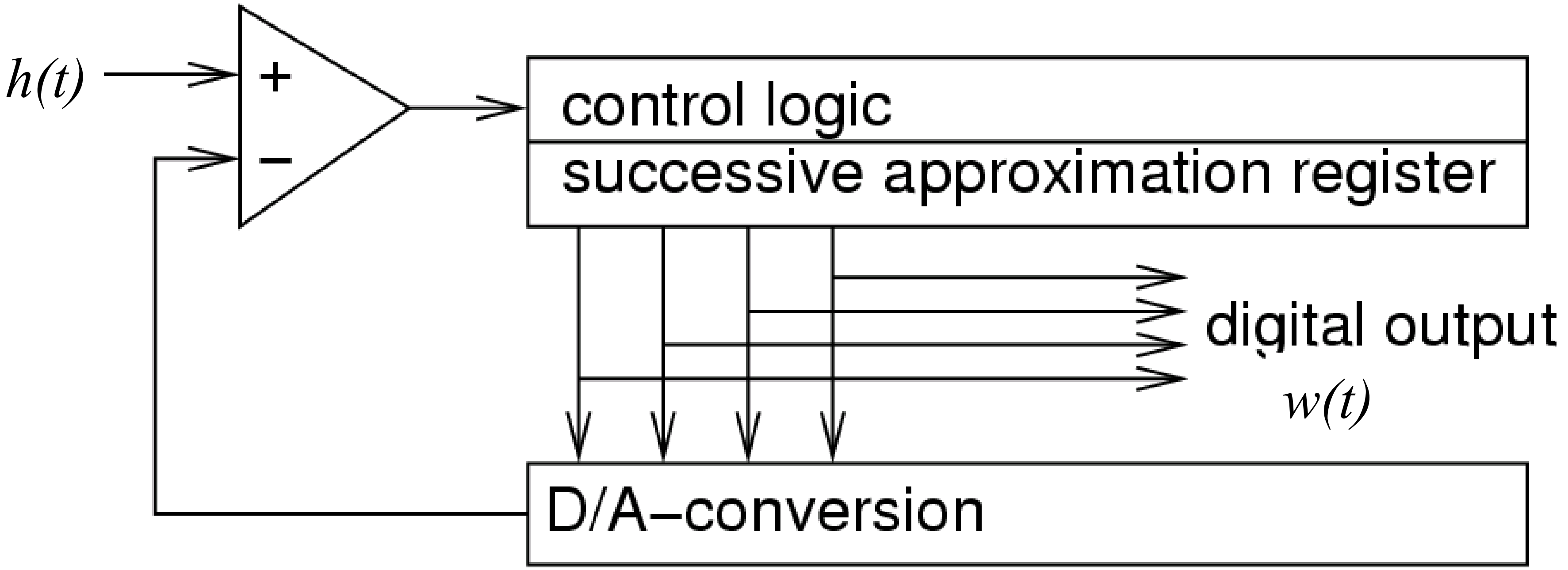
Parallel comparison with reference voltage

Speed: $O(1)$

Hardware complexity: $O(n)$

Applications: e.g. in video processing

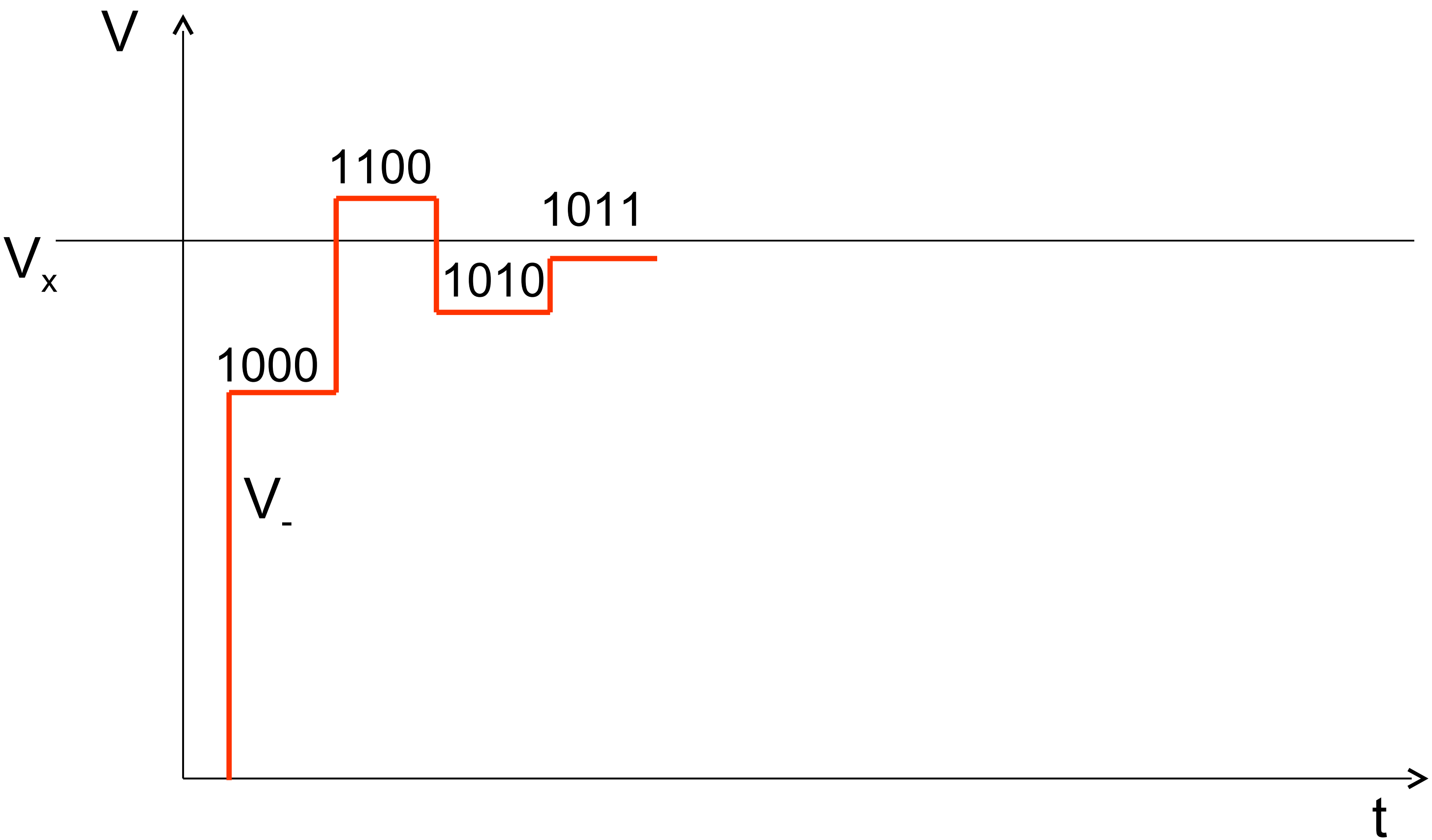
Higher resolution: Successive approximation



Key idea: binary search:
Set MSB='1'
if too large: reset MSB
Set MSB-1='1'
if too large: reset MSB-1

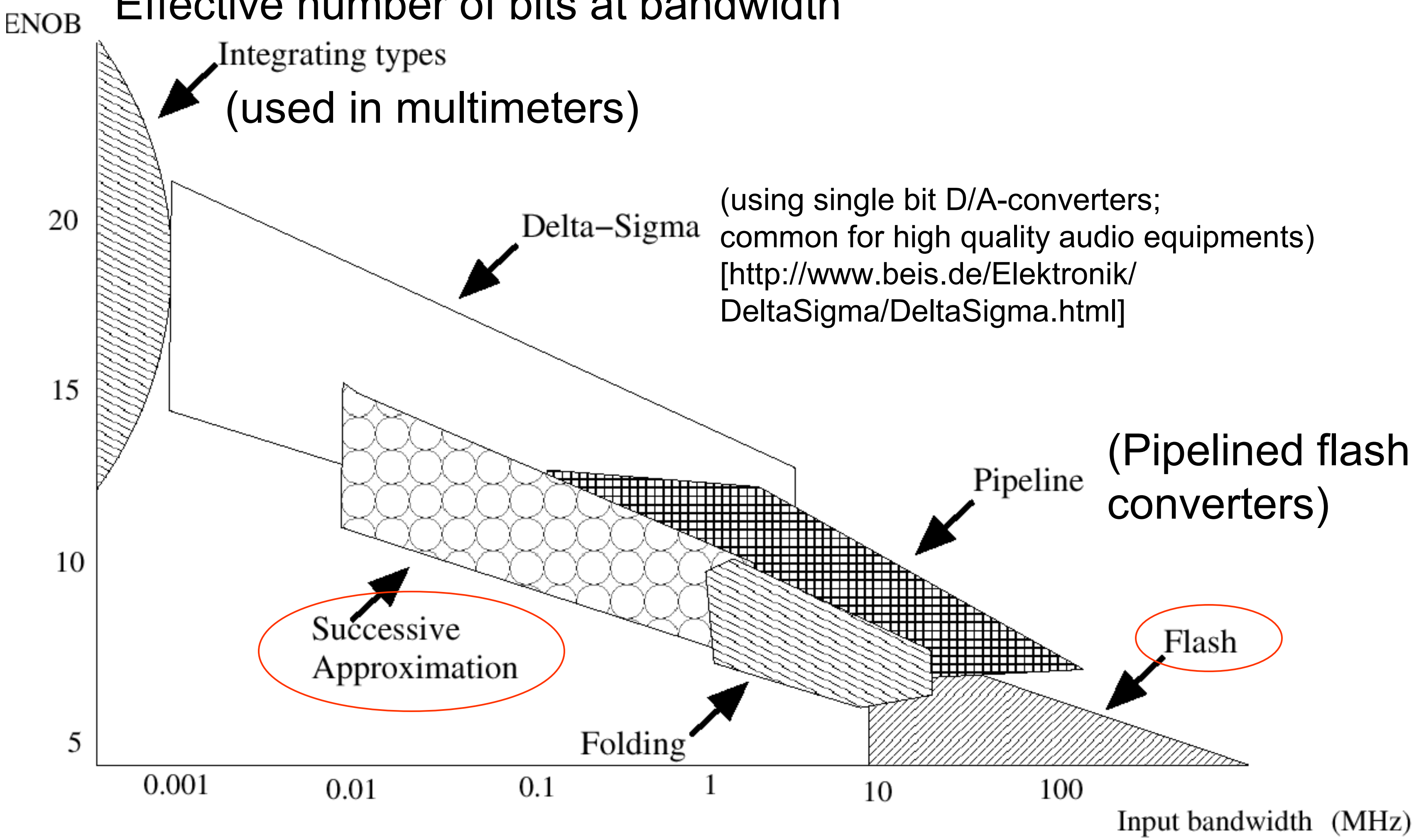
Speed: $O(ld(n))$
Hardware complexity: $O(ld(n))$
with n= # of distinguished voltage levels;
slow, but high precision possible.

Successive approximation (2)



Application areas for flash and successive approximation converters

Effective number of bits at bandwidth



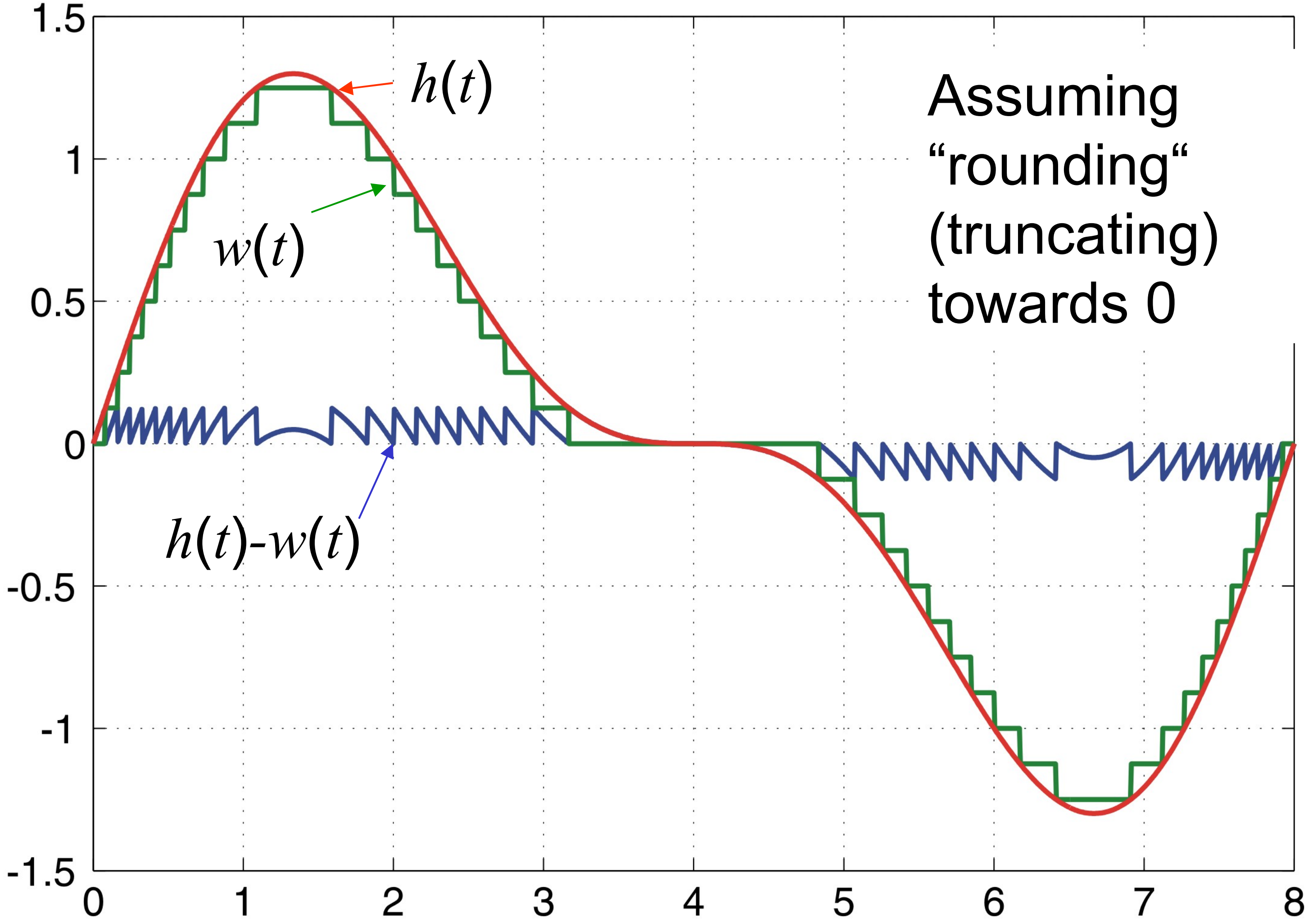
(used in multimeters)

(using single bit D/A-converters; common for high quality audio equipments) [http://www.beis.de/Elektronik/DeltaSigma/DeltaSigma.html]

(Pipelined flash converters)

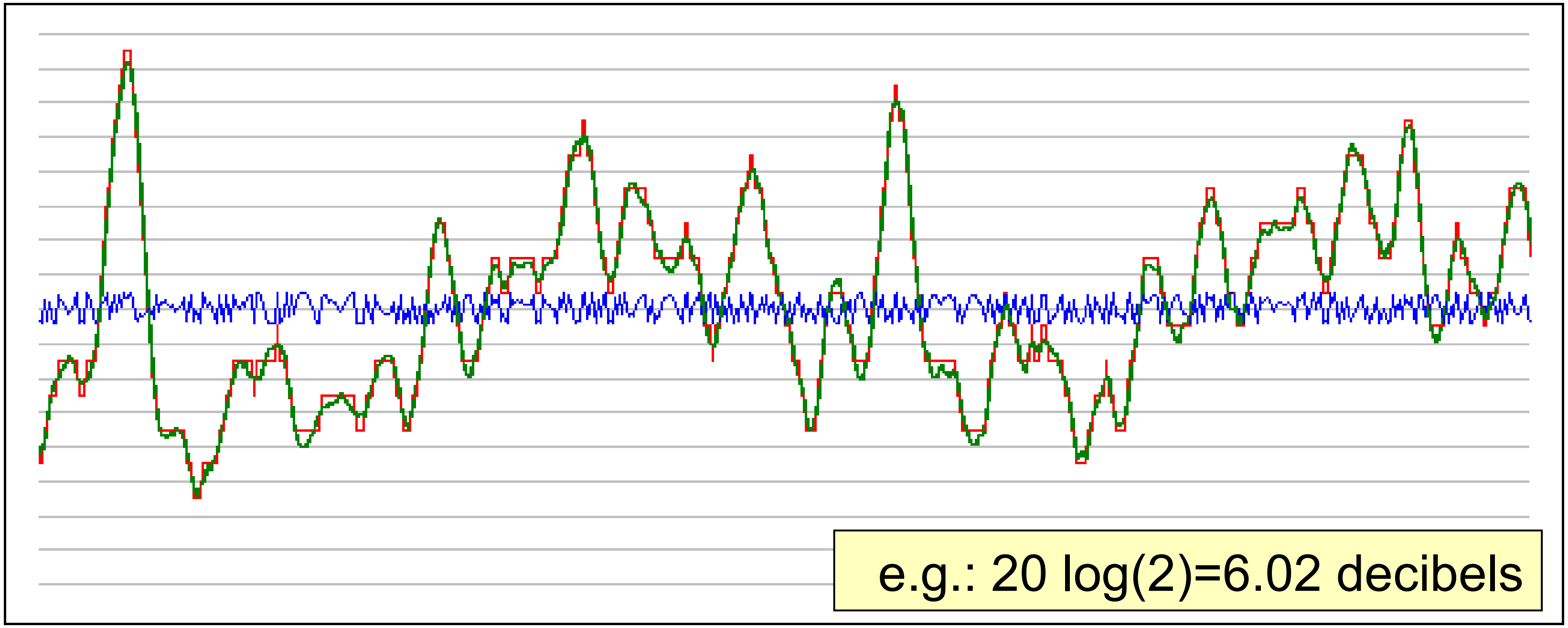
[Gielen et al., DAC 2003]

Quantization Noise



Assuming "rounding" (truncating) towards 0

Quantization noise for audio signal



e.g.: 20 log(2)=6.02 decibels

$$\text{signal to noise ratio (SNR) [db]} = 20 \log \left(\frac{\text{effective signal voltage}}{\text{effective noise voltage}} \right)$$

Signal to noise for ideal n-bit converter : n * 6.02 + 1.76 [dB]

e.g. 98.1 db for 16-bit converter, ~ 160 db for 24-bit converter

Additional noise for non-ideal converters

Source: [http://www.beis.de/Elektronik/DeltaSigma/DeltaSigma.html]

Summary

Hardware in a loop

- Sensors
- Discretization
  - Sample-and-hold circuits
    - Aliasing (and how to avoid it)
    - Nyquist criterion
  - A/D-converters
    - Quantization noise

technische universität
dortmund
fakultät für
informatik