Signals, Instruments, and Systems

School of Architecture, Civil and Environmental Engineering

EPFL, SS 2025-2026

https://disal.epfl.ch/teaching/signals_instruments_systems/

Signals, Instruments, and Systems – W1

Part I: Course Organization, Team, and Content

Team Beyond This Course

Distributed Intelligent Systems and Algorithms Laboratory:

https://disal.epfl.ch

- **Instructor:** Alcherio Martinoli

- **Teaching assistants:**
  - Alexander Wallén Kiessling (Head TA, PhD student)
  - Yacine Derder (TA, PhD student)
  - Wanting Jin (TA, PhD student)
  - Lucas Wälti (TA, PhD student)
  - Sara Maisonhaute (Help TA, master student)
  - Adriana Noujaim (Help TA, master student)

Course Public Pages

- Main entry point for current course:

https://disal.epfl.ch/teaching/signals_instruments_systems/

  - Dedicated pages for lecture, exercises, and project
  - Updated at the same time as the Moodle server

- Previous editions of the course:

https://www.epfl.ch/labs/disal/teaching/previous_courses/

  - 5 previous editions
  - Dedicated pages for lecture, exercises, and continuous assessment, including all previous assignments (e.g., self-verification and lab verification tests, homework or course project)

Course Rationale, Content, and Prerequisites

Typical Field Instrumentation for Environmental Monitoring

Ultrasound anemometer

Laser-based disdrometer

Integrated compact weather station (temperature, humidity, anemometer disdrometer)

Data logger

Local Climate Monitoring

Features:
- Very low sampling frequency < 1Hz
- Very low power consumption: 25mW
- Solar panel
- Radio communication

Sensors:
- Air Temperature and Humidity
- Infrared Surface Temperature
- Anemometer
- Solar Radiation
- Pluviometer
- Soil moisture
- Soil pressure

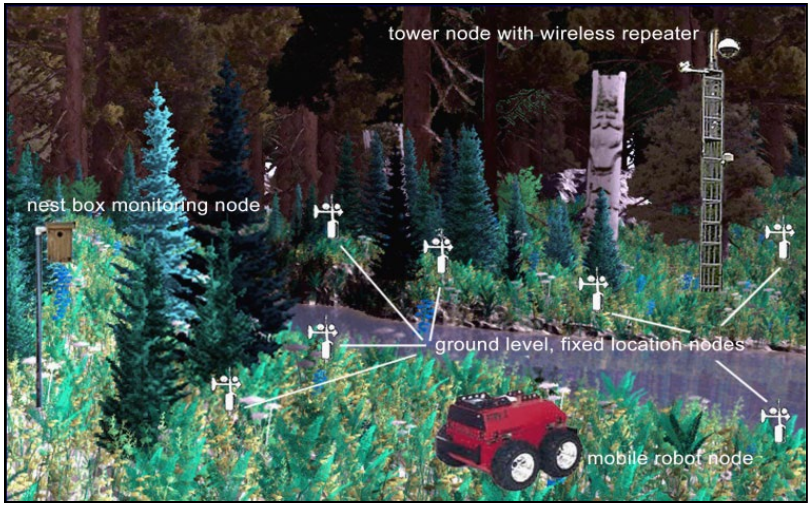
Ecosystem Monitoring

Science

- Understand response of wild populations (plants and animals) to habitats over time.
- Develop in situ observation of species and ecosystem dynamics.

Techniques

- **Data acquisition of physical and chemical properties, at various spatial and temporal scales, appropriate to the ecosystem, species and habitat.**
- **Automatic identification** of organisms (current techniques involve close-range human observation).
- Measurements over long period of time, taken *in-situ*.
- Harsh environments with extremes in temperature, moisture, obstructions, ...

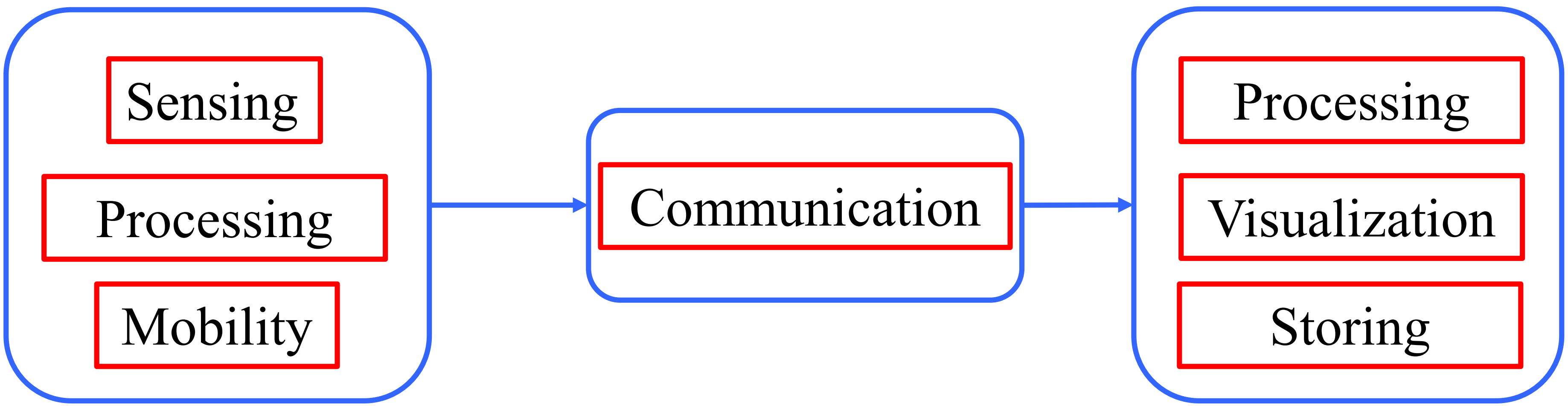


Source: D. Estrin, UCLA

Embedded Systems at the Heart of Modern Environmental Engineering

- Moving the lab to the field is "in"!
- Most of the applications require large spatial distributions (scale of the domain >> scale of a node) → sensor networks and/or mobility
- The underlying hardware/software technology (at the single device level) share the same principles

What These Systems Share at their Core?



In-situ instrument   Transportation channel   Base station

The goal of this course is to shed light on this process and blocks!

What This Course Is About

- Fundamentals of signal processing:
  - Analog/digital signals, sampling and reconstruction
  - Time/frequency domains and transforms
  - Filters, converters

- Fundamentals of embedded systems:
  - Programming skills consolidation: interpreted (Matlab/Python) and compiled (C) languages
  - C for embedded and real-time systems
  - Microcontrollers and memory
  - Sensors and actuators
  - Basic control, communication, and localization techniques

Prerequisites and Related Courses

- Fundamentals in compiled (C) and interpreted (Matlab, Python) language programming
- Fundamentals in probability calculus
- Fundamentals in analysis (Analysis I to IV, ODE, Fourier series and analysis, Fourier and Laplace transforms)
- Estimation methods
- ICC can also help with fundamentals in signal processing (though only time domain)

Rationale (1)

- This course must allow you to become a power user of the key instruments in environmental engineering used nowadays (unattended sensing systems, meteorological stations, data loggers, drones, remotely operated vehicles, etc.) and in the near future (robotic sensing systems, etc.)
- Being a power user means not only being an advanced user but also understand enough to collaborate with EE/CS/ME engineers to design the instruments of the future in environmental engineering; idea: complexity shifted more and more from hardware to software → a lot of domain knowledge can be transferred to the instruments at software level
- Confidence on well-established interpreted and compiled languages means essentially no barriers!

Rationale (2)

- Well-balanced course: theory, algorithms, tools and practical exercises
- It should prepare you to better follow a number of master courses (especially in the Environmental Sensing and Computation specialization)
- This course can be considered as an elevator for the master course “Distributed Intelligent Systems” (offered to multiple programs, including SIE)
- It should get you prepared for carrying out a design/semester/master project at DISAL

Organization of the Course

Credits and Workload

- 5 ECTS
- 1 ECTS = 30 h workload → 150 h workload total
- Rough breakdown
  – 60 h lecture (including reading, exam preparation)
  – 35 h labs and self-assessment programming test
  – 55 h course project with individual and team components (including reading, implementation, reporting, demonstration)

Grade

- 50% performance during semester, 50% performance during the exam
- During semester: only course project assignment
- Final written exam: 180 minutes
- During final exam: all covered material is subject to examination, but the exam is open book (no electronic equipment allowed other than a basic, non-programmable pocket calculator)

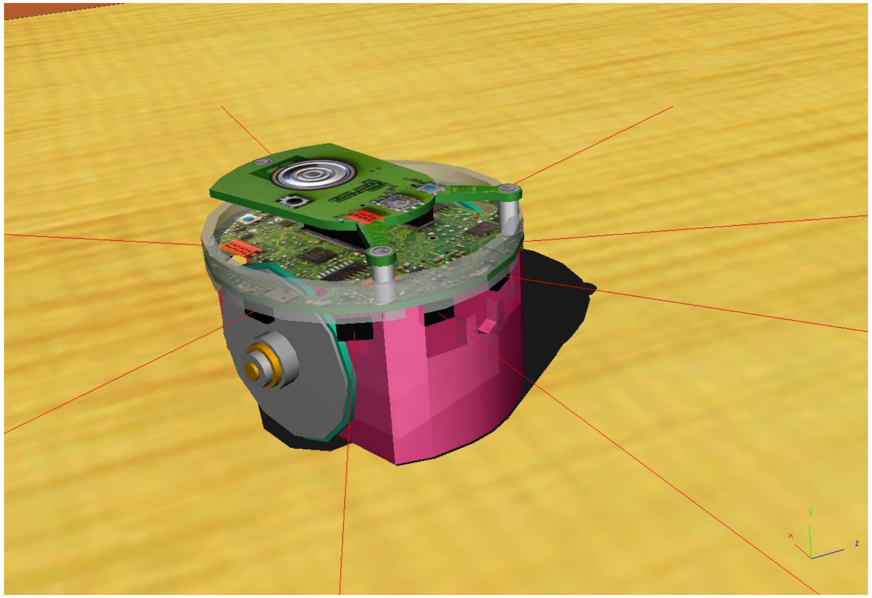
Lecture

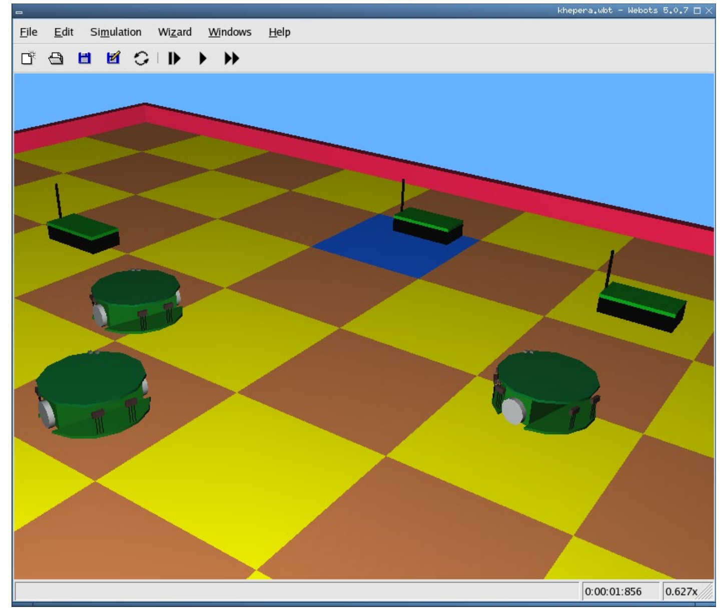
- Policy: no manuscript, but slides and your own notes
- Preliminary lecture slides in pdf format available for download before each lecture (Sun evening), definitive ones after lecture (e-mail notification sent)
- Lecture recordings of previous edition(s) available but some changes possible
- If you do not come to lecture you will miss to develop a fundamental skill for your study and professional life: grasping content/interacting in real-time
- Reading will be added on Moodle as the course progresses
- Access to Moodle: if issues contact Head TA (alexander.kiessling@epfl.ch)

Tools used in the Course

- Matlab and Python
  - Not the same: not all the Matlab toolbox/functions available in Python though libraries for machine-learning richer in Python
  - Engineers and engineering industry still massively using Matlab; IT industry and computer scientists prefer Python
  - Both languages are used in the SIE curriculum and is part of the agility training in programming
- C programming tools (GCC/VSCode) under Linux
- High-fidelity robotic simulator (Webots)
- Real sensor nodes: DISAL Arduino Xbee node

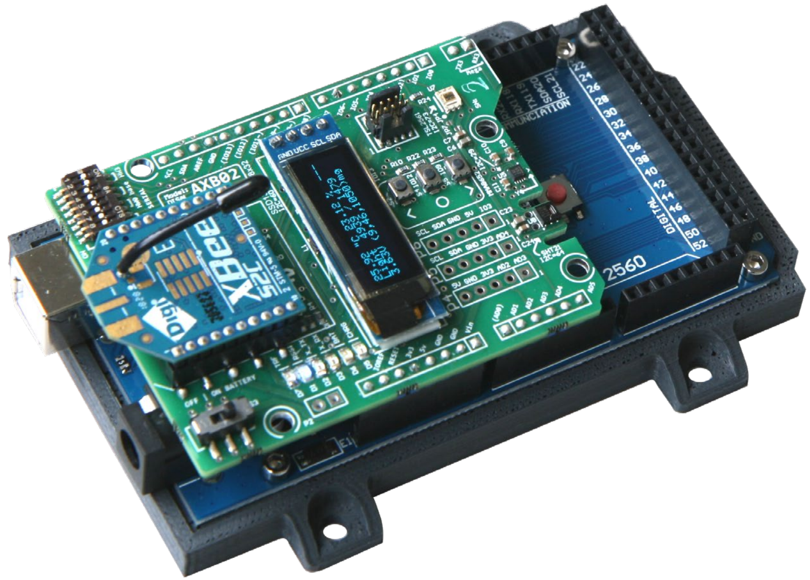
Webots





- High-fidelity simulation
- Discrete sensor and actuators
- Noise and nonlinear characteristics faithfully reproduced
- Different trade-offs faithfulness/computational cost

DISAL Arduino Xbee Node



- Arduino Mega 2560 board (ATMega 2560 microcontroller)
- Zigbee-complaint transceiver (Xbee); on-board mini-display
- Sensors:
  - Light sensor (TSL2561-T, ams)
  - Humidity and Temperature sensor (SHT20, Sensirion)
  - Digital Accelerometer (MMA8652, NXP)
- 14 hours autonomy fully on (70 mA on 1000 mAh Li-Po battery)
- Can be programmed in C leveraging Arduino libraries

Labs (1)

- Lab session: 08:15-11:00 on Thu, GR B0 01
- Mini-tutorial (5-10 min) by TAs at the beginning of the lab
- 9 lab sets total + self-verification programming test
- Official solution available on Moodle after the lab session
- All software tools needed available in the classroom
- Instructions about installation on personal computers communicated through Moodle but no assistance!
- sis-ta@groupes.epfl.ch for ANY issue!

Labs (2)

- Lab assignment posted by Wednesday noon of the same week at latest
- Lab presence not compulsory but
  - We do not repeat labs
  - If you do not come to labs you will have a hard time in the course project, as well as in the final exam (due to partial assimilation)
  - I think it is a lot of fun and really helps you understand lecture stuff
- Assistants are well-prepared (2 TA + 2 Help TAs per lab session)
- Dedicated feedback forms for each lab on Moodle

Suggestions for a Successful Exercise Series

From our teaching experience:

- Read the assignment in advance (i.e. before the lab), in this way you will be more efficient when the TAs are around for helping you on the toughest questions
- Carry out the labs in the GR computer room to benefit of a stable environment with minor logistic overhead
- Take advantage of office hours (upon appointment using the sis-ta list) if additional explanations are needed
- Do the labs seriously (e.g., solve individually while cooperating with your classroom colleagues), do not wait for the solution distributed a couple of days later.

Course Project (1)

- Purpose: verify acquisition of lab knowledge
- Team of 3 students (ad hoc solution with 2 student teams for class numbers not divisible by 3)
- Will leverage uniquely Webots
- Will involve submission of deliverables (report and code)
- From la session Week 8 until end of Week 14 to solve it
- Demo sessions in the last week of the semester

Course Project (2)

- Exclusively 2 lab sessions dedicated to course project assistance for pushing autonomy (one session canceled because of Ascension)
- Solution to be demonstrated by the team on one of the very same machines of the computer rooms during the last week (logistically it will happen in another room)
- Demo conditions might challenge the robustness of the developed solutions
- For 3 student teams (tentative): 5 minutes demo, 9 minutes questions, 1 minute for buffer and changing team.

Suggestions for a Successful Course Project

From our teaching experience:

- Exploit assistance time windows in an efficient way (e.g., ask tough technical questions but do not ask assistance to debug your code!)
- Plan your effort, coordinate your team
- Make sure your code works correctly on one of the machines of the computer rooms
- Be ready to technically arguing about your design choices and comment performance of your solution in real-time during the demo session

Final Notes

18th Iteration at ENAC

- Same graded continuous assessment as last year: course project only, involving an individual and a team component
- Course project topic maintained from last year, some fine tuning based on the student feedback of last year
- Self-verification test in Matlab and C programming maintained

Self-Verification Test

- C and Matlab programming
- Previous edition test:

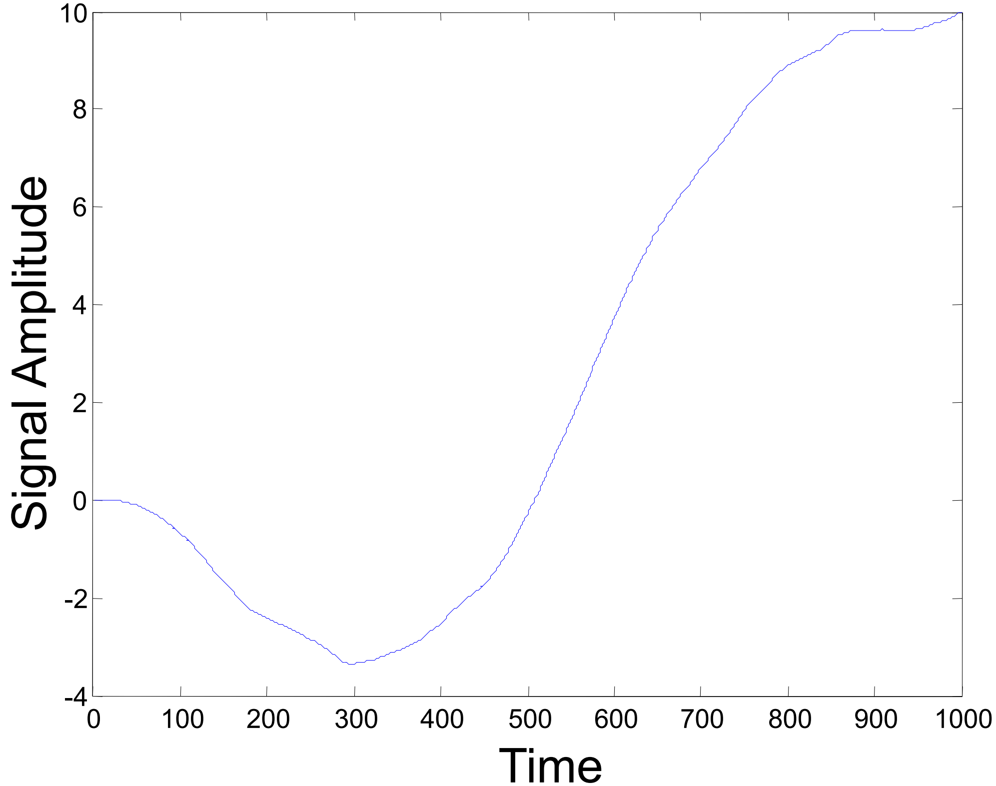
  https://www.epfl.ch/labs/disal/teaching/sis-2024-25/exercises/
- Material for self-study and cheat sheets available on Moodle
- Compulsory for SIE students, score penalty in the individual continuous assessment in case of not carrying out the test by the announced deadline
- Self-assessment uploaded on Moodle in an anonymous fashion
- Test feedback on Moodle highly welcome

Signals, Instruments, and Systems – W1

Part II: An Introduction to Signal Processing – Analog and Digital Signals

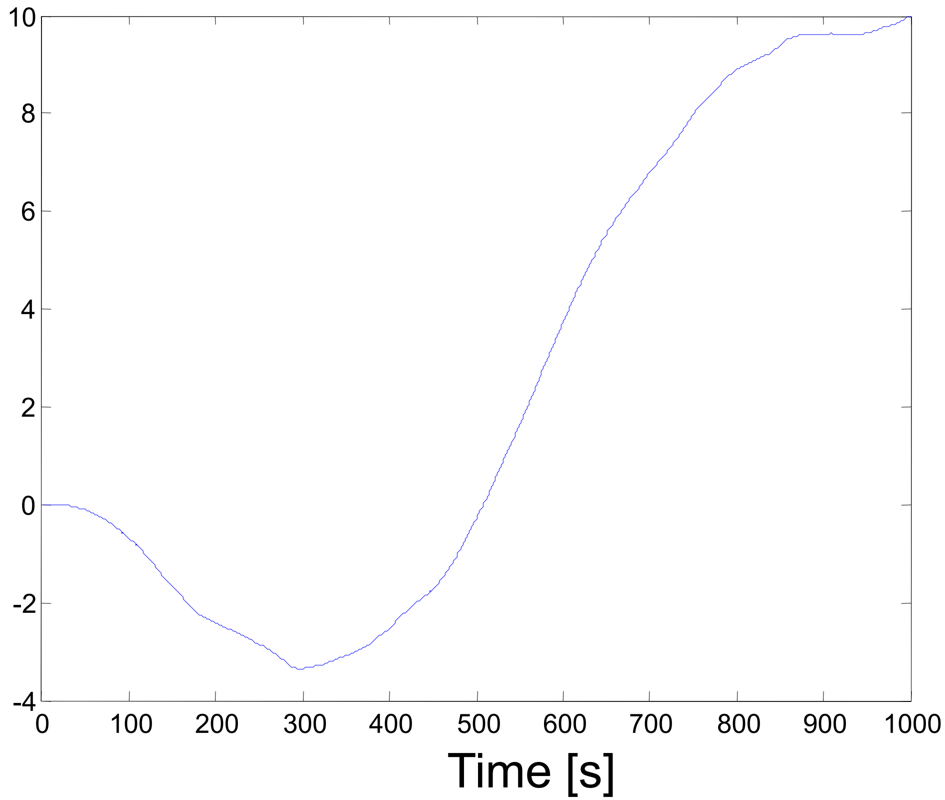
Signal – Definition

"A signal is any **time**-varying or **spatial**-varying quantity"



Signal – Definition

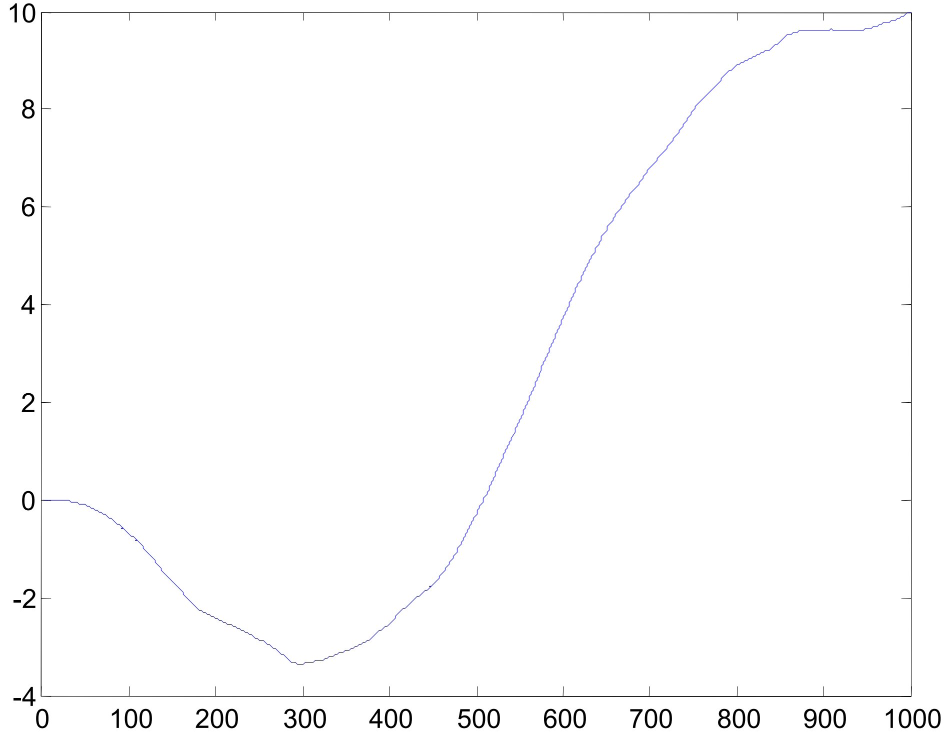
"A signal is any **time**-varying or **spatial**-varying quantity"



$$T = f(t)$$

Signal – Definition

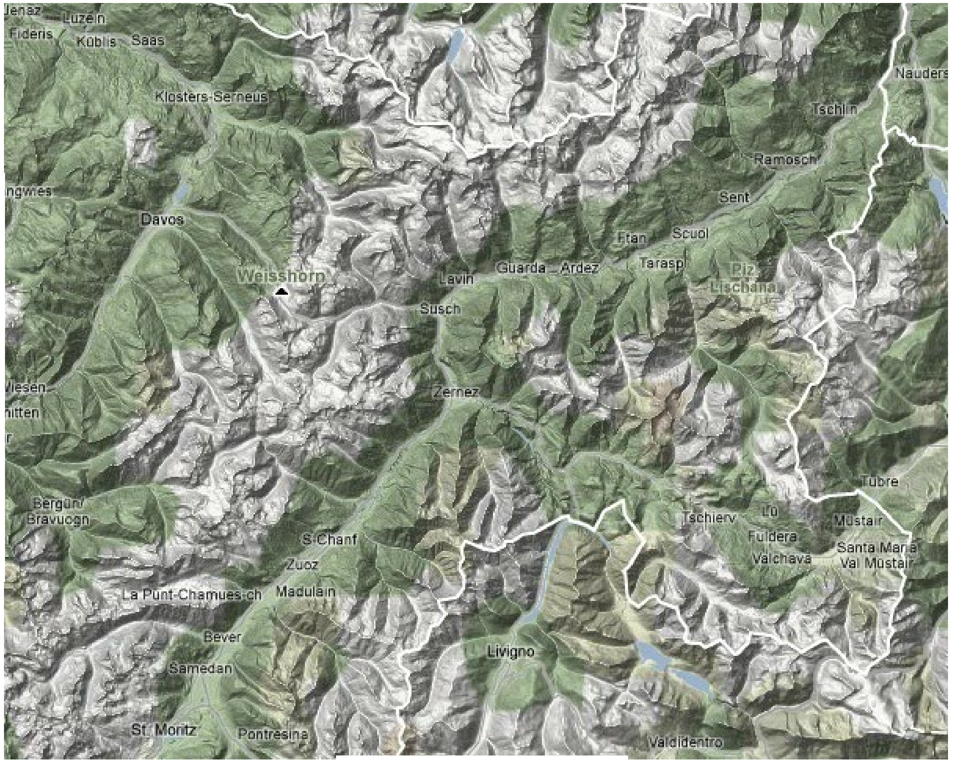
"A signal is any **time**-varying or **spatial**-varying quantity"



$$h = f(x)$$

Signal – Definition

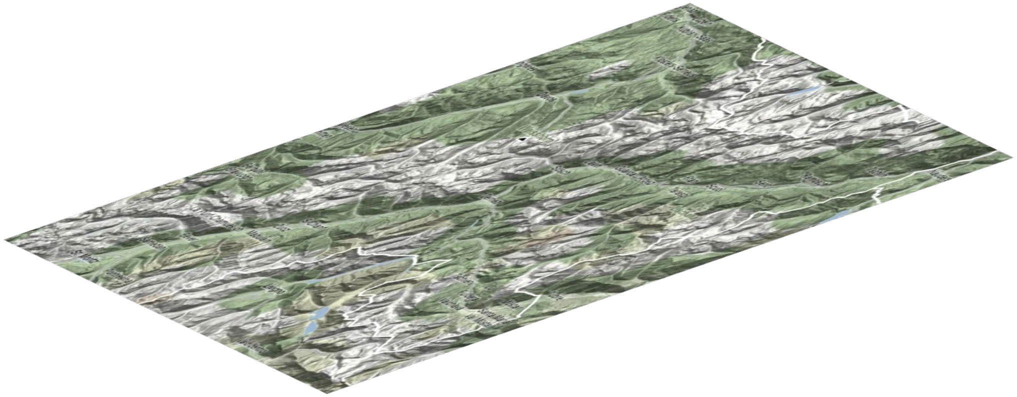
"A signal is any **time**-varying or **spatial**-varying quantity"



Signal – Definition

"A signal is any **time**-varying or **spatial**-varying quantity"

$$h = f(x, y)$$



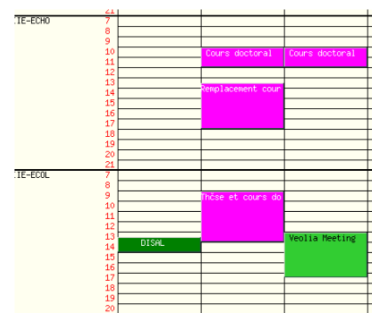
Signal – Definition

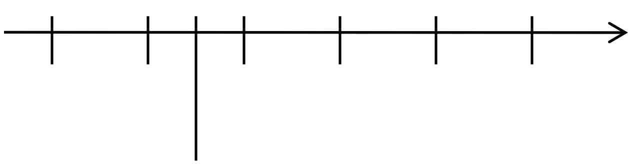
"A signal is any **time**-varying or **spatial**-varying quantity"

Continuous versus **Discrete**

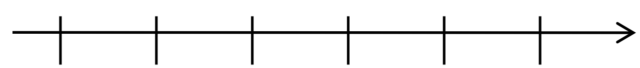
Continuous

Discrete





$x \in R$



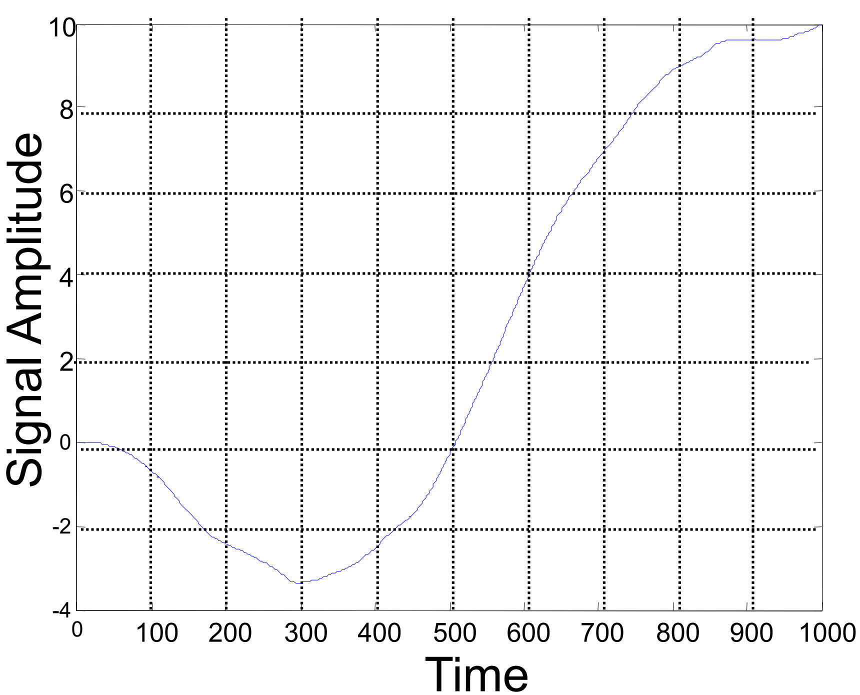
$x \in N$

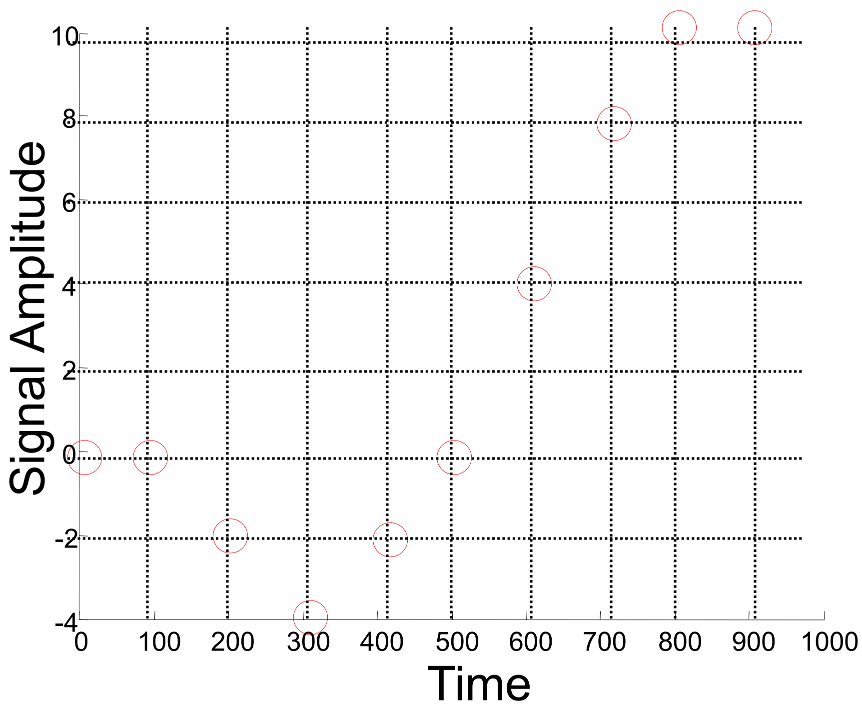
Analog versus **Digital**

Analog versus **Digital**

- An analog signal …
  – is **continuous** in time
  – has **continuous** values
- A digital signal …
  – is **discrete** in time
  – has **discrete** values

Analog – Digital





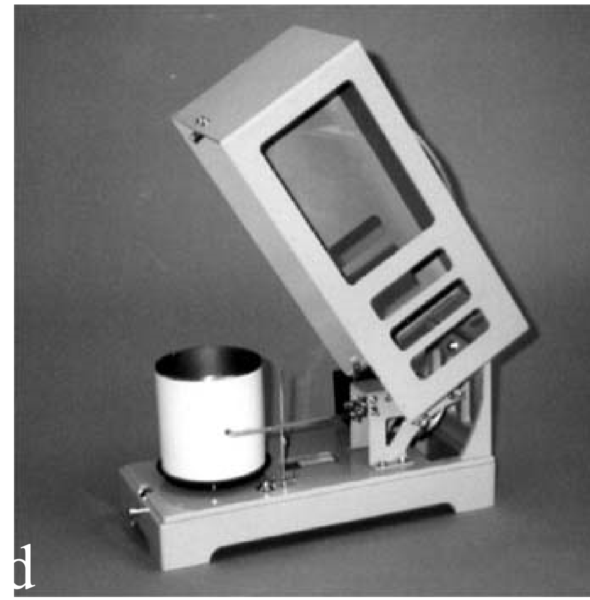
- Analog
  - Continuous
  - Real world
- Digital
  - Discrete
  - Digital world

Sensors Classification Continuous – Discrete

	Continuous	**Discrete**
Time		
Space		
Amplitude		

Continuous: Time and Amplitude

Continuous: Time and Amplitude



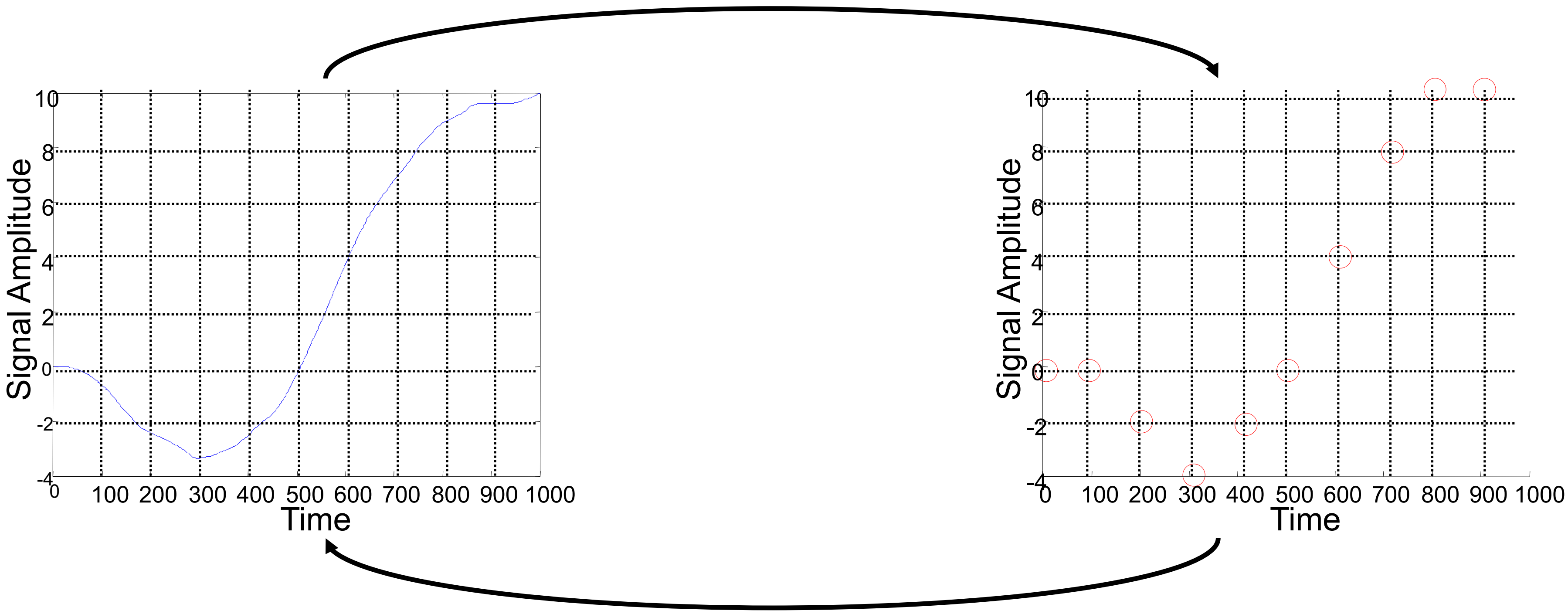
d

Continuous: Space and Amplitude
Discrete: Time

Discrete: Time and Amplitude

Discrete: Time, Space and Amplitude

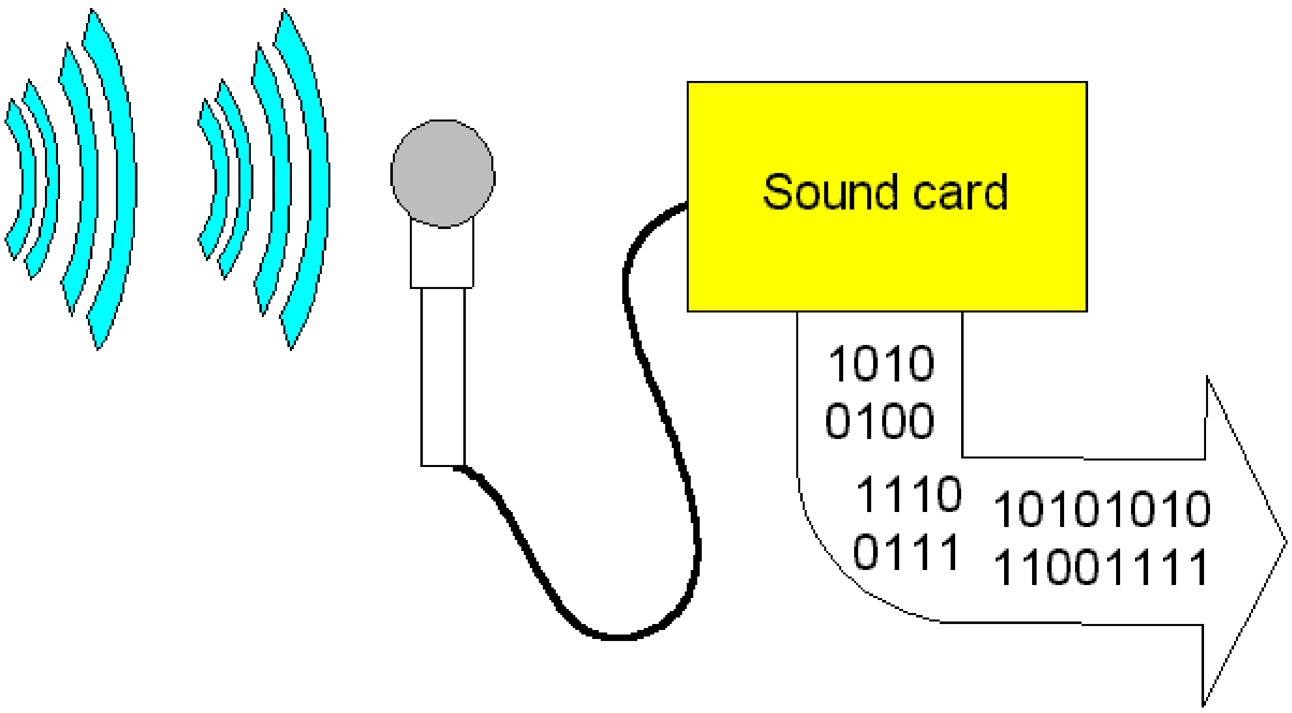
Analog-Digital-Analog Conversion



y=x(t)

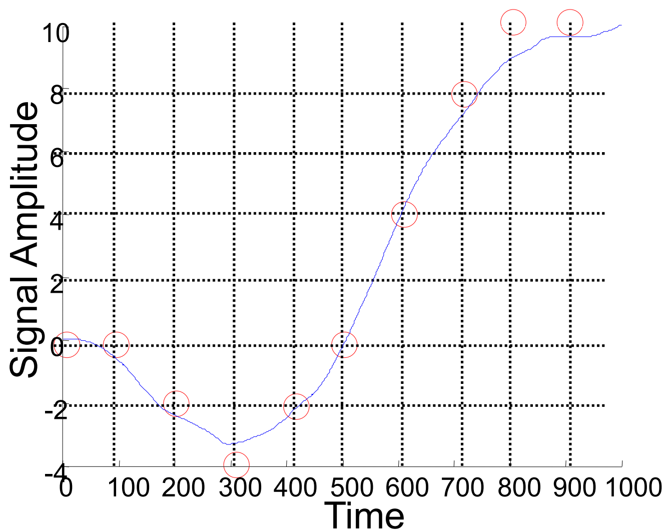
y=x[n]

Analog-Digital Conversion



Analog-Digital Converter (ADC)

- Transforms continuous analog signal into series of values
- Two key elements
  - **Sampling** (in time)
  - **Quantization** (of values)
- Different ADC types:
  - Resolution
  - Speed/frequency
  - Linearity

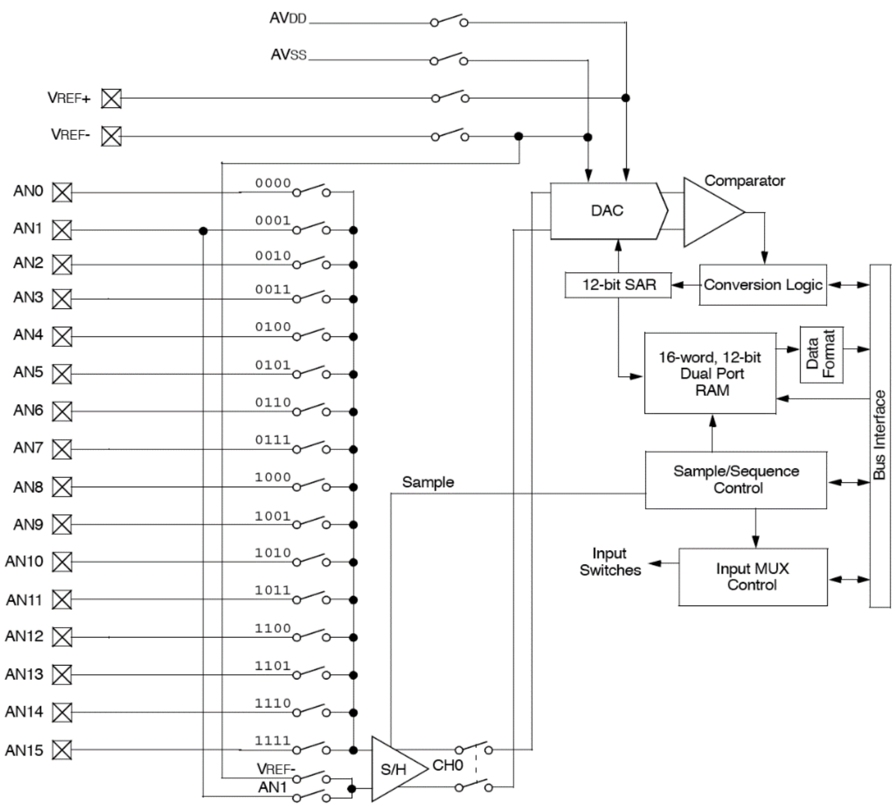


y[n]=0 0 -2 -4 -2 0 4 8 10 10

Examples of ADC Applications

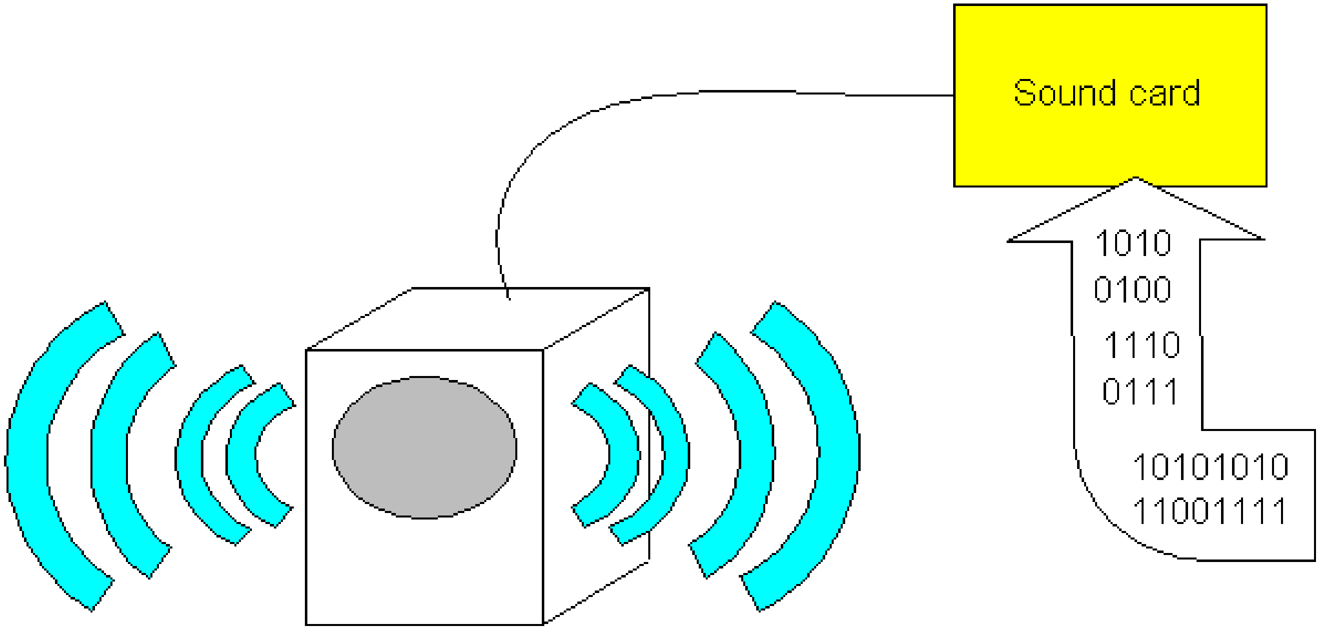
- Input line of a sound card
- Sensor of digital camera
- Mobile phone
- Computer mouse
- …

An Example from an Educational Tool



- From datasheet dsPIC on e-puck: ADC 16 channels, 12 bits, 200 ksps total capacity
- 2^{12} -1 discrete levels of resolution per channel
- Max 200k Hz sampling frequency on 1 channel -> e.g., 10 channels: 20k Hz

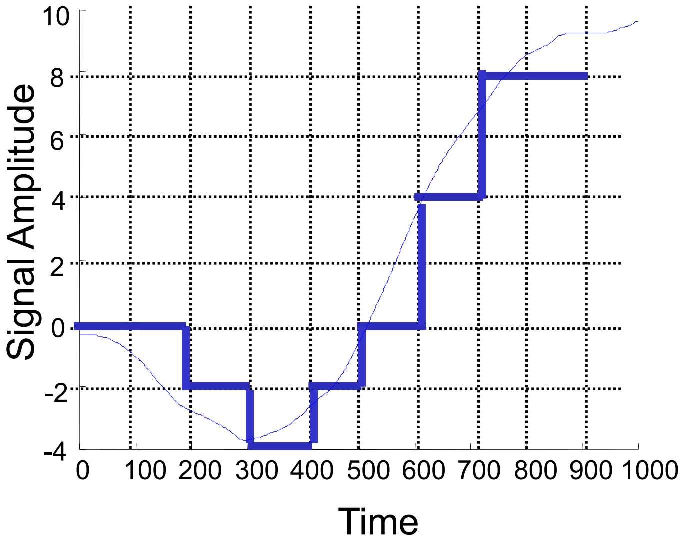
Digital-Analog Conversion



Digital-Analog Converter (DAC)

- Transforms a series of values into a continuous signal
- Multiple types: e.g., a DAC outputs piecewise constant signal
- Additional filter stage to smooth signal (often carried out by the properties of the transducer/actuator itself)

y[n]=0 0 -2 -4 -2 0 4 8 10 10



Examples of DAC Applications

- Mobile phone
- MP3 player
- Graphics card (with older monitors)
- Monitors (newer systems)

Waves as Example of Analog Signals

Waves

"A wave is a disturbance that propagates through space and time, usually with transference of energy."

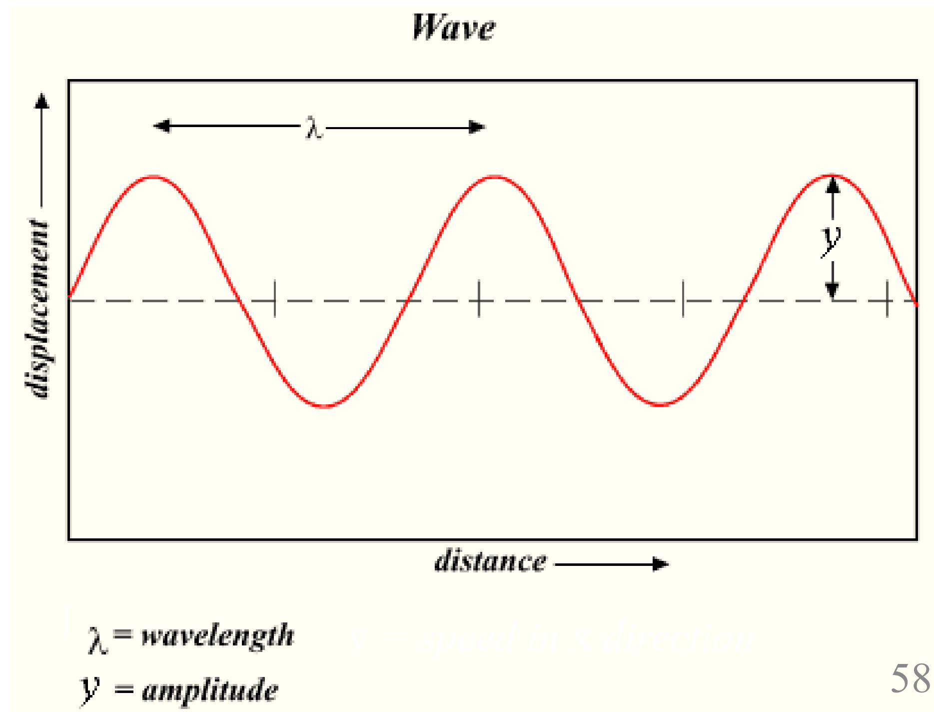
Wave function:

$$f(x,t) = \gamma \sin(kx - \omega t)$$

$$k = \frac{2\pi}{\lambda}$$

$$\omega = 2\pi f$$

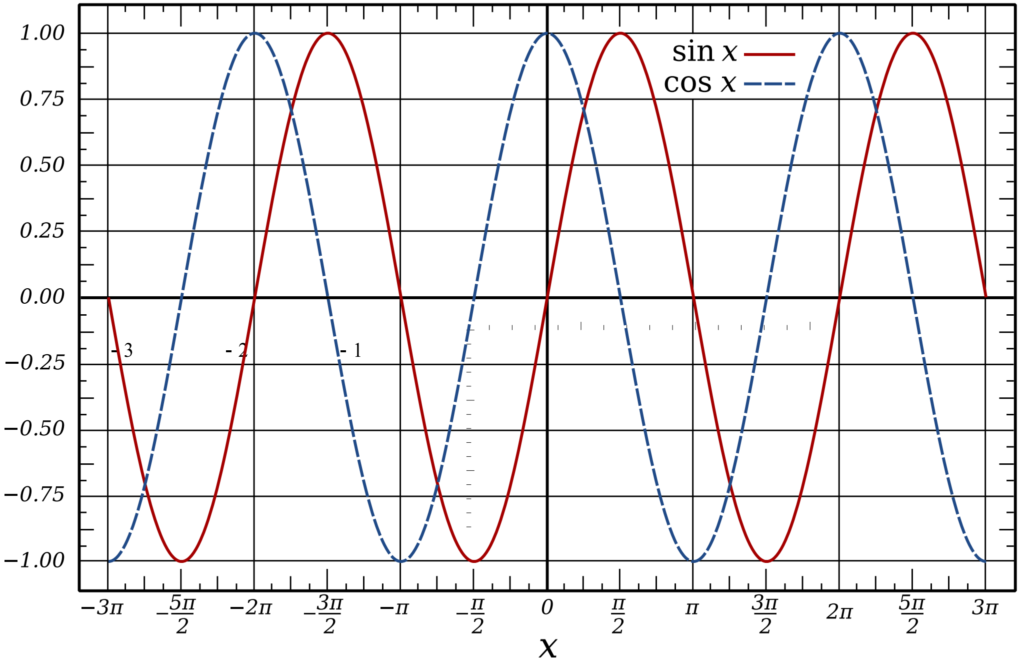
$$\lambda = \frac{\text{v}}{\text{f}}$$



Waves

- **Mechanical** waves: require a physical medium to travel (e.g., sound, water waves)
- **Electromagnetic** waves: can propagate through vacuum (e.g., light, radio waves)
- **Longitudinal** waves: particles move parallel to the wave's direction of motion (e.g., sound waves)
- **Transverse** waves: particles move perpendicular to the wave's direction of motion (e.g., electromagnetic waves)

Sinusoid (Sine Wave) in Time



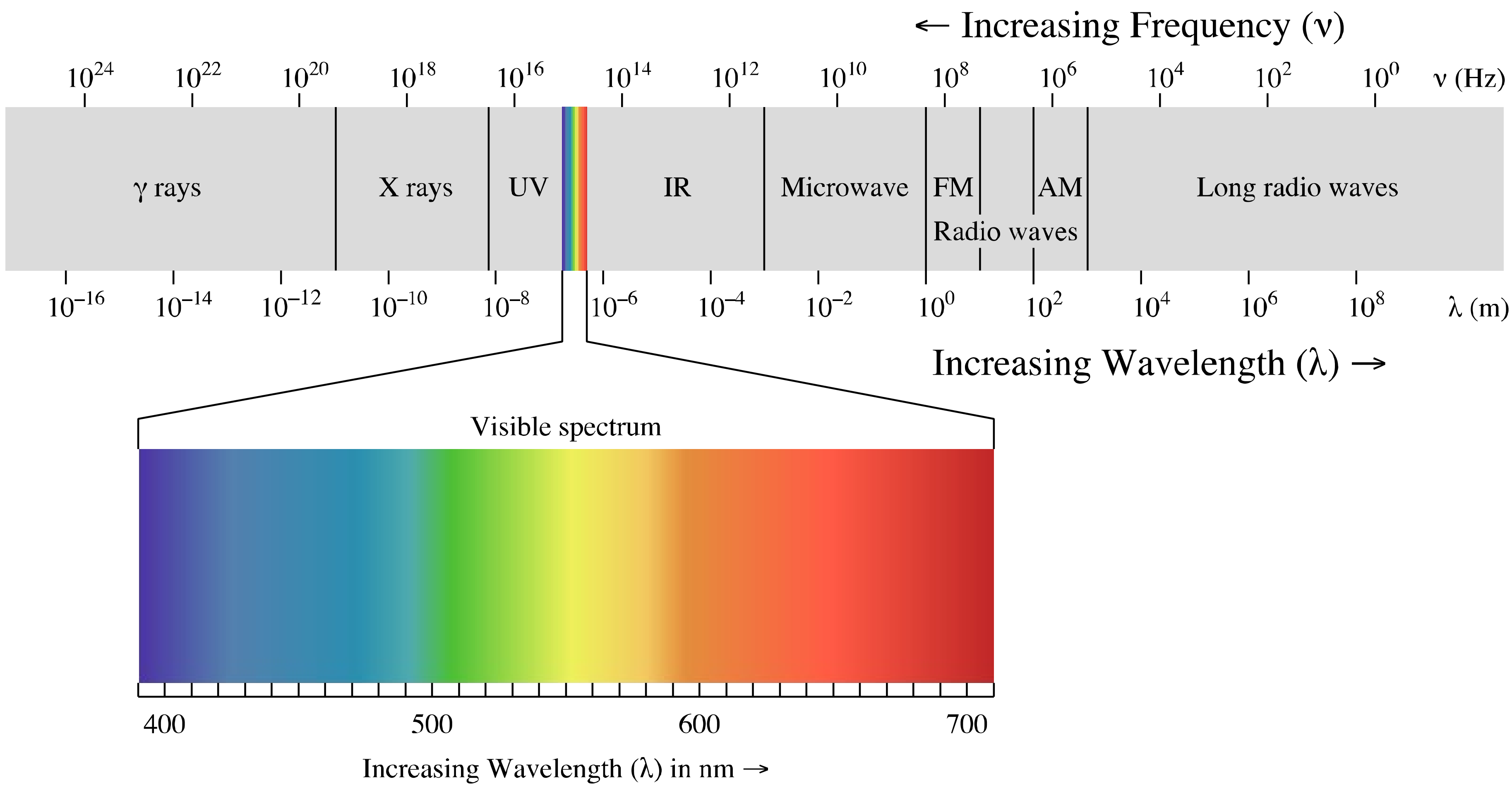
A : amplitude

ω : angular frequency

θ : phase

$$y(t) = A \cdot \sin(\omega t + \theta)$$

Electromagnetic Spectrum



Common Frequencies

• Car motor:	~50 Hz (3000 rpm)
• Power lines:	50 Hz
• Ear:	20 Hz – 20 kHz
• Ultrasound:	20 kHz – 200 MHz
• Medical sonograph:	~2-18 MHz
• Watch quartz:	32 kHz
• FM radio station:	88-108 MHz
• CPU:	2-3 GHz
• 5G cell phone :	0.7/1.4/3.5 GHz
• WiFi :	2.4/5 GHz
• Bluetooth :	2.4 GHz

Conclusion

Take Home Messages

- A signal can be a varying quantity in time and/or space
- Signals can be mono-dimensional or multi-dimensional
- Signal classification:
  - continuous vs. discrete in time
  - continuous vs. discrete in space
  - continuous vs. discrete in amplitude
- Analog signals are continuous in both time (or/and space) and amplitude; digital signals are discrete in both time (or/and space) and amplitude
- Sampling and quantization are two key elements in the transformation of analog into digital signals
- Fundamentals of wave theory and numerical values for common frequencies help in engineering applications

Additional Literature – Week 1

Related course material

- Information, Calcul, Communication (ICC)

Books

- J. H. McClellan, R. W. Schafer, M. A. Yoder
  “DSP First: A Multimedia Approach”, Prentice Hall, 1999.
- A. Oppenheim and A. S. Willsky with S. Nawab, “Signals and Systems”, Prentice Hall, 1997.