Course organization and introduction

Signal Processing Systems Fall 2025

Lecture 1 (Monday 27.10.)

Outline

- Course organization
  - Schedule, materials, assessment, …
- DSP overview
  - Applications, platforms, processing characteristics
- Course contents
  - Perspective on DSP implementation, topics with motivation

https://oulu.zoom.us/j/67624714915?pwd=zCEs66uAZfUyQZgvYZaBeLNadFCaRY.1

Organization

Instructors:

Pekka Sangi, CMVS, pekka.sangi@oulu.fi, TS328 (office time: Thu 9-11)

- lectures, quizzes

Matti Matilainen, CMVS, matti.matilainen@oulu.fi, TS324 (office time: Thu 14-16)

- homework checking

The course is divided roughly to **6 parts**, each taking about one week

Each part has two lecture times (pattern: 1. Thursday 2. Monday)

Assessment is based on the points collected during the course

To pass the course, **one must be active** during the course period

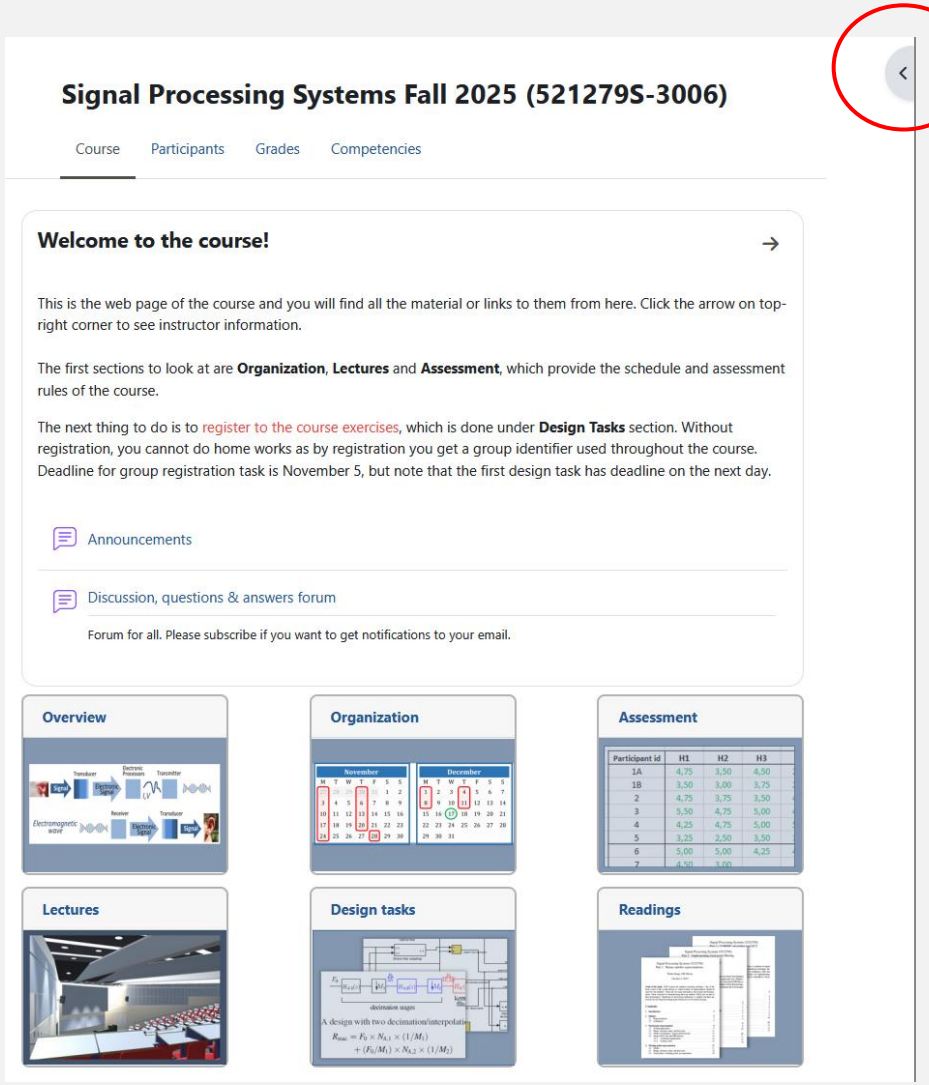
If you need some personal arrangement, contact us

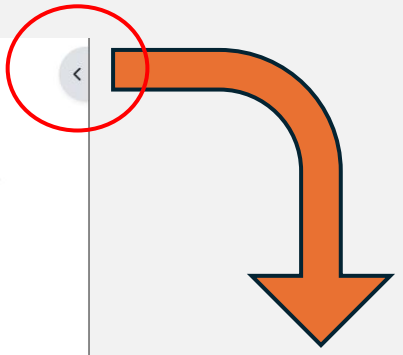
Moodle page

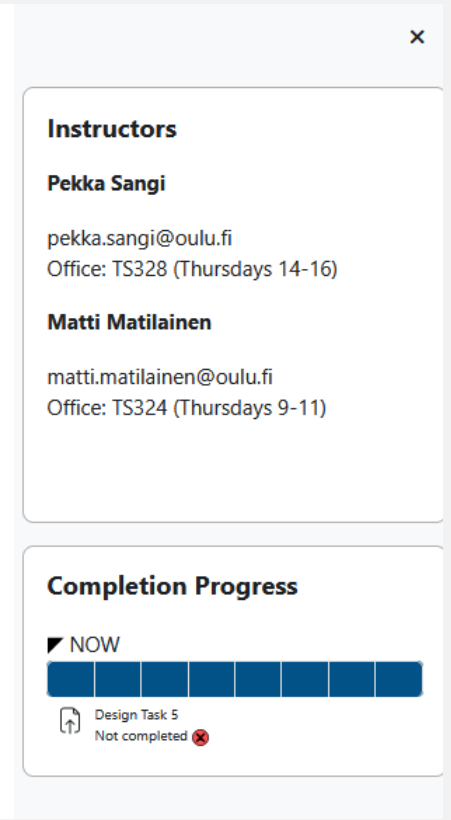
https://moodle.oulu.fi/course/view.php?id=30263

Note possibility to see your completion progress. Eight slots: one for each design task, one for quiz points, and one for extra participation points.

Two forums:
Announcements for messages from teachers (forced subscription – you get email notifications)
Discussion, questions and answers forum (for all, optional subscription)







Schedule

On **Thursdays**, new homework is given and the previous one is returned

On the same day, a quiz is arranged on the homework topics

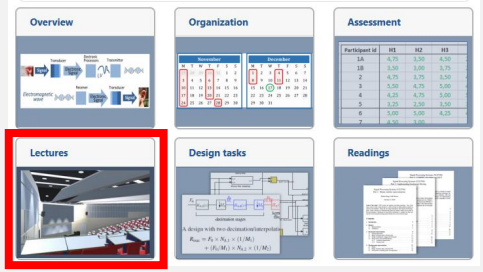
On Mondays, 2nd lecture on current topic + dealing with current design task in the class

On **December 11**, when the Quiz 6 is arranged, it is possible to retake all other quizzes

Day		Activities / milestones
Mon	27.10.	Lecture 1
Thu	30.10.	Lecture 2; Handout of DT1 (08:00)
Mon	3.11.	Lecture 3 + DT1 hands-on
Thu	6.11.	Lecture 4 + **Quiz 1** (after lecture); Deadline of DT1 (23:59); Handout of DT2
Mon	10.11.	Lecture 5 + DT2 hands-on
Thu	13.11.	Lecture 6 + **Quiz 2**; Deadline of DT2; Handout of DT3
Mon	17.11.	Lecture 7 + DT3 hands-on
Thu	20.11.	Lecture 8 + **Quiz 3**; Deadline of DT3; Handout of DT4
Mon	24.11.	Lecture 9 + DT4 hands-on
Fri	28.11.	Lecture 10 + **Quiz 4**; Deadline of DT4; Handout of DT5
Mon	1.12.	Lecture 11 + DT5 hands-on
Thu	4.12.	Lecture 12 + **Quiz 5**; Deadline of DT5; Handout of DT6
Mon	8.12.	Lecture 13 + DT6 hands-on
Thu	11.12.	**Quiz 6 + retake of Quizzes 1-5**
Mon	15.12	Deadline of DT6
Wed	17.12	Assessments and grading done.

DT = design task (homework)
Quiz = small exam related to previous lectures & homework, done in class

See **Lectures** section for information about studied topics, lecture times and places
Note impact of Monday hands-on activities on grading (see **Assessment** section)
Note irregularity of schedule on 28.11.



Lectures

Lectures and quizzes in class.

Slides will be available before the lecture for pre-study (couple of hours before the lecture).

Readings in Moodle will contain additional materials for study.

Lectures can be followed in Zoom.

However, on Mondays only partly as there will be hands-on session related to current homework.

Day		Time	Place	Lecture topics	Hands-on	Quiz
Mon	27.10.	12:15 (2h)	TS101	Course organization & introduction		
Thu	30.10.	10:15 (2h)	L5	Real number formats		
Mon	3.11.	12:15 (2h)	TS101	Arithmetics implementation (incl. NN quantization)	DT1	
Thu	6.11.	10:15 (1h)	L5	Fixed-point processing		Q1
Mon	10.11.	12:15 (2h)	TS101	Error sources in fixed-point filtering; IIR case	DT2	
Thu	13.11.	10:15 (1h)	L5	Introduction to CORDIC algorithm		Q2
Mon	17.11.	12:15 (2h)	TS101	CORDIC and function implementations; DCT	DT3	
Thu	20.11.	10:15 (1h)	L5	Multirate techniques. Decimation and interpolation		Q3
Mon	24.11.	10:15 (2h)	TS101	Multirate filtering, oversampling ADC	DT4	
Fri	28.11.	14:15 (1h)	L8	Filtering in frequency domain. Correlation		Q4
Mon	1.12.	12:15 (2h)	TS101	Filter banks; DFT; Simulink introduction	DT5	
Thu	4.12.	10:15 (1h)	L5	Adaptive filtering		Q5
Mon	8.12.	12:15 (2h)	TS101	Adaptive algorithms	DT6	
Thu	11.12.	10:15 (0h)	L5	Lecture time reserved for the final quiz		Q6, Q1-5

Notes:

Current design task (homework) will be considered during classes on Mondays 3.11.-8.12.

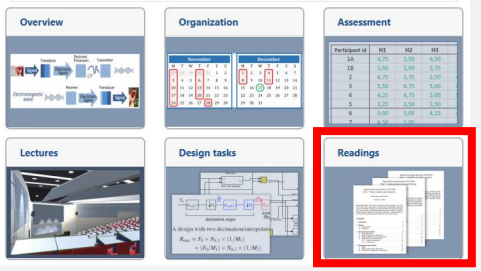
Quizzes are organized on the second half of Thursday classes (6.11.-4.12.)

Last class is reserved for Quiz 6 and retake of quizzes 1-5

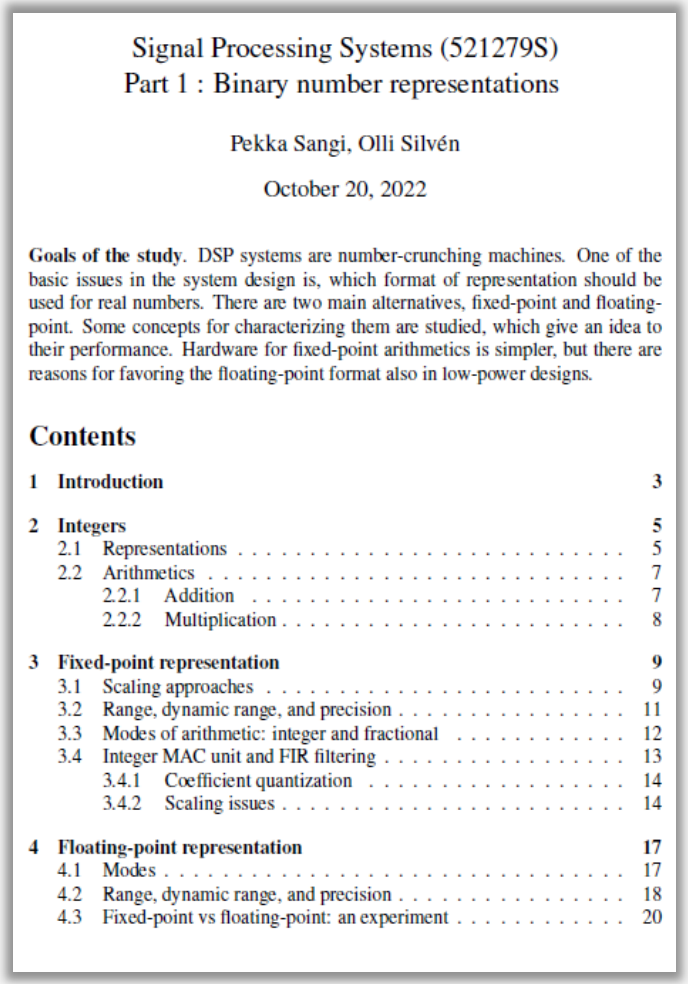
Please note irregularity in time and place on 28.11.!

Zoom link:
https://oulu.zoom.us/j/67624714915?pwd=zCEs66uAZfUyQZgvYZaBeLNadFCaRY.1
Meeting ID: **676 2471 4915** Passcode: **412716**

Readings



- Lecture slides will be put available under Lectures section in Moodle.
- It is recommended to take a look in Readings section too, where materials for self-study will be available.
- Files provide also background for the next design task
- There are two books related to the course.



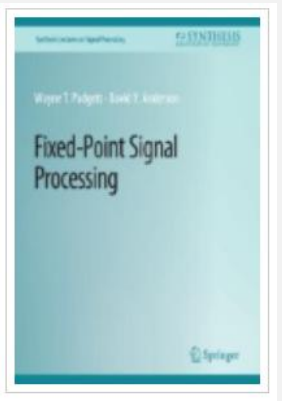
Signal Processing Systems (521279S)
Part 1 : Binary number representations

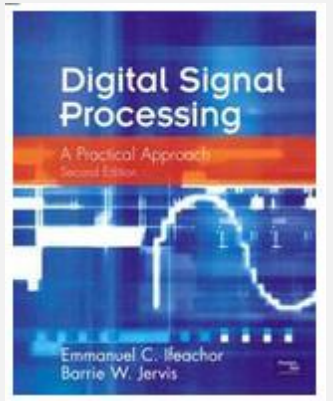
Pekka Sangi, Olli Silvén

October 20, 2022

Goals of the study. DSP systems are number-crunching machines. One of the basic issues in the system design is, which format of representation should be used for real numbers. There are two main alternatives, fixed-point and floating-point. Some concepts for characterizing them are studied, which give an idea to their performance. Hardware for fixed-point arithmetics is simpler, but there are reasons for favoring the floating-point format also in low-power designs.

Contents





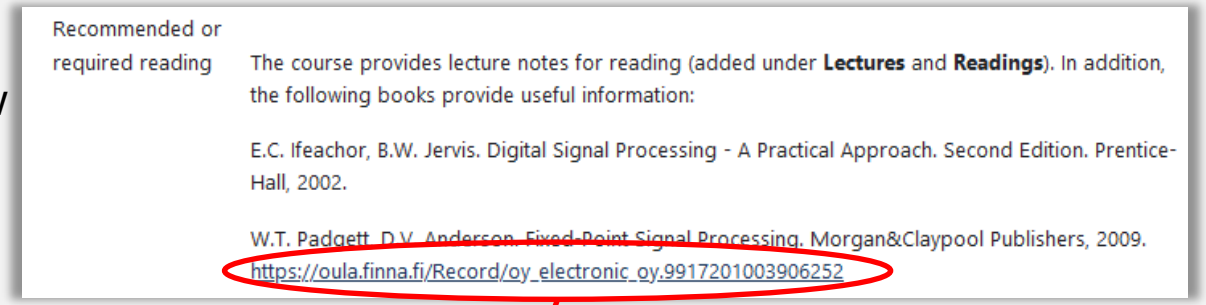
Books:

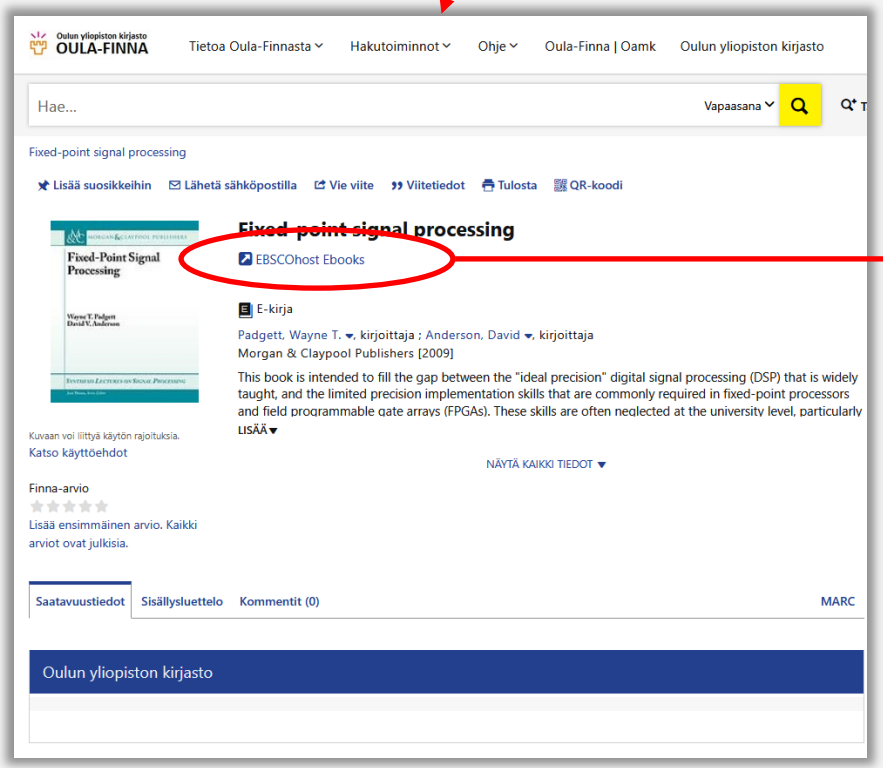
W.T. Padgett, D.V. Anderson. Fixed-Point Signal Processing. Morgan&Claypool Publishers, 2009.

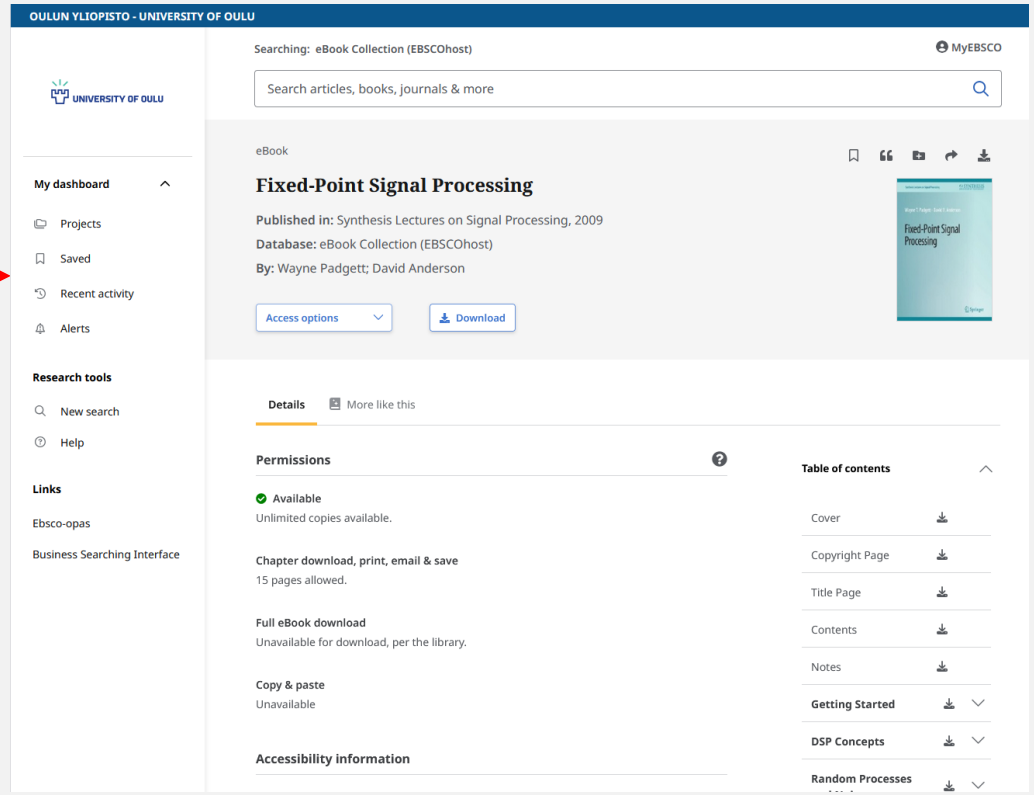
E.C. Ifeachor, B.W. Jervis. Digital Signal Processing - A Practical Approach. Second Edition. Prentice-Hall, 2002.

Link to Padgett's E-book

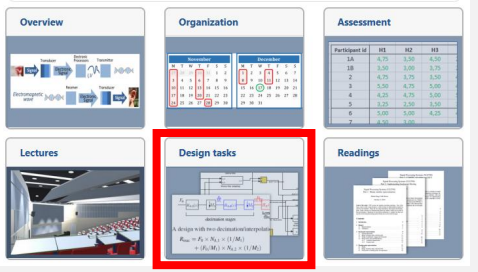
Moodle Overview





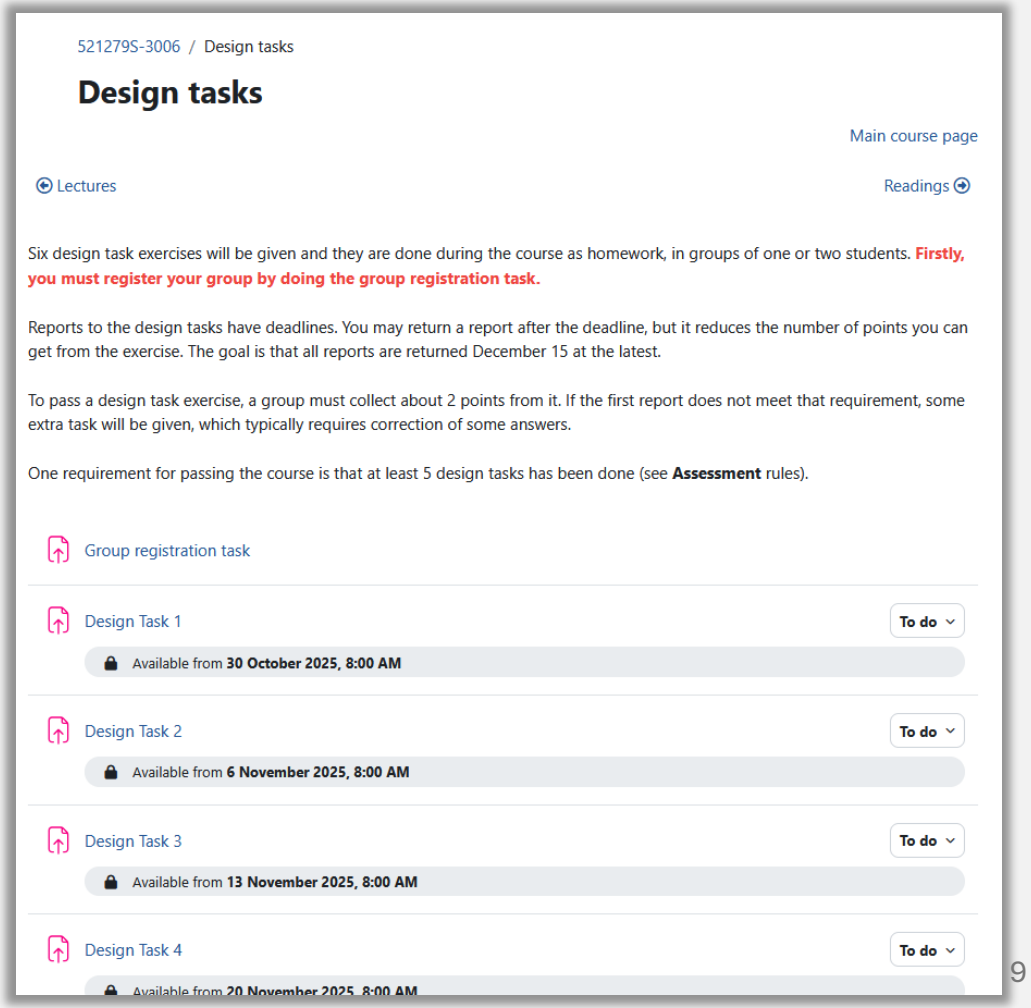


Homework (design tasks)



- Main activity in this course
- Done in groups of 1-2 students
- Handouts given weekly
- Deadlines set so that about one week to complete each work and return a report
- Max. points 4 per task, in total 24
- Requirements:
  - Get about 2 points to pass one
  - Pass at least 5 design tasks
- Some activity in-class on Mondays
  - Hopefully helps in doing the tasks
  - Participation: **0.5 points extra**

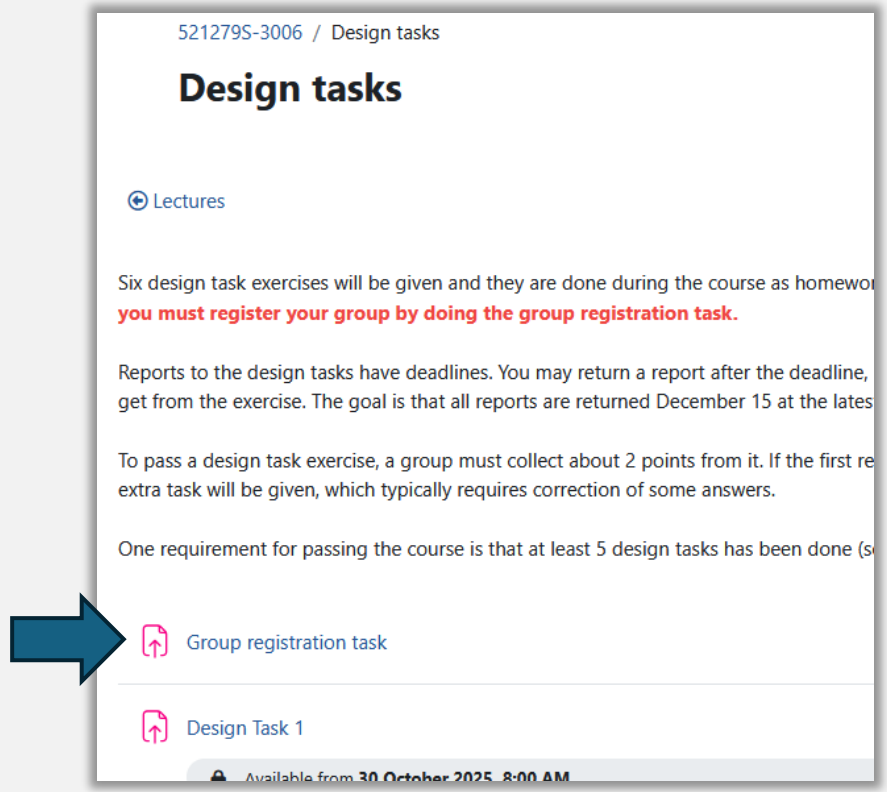
Group must be registered! ➡



Group registration

- 1 or 2 member groups
- Done with a task under Design Tasks section
- You get group and individual identifier, which you use throughout the course
  - Design task assignment parameters depend on the group identifiers
  - Point tables under Assessment use your individual identifiers
- **Registration deadline**: Wednesday, Nov 5

- It is possible to combine single member groups if you find partner later (request by email)
- If you are looking for a group mate, you may use **Discussion, questions & answers forum**



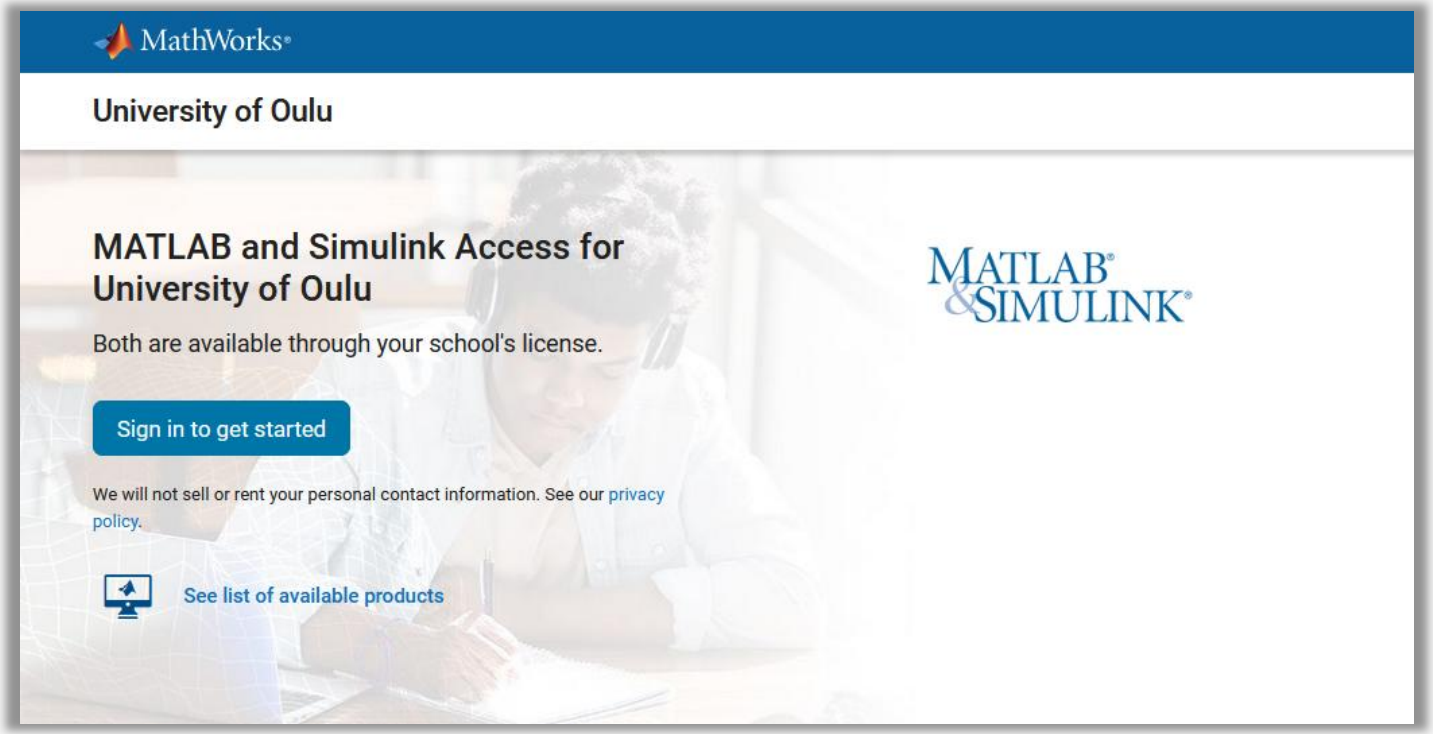
Signal Processing Systems (521279S) updated 11.2.2024

Fall 2023 Points (Design task + Quiz)

Participant	P1	P2	P3	P4	P5	P6	Sum
1	5.4	4.0	5.4	5.2	5.1	5.0	**30.1**
2	3.0	4.5	2.5	4.7	3.5	3.8	**22.0**
3A	5.2	5.6	5.3	5.9	5.3	5.2	**32.5**
3B	4.3	4.9	5.0	4.8	4.8	4.3	**28.1**
4	4.3	4.9	5.7	4.1	3.9	2.9	**25.8**

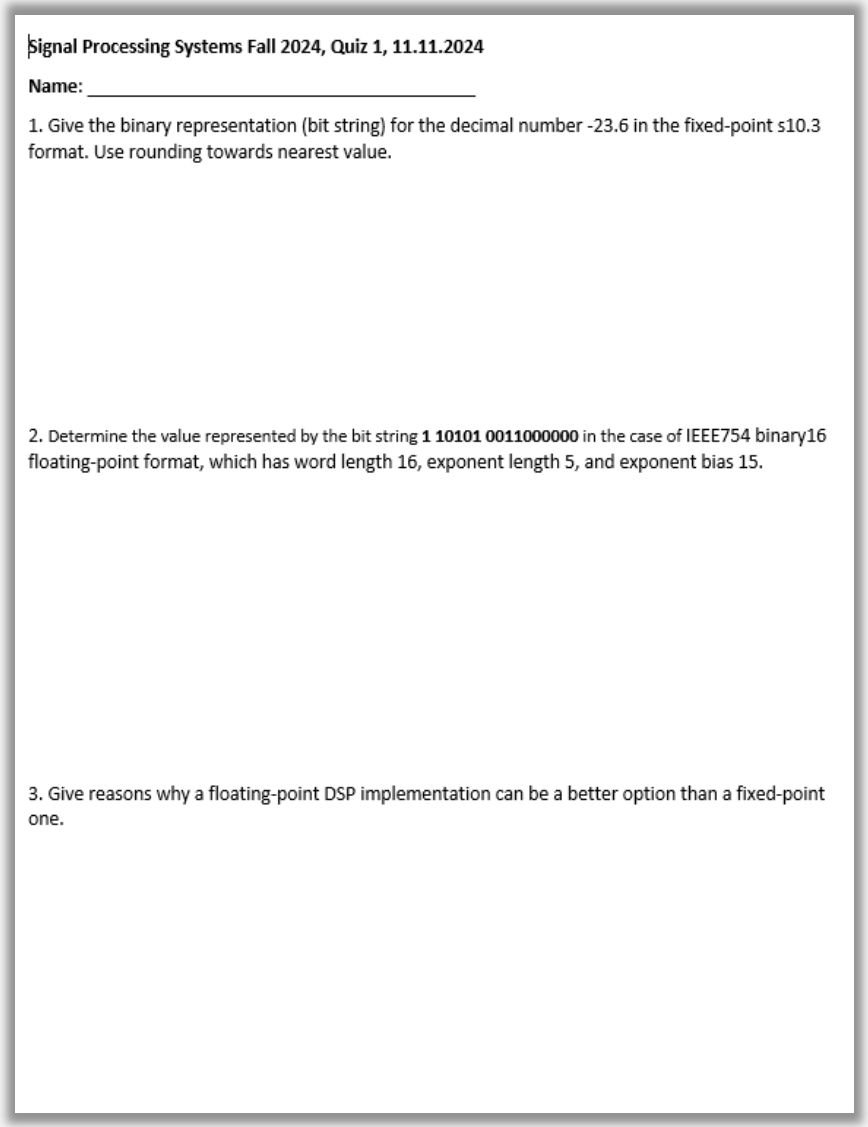
Homework tool: Matlab

- Matlab will be used in many design tasks
- Has support for DSP development in large number of application areas e.g.
  - Fixed-point design tools
  - Block-based DSP modelling and simulation
- University has a campus licence
- Portal for downloading: https://se.mathworks.com/academia/tah-portal/university-of-oulu-873976.html
- Also available in PC classes



Quizzes

- Q1-5
  - In class after a lecture
  - Thursdays 6.11., 13.11, 20.11., 4.12., Friday 28.11.
  - About half an hour time to answer
- Final quiz: Q6 + retake of Q1-5
  - Day 11.12.
  - 1h 45min time to answer
- In each quiz, 2-3 questions related to the topics of lectures and homework
- Max. 2 points per quiz, in total 12
- Sample questions for each quiz will be put available under Lectures section in Moodle



Signal Processing Systems Fall 2024, Quiz 1, 11.11.2024

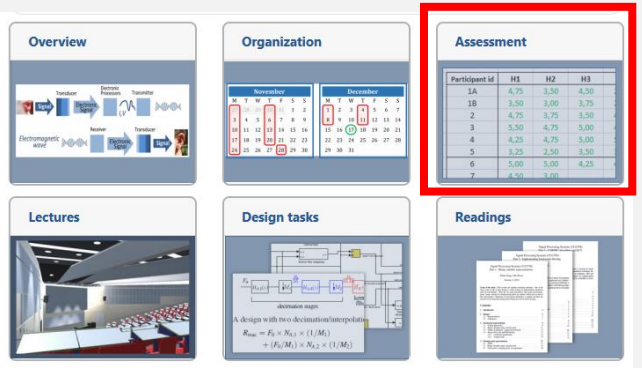
Name: _____

1. Give the binary representation (bit string) for the decimal number -23.6 in the fixed-point s10.3 format. Use rounding towards nearest value.

2. Determine the value represented by the bit string **1 10101 0011000000** in the case of IEEE754 binary16 floating-point format, which has word length 16, exponent length 5, and exponent bias 15.

3. Give reasons why a floating-point DSP implementation can be a better option than a fixed-point one.

Assessment



- Continuous evaluation (no final exam)
  - Can be followed under **Assessment** section (point tables)
- For each course part, you can get max. 6 points
  - Design task homework: 4 points (done in groups of 1-2 persons)
  - Quiz: 2 points (individual points)
- Design task homeworks have deadlines
  - If the report is returned after deadline, max. points may be reduced slightly
  - If your report is not accepted (<2p), you will get some extra work in order to pass
- Maximum points is 6 x (4+2) = 36 points
  - In-class participation on Mondays gives max. 3 points extra (=> max. 39 points)
- To pass the course with grade 1, you need
  1. **To pass at least 5 design tasks**
  2. **To get 3 points from quizzes**
  3. **To get 16 points total (design tasks + quizzes)**
- Grade 5 requires about 31 points

DSP overview

applications, requirements, platforms, characteristics

DSP applications

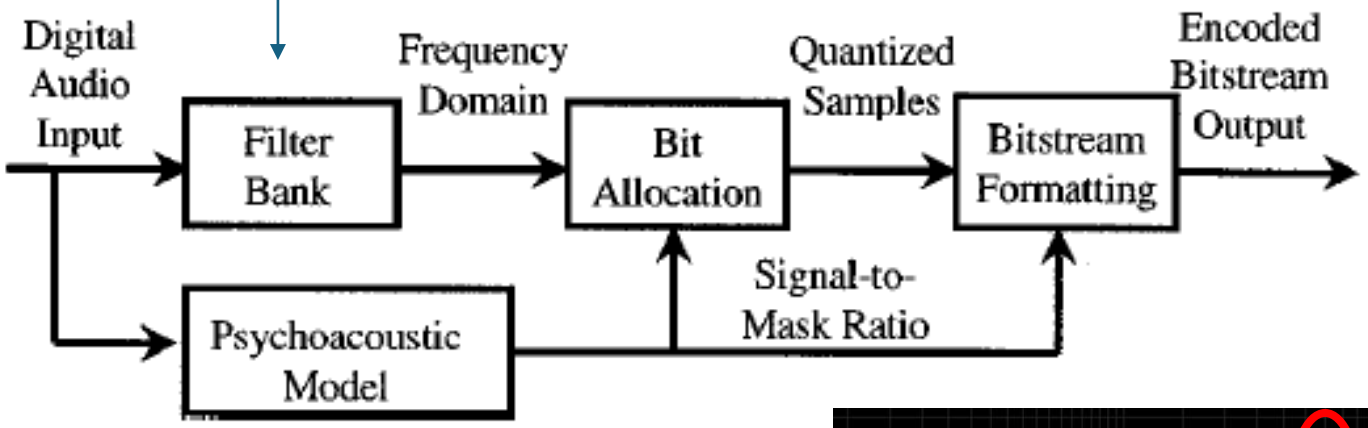
- Digital video encoders and decoders
- Audio encoders and decoders, speech codecs
- Mobile communications baseband implementations, wireless transceivers
- Wireless sensor network (WSN) nodes, Internet-of-Things (IoT)
- Smart cameras
- Biosignal processing
- Medical image processing (e.g., computational tomography, CT)
- Control systems (sensors)

Etc.

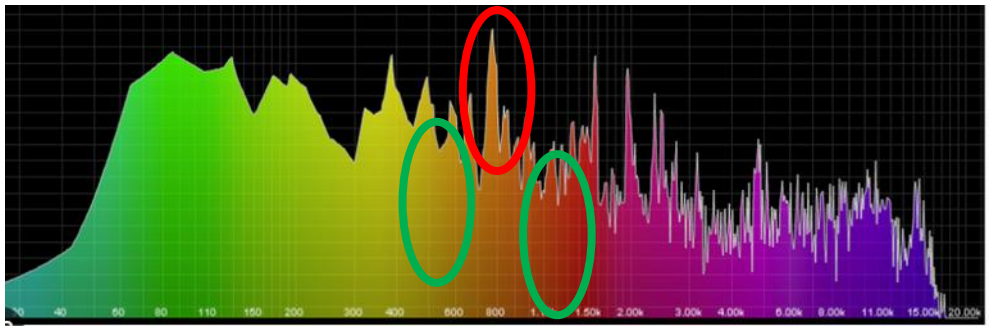
Ex1. Audio codecs

Splitting signal to frequency bands:
- set of bandpass filters
- how to implement?

Encoder



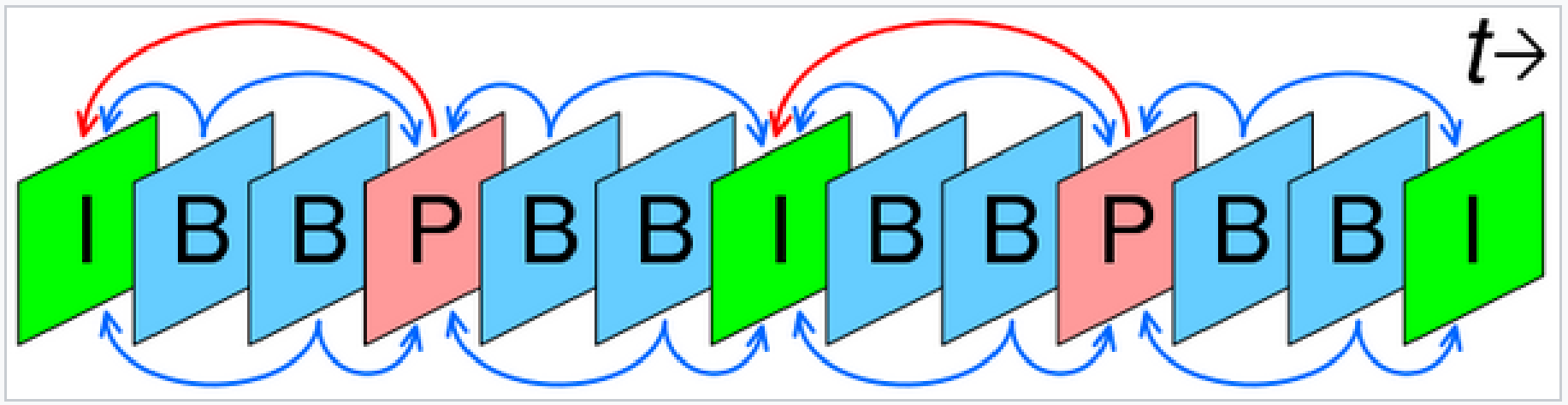
A model of what listener can hear



Ex2. Digital video codecs

I-frames : intra, encoded independently
P-frames encoding: based on I-frames
B-frames encoding: based on I- and P-frames

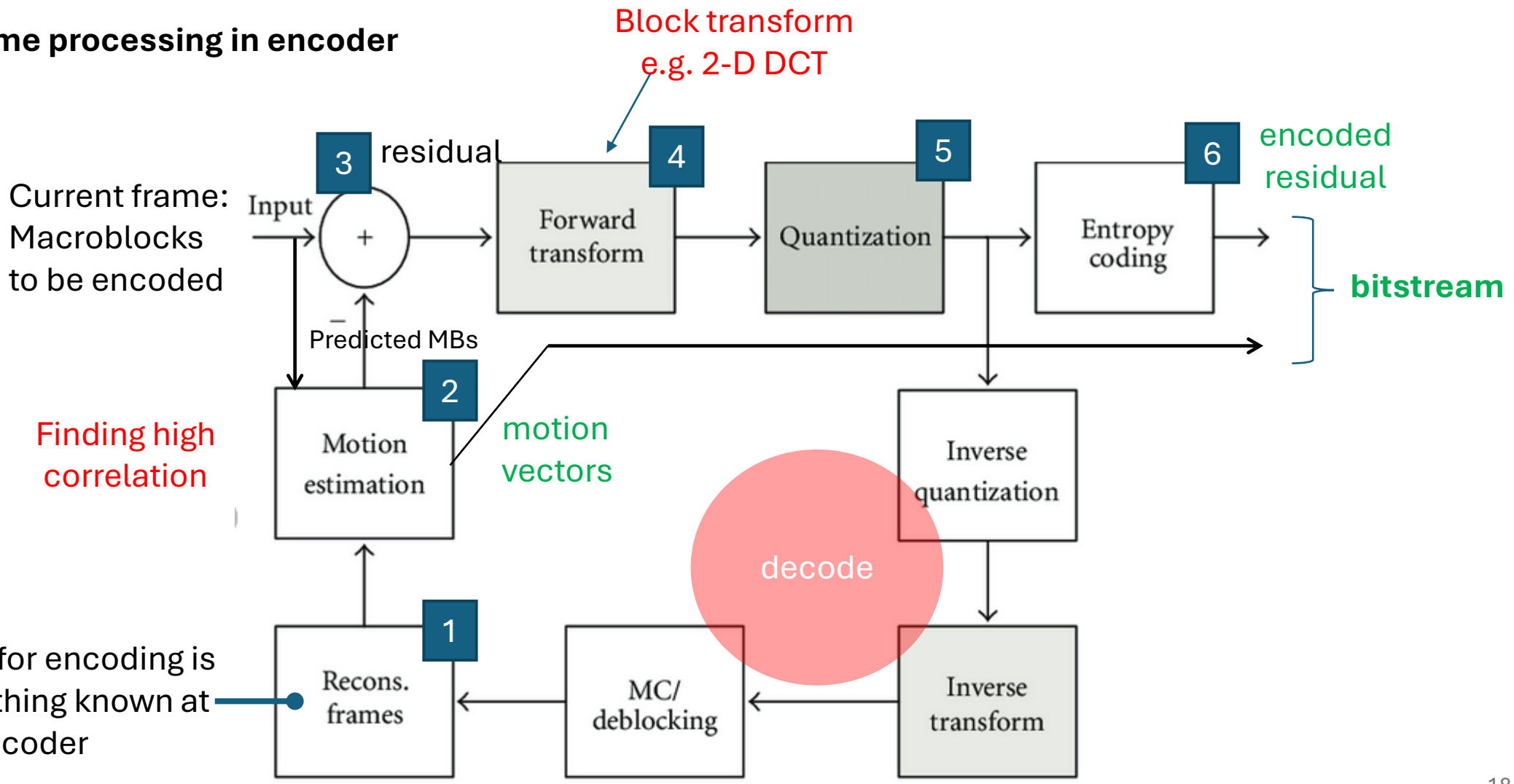
Frame content prediction by utilizing motion estimation and compensation



Frames & their encoding dependences

Ex2. Digital video codecs

P/B-frame processing in encoder



Ex3. Wireless transceivers

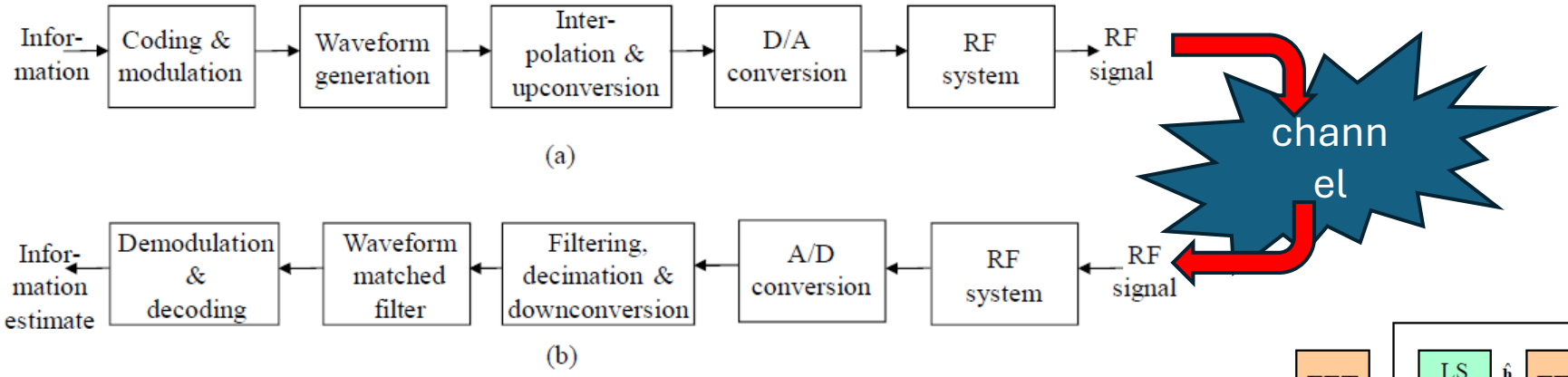
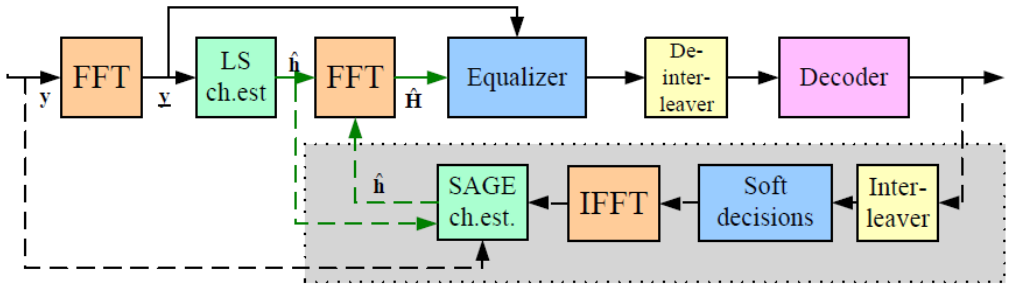


Fig. 1 Simplified wireless transceiver processing chain: (a) transmitter, (b) receiver.

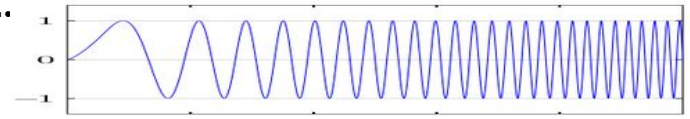


Channel estimation in a MIMO receiver

Features: High data rates
Lots of processing for dealing with radio channel impairments
Adaptive processing

Ex4. FMCW radar

Voltage controlled oscillator (VCO) is used to generate a **chirp signal**.

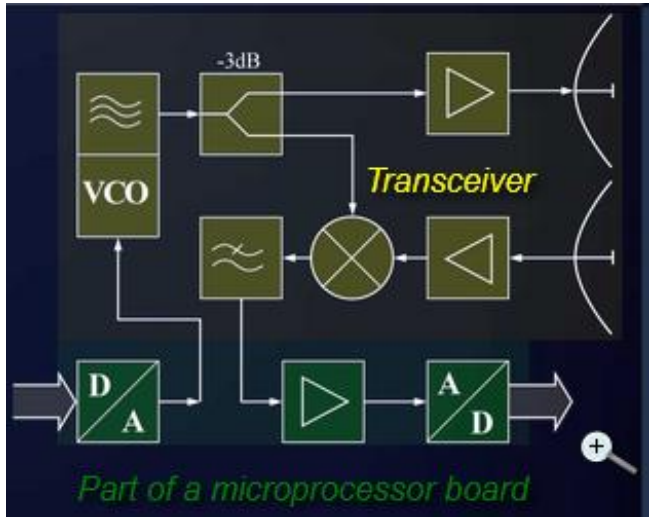


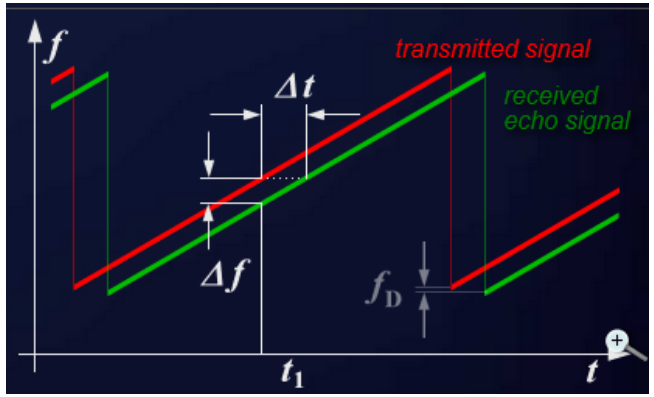
Waveform reflected from environment comes with some time delay.

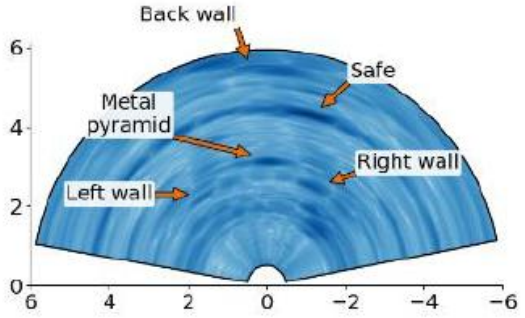
At receiver, mixing with the chirp signal and low-pass filtering produces a signal, which corresponds to **frequency difference Δf of current transmitted and received chirp**.

Analysis of this signal with discrete Fourier transform (DFT) based techniques => find out what Δf is.

More advanced approach combines radar with communications e.g. for automotive applications: joint communication and sensing (JC&S, ISAC)







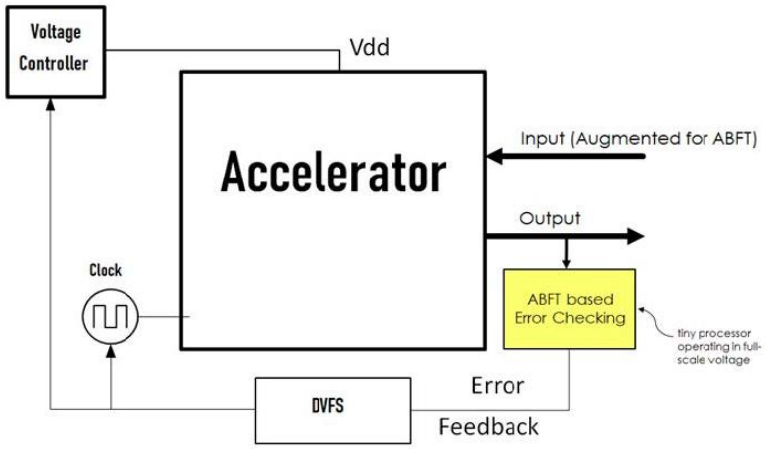
https://www.radartutorial.eu/02.basics/Frequency%20Modulated%20Continuous%20Wave%20Radar.en.html

Ex5. Wireless sensor nodes (WSN)

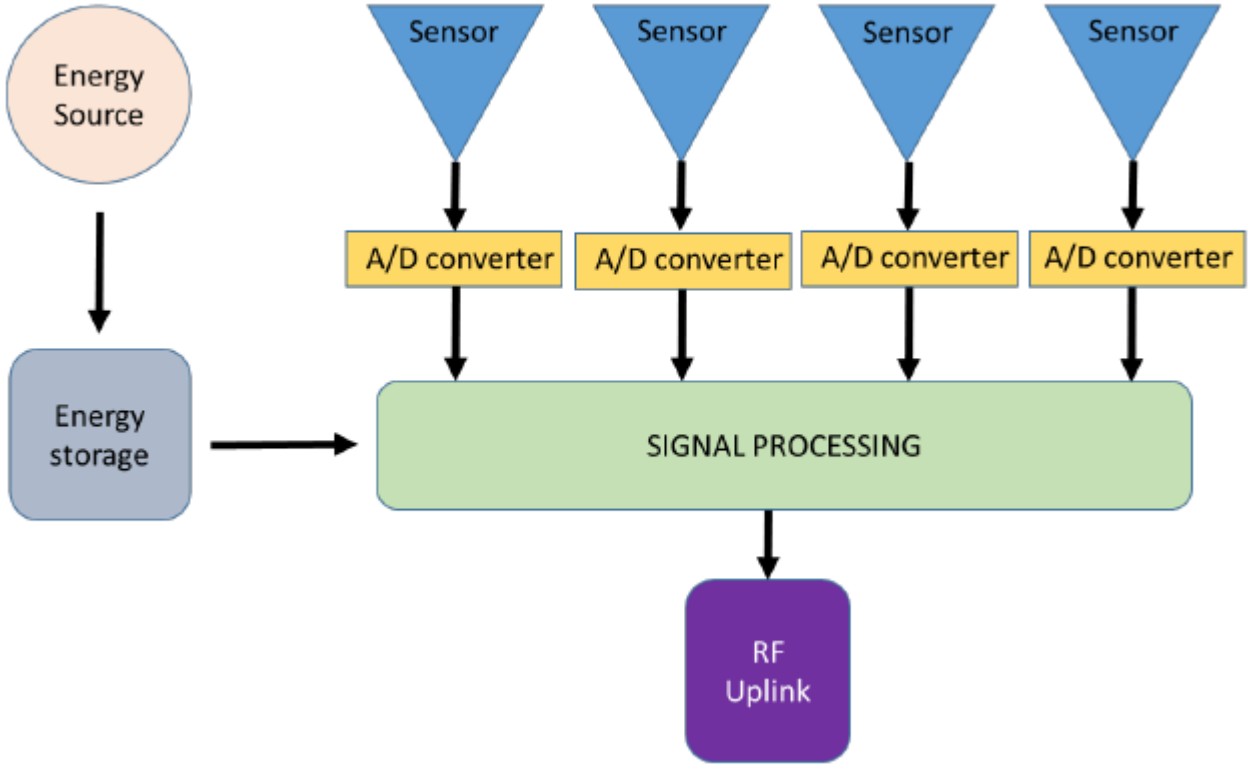
Battery vs harvesting energy

Tradeoff: local processing
vs. communication

Energy saving e.g. by control of operating voltages
– seek for a value where the processing is still without errors



Mehdi Safarpour, PhD thesis 2021

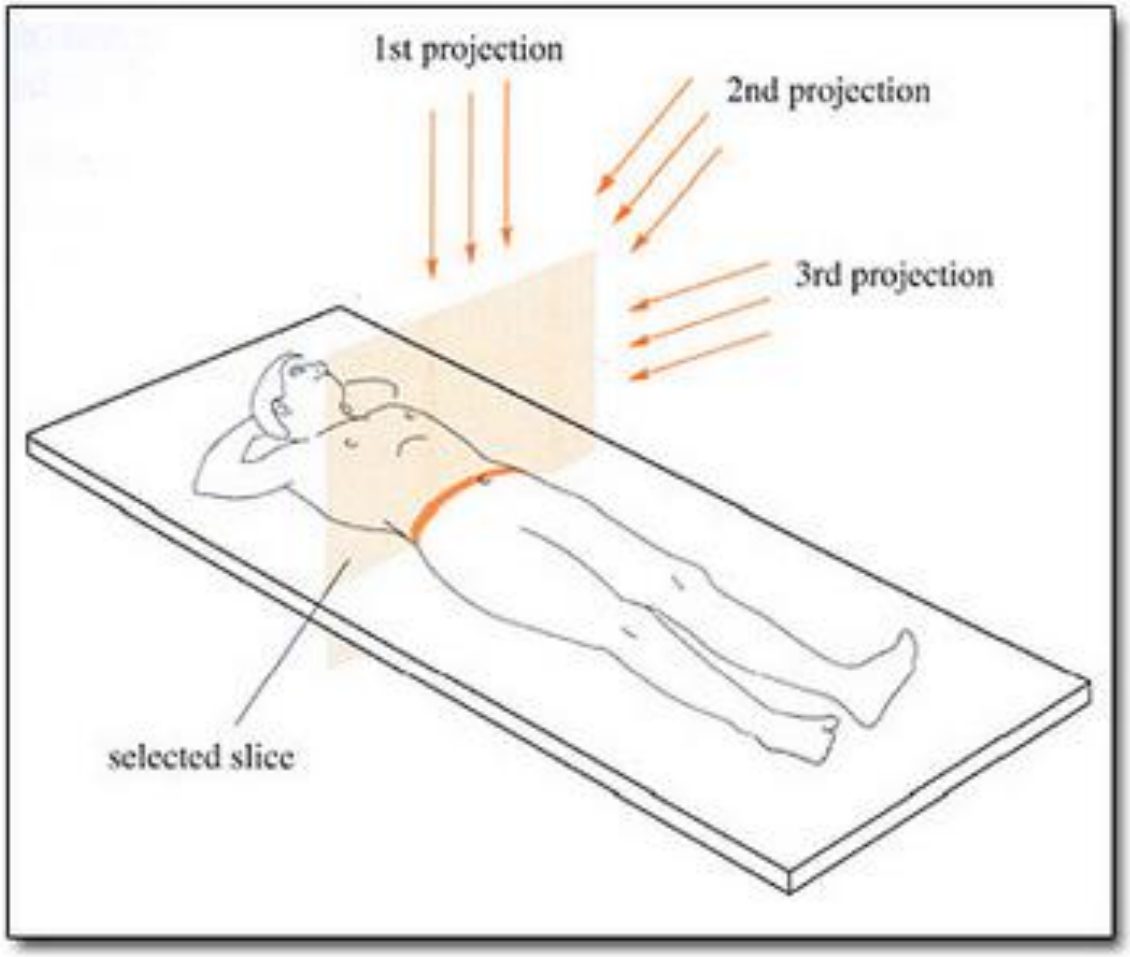


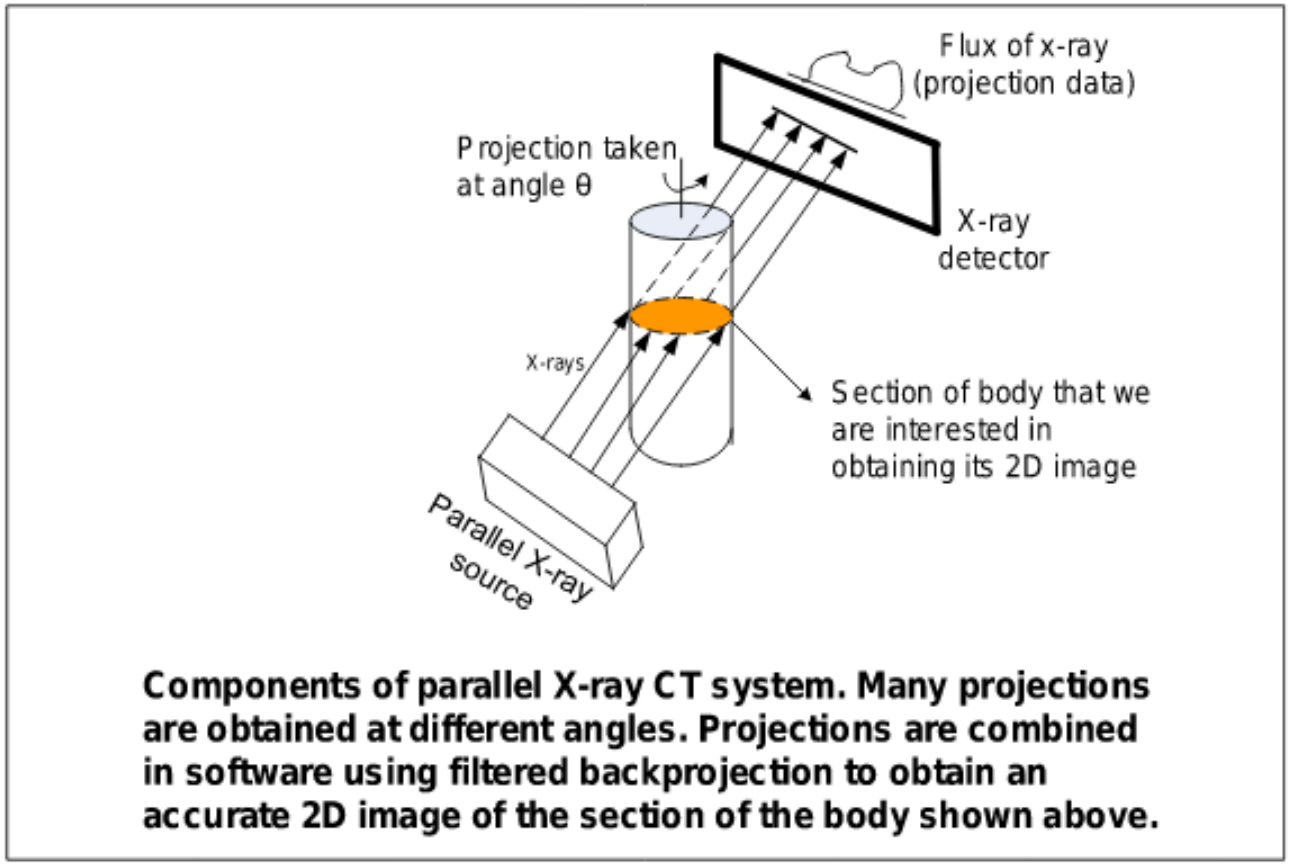
Ex6. Smart cameras

- Extracting **features**
- Detecting regions of interest
- Suppression of disturbances e.g. variations due to lighting
- Robust feature design, application of machine learning
- Distributed smart cameras: computation/transmission tradeoff like in the case of WSN
- Well-known real-time SW: YOLO, see https://docs.ultralytics.com/

Wikipedia: Signal processing is an electrical engineering subfield that focuses on analysing, modifying, and synthesizing signals such as sound, images, and scientific measurements.[1] Signal processing techniques can be used to improve transmission, storage efficiency and subjective quality and to also emphasize or detect components of interest in a measured signal.

Ex7. Computed Tomography (CT)



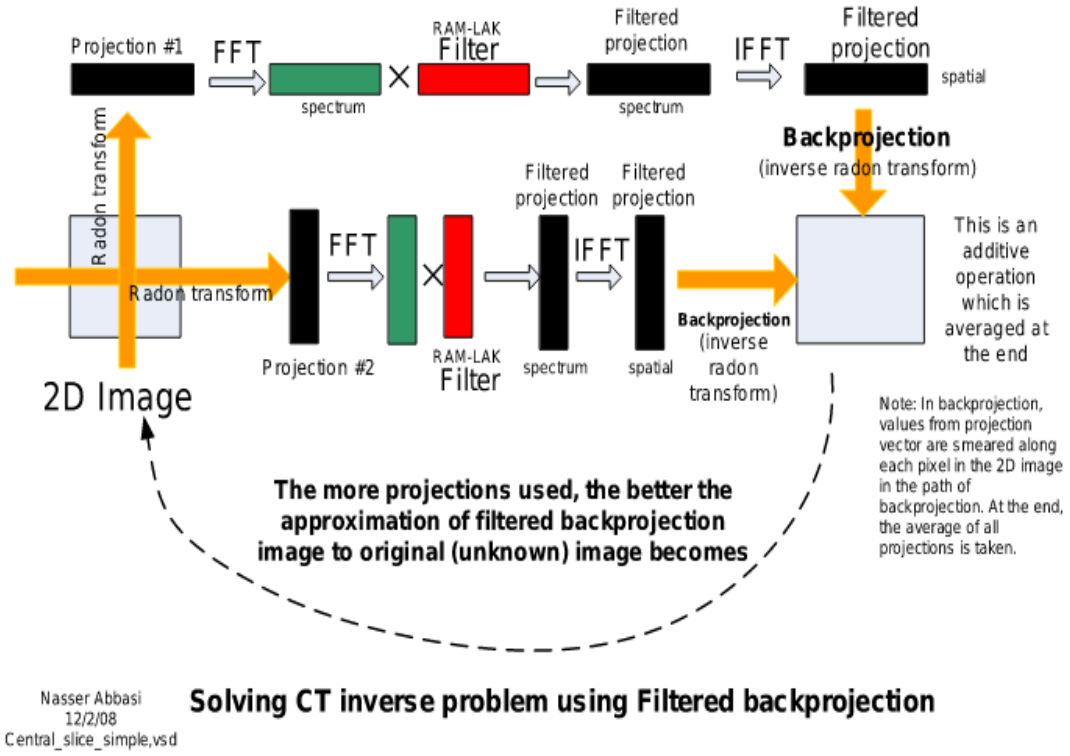


Components of parallel X-ray CT system. Many projections are obtained at different angles. Projections are combined in software using filtered backprojection to obtain an accurate 2D image of the section of the body shown above.

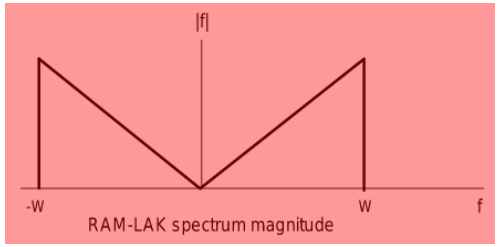
https://www.12000.org/my_notes/EE518_CT_project/REPORT/index.htm

Ex7. Computed Tomography (CT)

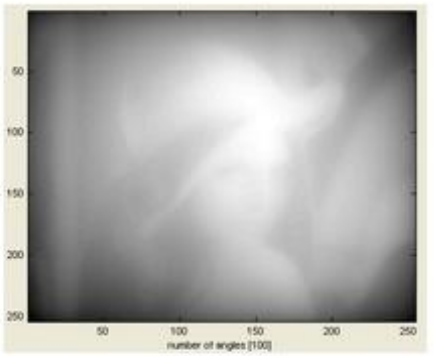
$$h(x,y)=\frac{2W\pi}{MN}\sum_{m=0}^{M-1}\left[\sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1}S\left(n\frac{2W}{N},m\frac{\pi}{M}\right)e^{nj\frac{4W\pi}{N}\left(x\cos\left(m\frac{\pi}{M}\right)+y\sin\left(m\frac{\pi}{M}\right)\right)}\left|n\frac{2W}{N}\right|\right]$$



Original Image



Reconstruction, 100 projections, Filtered backprojection (RAM-LAK filter)



Reconstruction, 100 projections, No filtering used before backprojection

Products from Oulu (1)

All requiring some forms of signal processing

Energy harvesting sensor node (Solmu)

Thingsee One IoT platform (Haltian)

Counter-noise earplugs (QuietOn)

Kognitive short-wave radio network (Kyynel)

Video content search engine (Valossa Labs)

Diagnostic viewer (Neagen)

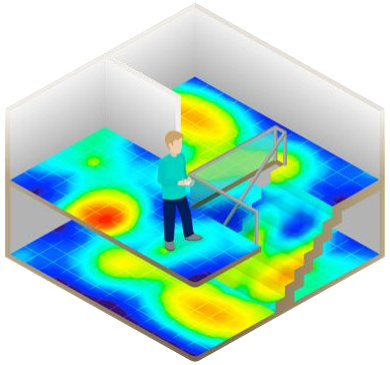
Opthalmologic camera (Optomed)

Imaging spectrograph (Specim)

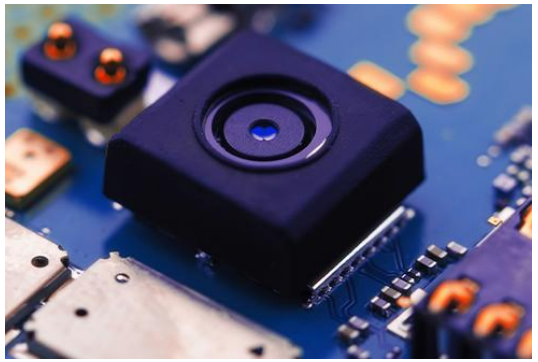
Products from Oulu (2)

Heart rate meter (Polar)

Intelligent shoe sole (MoveSole)

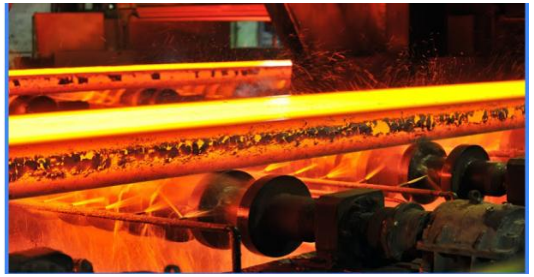


Magnetic field navigation (IndoorAtlas)



Image/video processing, computed vision (Visidon)

Optical 3-D scanner (FocalSpec)



3-D measurements for metal Industry (Sapotech)

Ultra-low energy data acquisition device (Offcode)

ŌURA

Smart rings for health monitoring

And more: NORDIC SEMICONDUCTOR, SPREADTRUM, ROHM SEMICONDUCTOR, ARM, MEDIATEK, NOKIA, ...

Do you have any specific interest? Any experience?

- Go to vote at

  https://presemo.oulu.fi/spslec1

Requirements

GOPS = gigaoperations per second

- Complexity tends to increase in new technology generations, e.g.
  - higher throughputs in communications
  - higher qualities in video processing
- This is visible also in current drivers of change in digitalization:
  - **artificial intelligence**: machine learning, computer vision, speech recognition, affective computing, robotics, perceptual interfaces, etc.
  - **5G and toward 6G**: massive wireless system ultra-densification, device-to-device communications, ultra reliable communications, sensing: communicate-to-view
  - **IoT**: internet connected physical devices, vehicles, etc. with embedded sensors, actuators, electronics, and software.

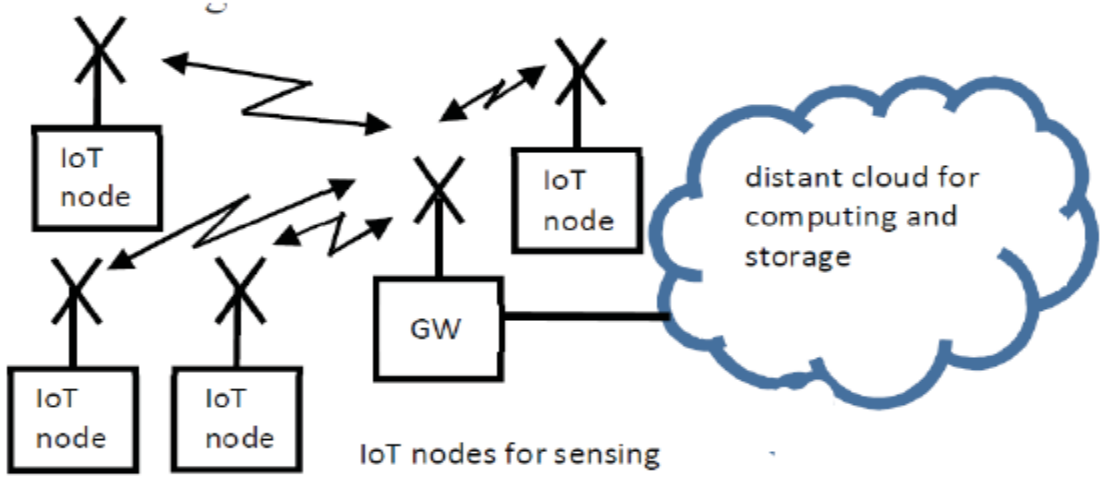
Channel type	Data rate	GOPS needs
Release 99 DCH channel	0.384 Mbit/s	1-2
Release 5 HSDPA channel	14.4 Mbit/s	35-40
OFDM channel	100 Mbit/s	210-290

3GPP receiver requirements for different channels

Video standard	operations/pixel	Processing speed (GOPS)
MPEG-4	600-900	2-3
H.264-AVC	600-900	6-10
H.265-HEVC	2000-3000	20-30

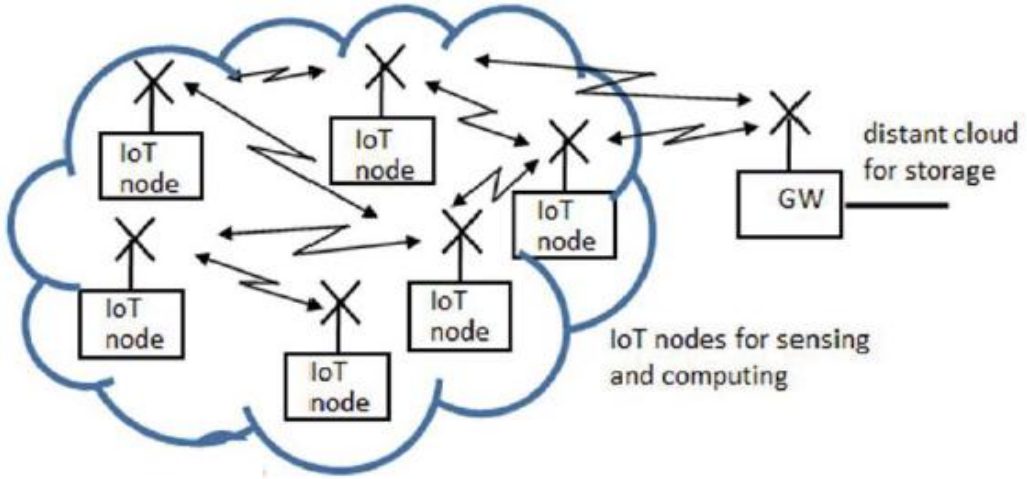
Computing needs of video **encoders** for VGA (640x480) video (decoding much cheaper)

Trend in IoT systems



From centralized processing …

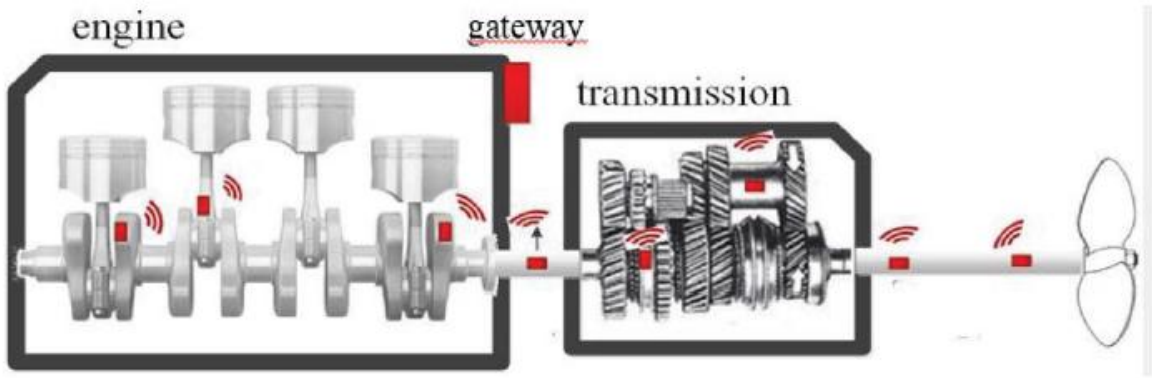
Nodes act merely as data sources and sinks. The cloud does the big data computing.



… to distributed processing

Edge computing: More and more advanced modalities such as imaging, audio, and RF based sensing. Local processing may require exchange of information between nodes.

Distributed sensor network



An industrial diesel engine designed through heavy modeling and model-based simulations. Need to **verify** that the simulations and reality match. Also **condition monitoring** needed.

Therefore, simultaneous strain, temperature, and acceleration measurements are collected using miniature wirelessly communicating self-powered devices.

Alternative A: just collect data and forward it for analysis by communication? Realistic?

Alternative B: local signal processing to compute essential features + their analysis. Then less communication => probably more energy-efficient, feasible solution

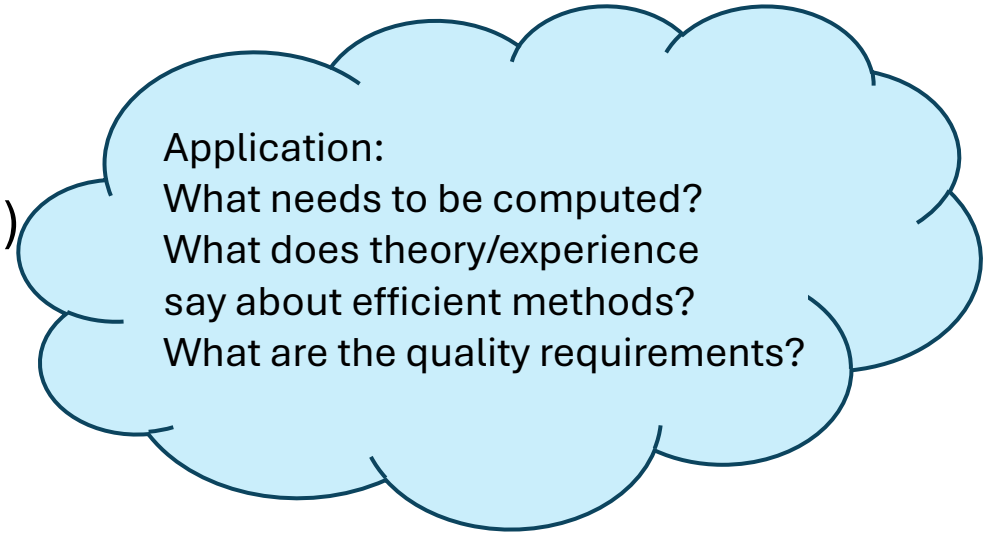
For efficient implementation, basic know-how for optimizing needed

Typical DSP requirements

- low power consumption
  - important in battery-powered/energy harvesting devices
  - also related to management of thermal dissipation
- good silicon efficiency (e.g. low number of logic ports)
- high throughput
- low latency
- low design and manufacturing costs
- flexibility needs (programmability)

- Also, how to develop systems quickly in order to meet the market demands?

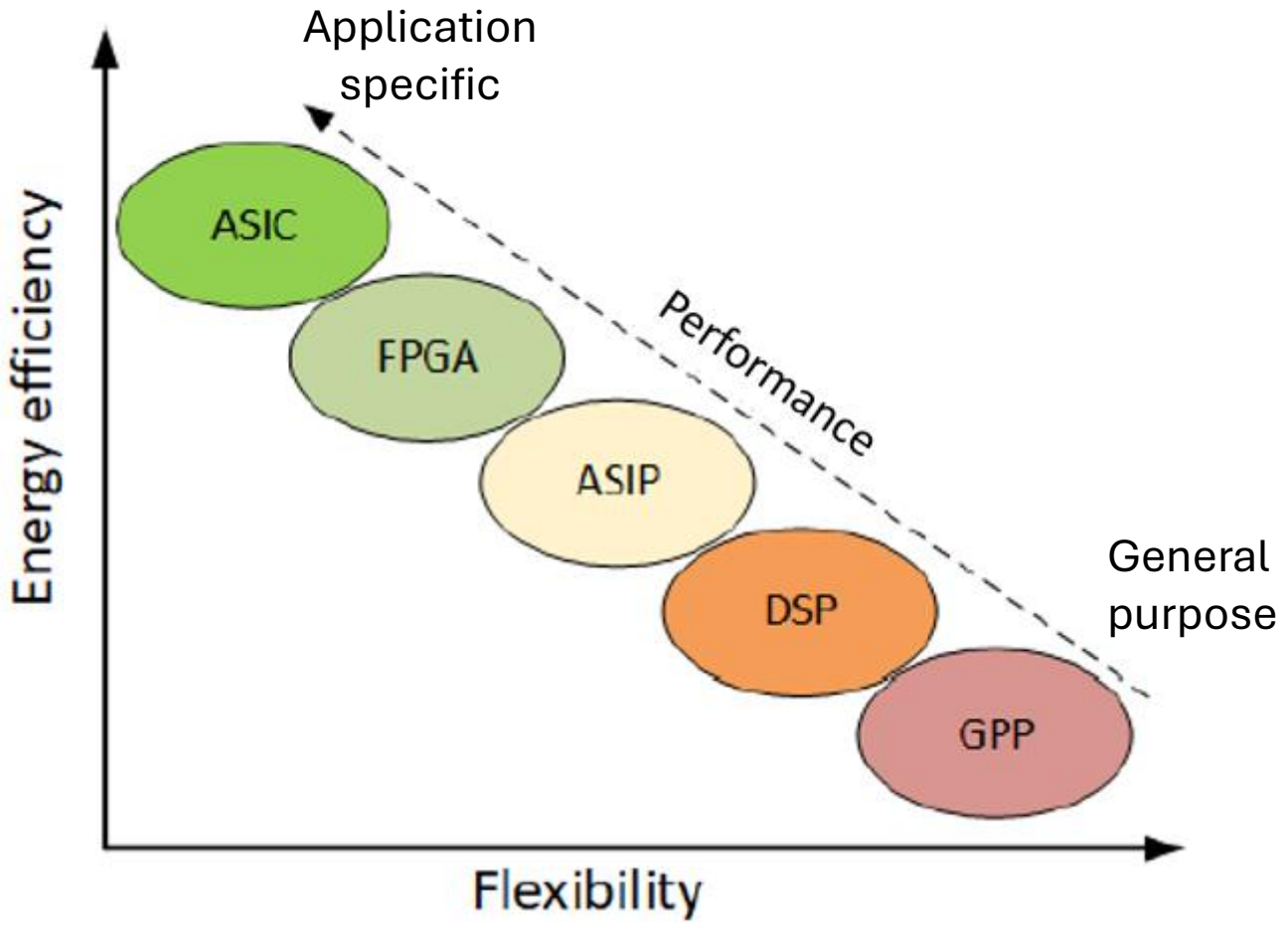
- Choice of **implementation platform** is one of the first issues here



Platforms for signal processing

- DSP algorithms are run on different platforms like
  - general purpose microprocessors (GPP)
  - microcontroller units (MCU) with signal processing accelerators
  - general purpose graphics processing units (GPGPU)
  - digital signal processors (DSP)
  - application specific integrated circuits (ASIC)
  - field programmable gate arrays (FPGA)
- In research, attention is paid on application specific (signal) processors (ASP, ASSP, ASIP)
  - Requires advances in development methods and good tools
  - Synthesis of application development and architecture specification
  - Hardware-software codesign

Platform comparison



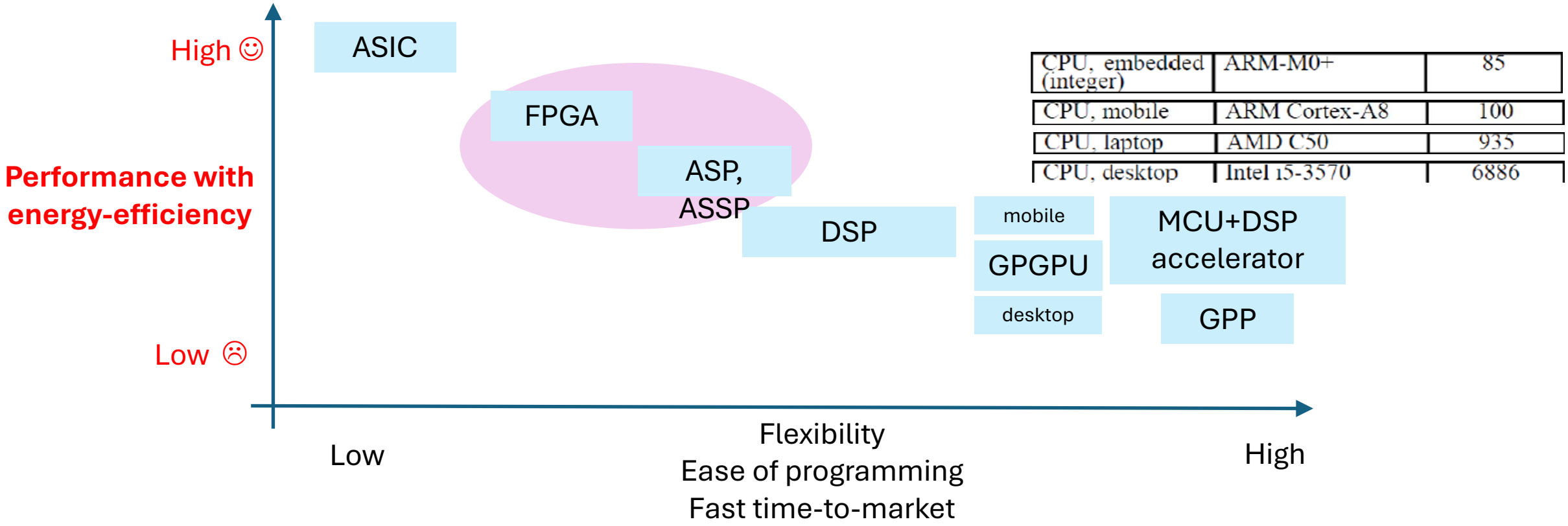
Platform comparison / EPI

Energy Per Instruction

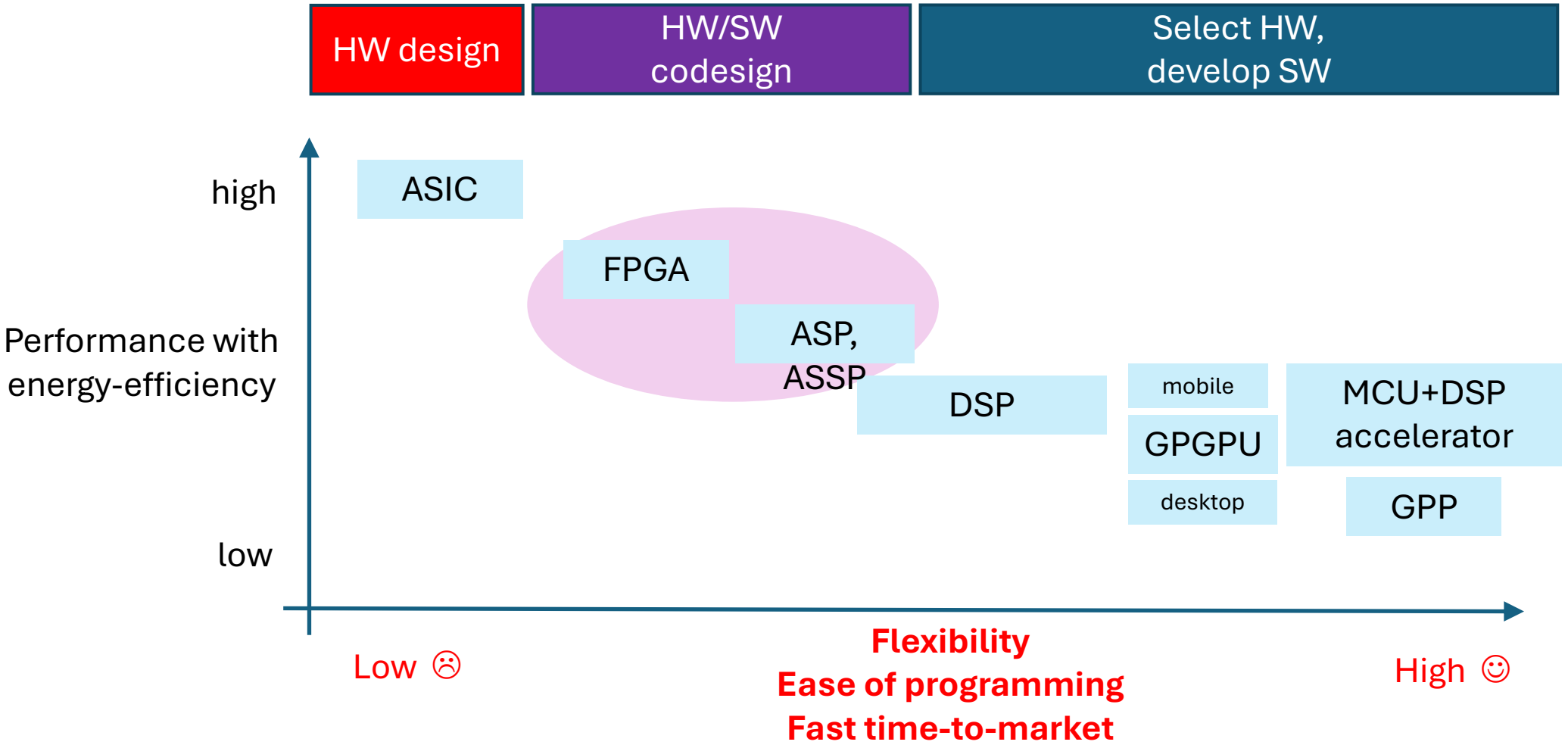
HW accelerators	(not programmable)	~1pJ/32bit arith. op.

GPU, mobile	I.T. PowerVR 540	16
GPU, laptop	AMD Radeon 6250	38
GPU, desktop	Nvidia GTX6750	60

CPU, embedded (integer)	ARM-M0+	85
CPU, mobile	ARM Cortex-A8	100
CPU, laptop	AMD C50	935
CPU, desktop	Intel i5-3570	6886

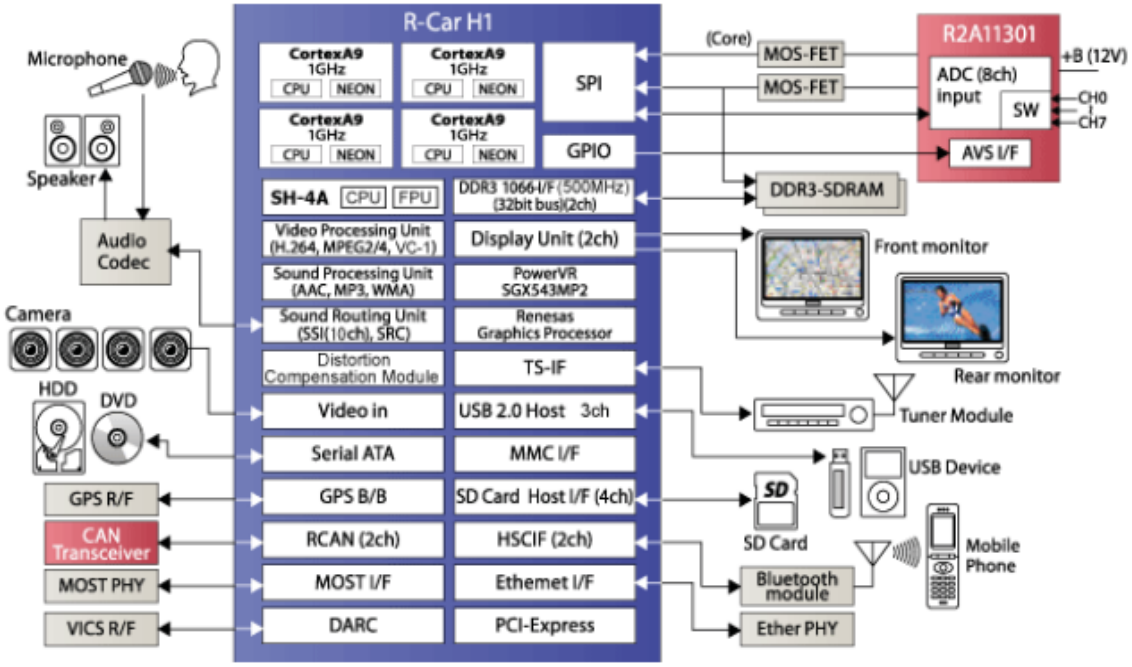


Platform comparison / HW-SW



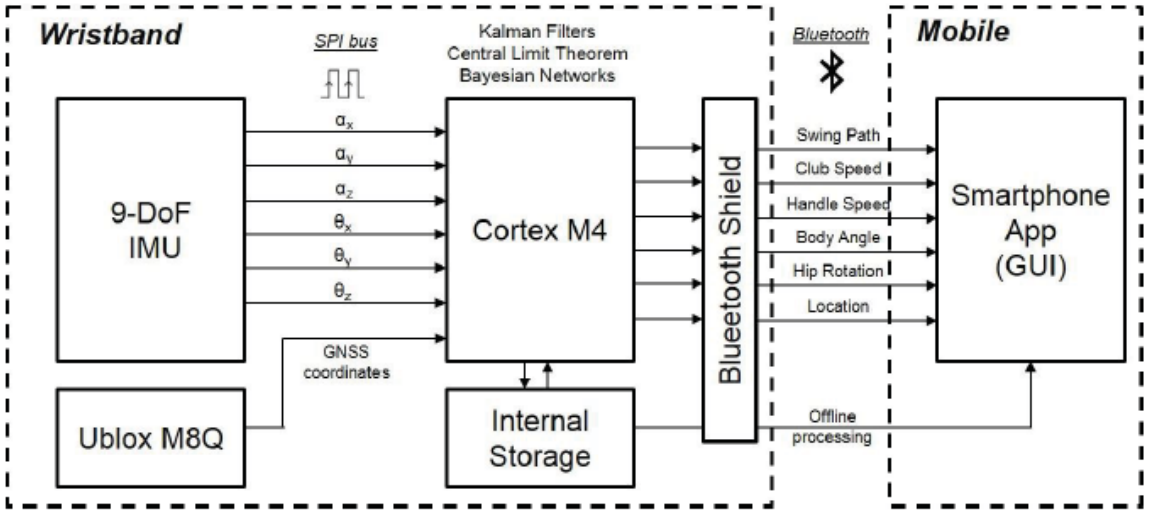
System-on-Chip (SoC)

- A typical implementation of a signal processing system is currently a SoC (System-on-Chip). SoCs integrate all components of an electronic system into a single integrated circuit
- E.g. typical features of SoCs for smartphones
  - Modem processing may be done with multiple different CPUs, some of them application specific designs, handling different protocol layers and hardware accelerators
  - Multimedia processing includes multiple different cores and hardware accelerators to support multiple coding standards
  - Graphics unit often used for GPGPU style computing to enhance speed and to provide for improved energy efficiency
  - External chips, e.g., GPS, in the platform have their own CPUs.



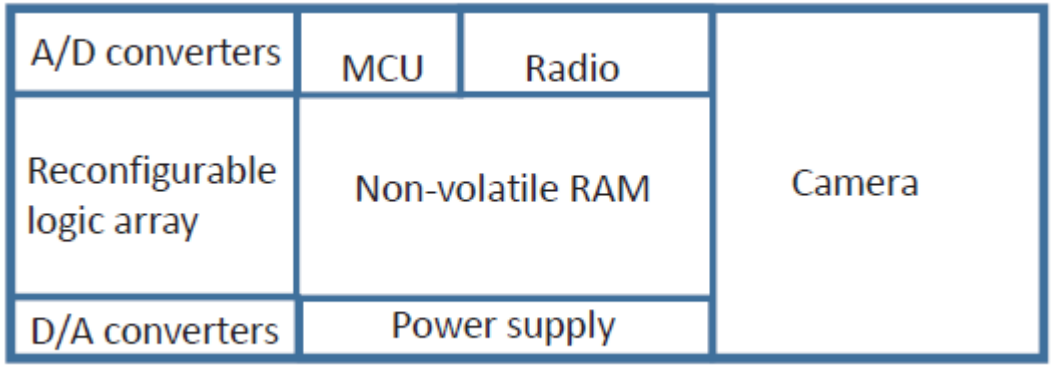
SoC for vehicle applications (Renesas Inc)

Less complex applications ...



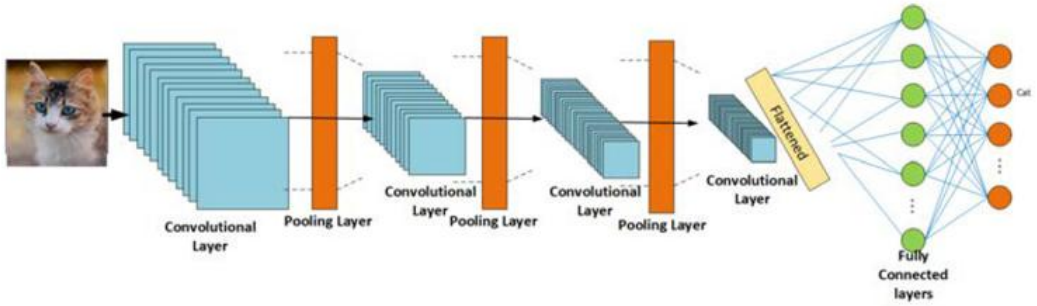
A sports bracelet application

No DSP needed: MCU (Cortex M4) includes floating-point unit + multiply-accumulate (MAC) instruction => fast development



Ideal IoT platform?
Single chip with implementation of basic functionality + reconfigurable logic for ASP: DSP acceleration (and machine learning/neural computation)

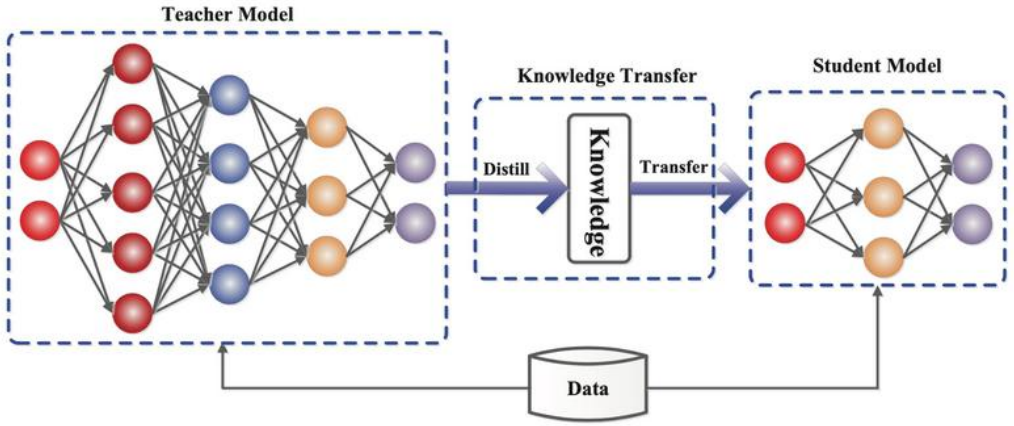
Machine learning processors



Modern DSP increasingly mixes data-driven methods with classic pipelines. Neural networks are already used for signal processing in digital predistortion, acoustic and communications post-filters, and factory anomaly detection and for image processing in image classification or object detection on the edge (e.g., UAVs). These deployments run on resource-limited edge devices (MCUs and SoCs) **and** on tiny AI accelerators such as embedded TPUs or NPUs.

Keys to success:

- Accelerators
- Optimized arithmetic
- Finding simpler models: **Knowledge distillation**



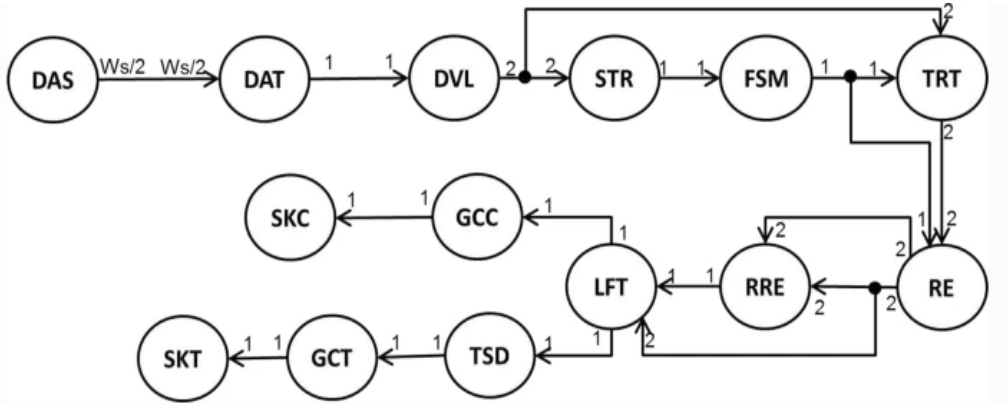
Characteristics of digital signal processing

- Data flow
- Arithmetics

Characteristics of DSP: **data flow**

- **Not reactive**, event-driven computing
- **Transforming data** under timing constraints
- **Deterministic computation**: little branching of control (ie. if-then statements)
- **Dataflow based formalisms** are appropriate for modeling computation (MoC)
  - Basis for tools for design and analysis
  - Example: Synchronous Data Flow (SDF)
  - Application-spesific signal processing (ASSP) development relies on such formalisms
- **Static scheduling** of computation: fixing resource use at the design time
  - Or at run-time before the start of computing (e.g. reconfigurable video coding)
  - Resource utilization can be very high

Dataflow example:



DAS = data acquisition source
SKT, SKC = storage of measurement results

Numbers attached to arrows correspond to production and consumption rates of data items (tokens).

Characteristic of DSP: **arithmetics**

- DSP relies on heavy use of arithmetics
  - Matrix operations
  - Fixed/adaptive filtering
  - Transforms (FFT, DCT, …)
  - Correlation
- Number of operations can be huge, but embedded platforms have constrained resources
- It is essential to optimize computations
  - Prefer algorithms having simple arithmetic (e.g. no multiplications)
  - Reusing results: compute intermediate values only once
  - Minimize word lengths, favor cheaper operations, minimize operation counts, minimize use of memory etc.
  - At the same time: avoid stability problems, avoid large numerical errors, keep SNR sufficient etc.
- **Emphasis in this course is in these algorithmic implementation aspects**

Course contents

Six parts

1. Binary number systems & arithmetic
2. Implementing fixed-point filters
3. Cordic & DCT
4. Multirate processing
5. Frequency-domain processing
6. Adaptive filtering

Context: Related courses at ITEE

Signal Analysis (031080A)

Digital Techniques 1 (521301A)

Introduction to Computer Systems (521287A)

Digital Filters (521337A)

This course

Application Specific Signal Processors (521281S)

Statistical Signal Processing (521348S)

Signal processing theory, algorithm side

HW/SW implementation side

Tools

Multiprocessor Programming (521288S)

Internet of Things (521043S)

Special courses on biosignals, communications, image/video processing, machine learning ...

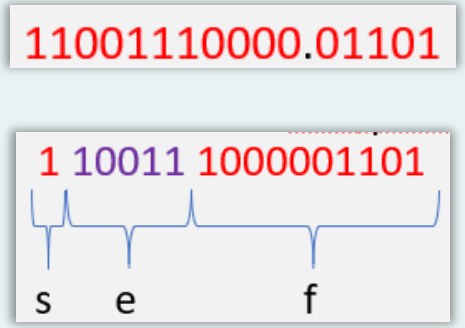
Background query

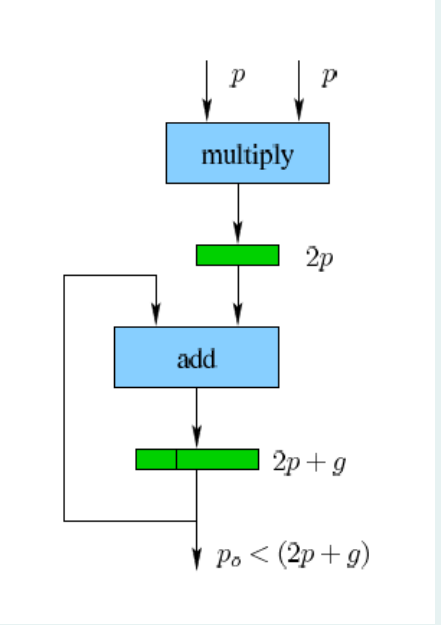
Go to https://presemo.oulu.fi/spslec1

and select options describing your background.

1. Binary number systems & arithmetic

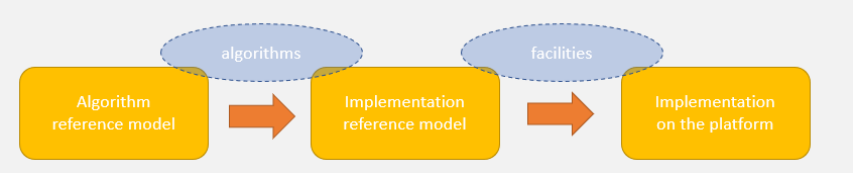
- Motivation
  - Understanding of binary arithmetic is mandatory for DSP software and hardware designers
  - Many design problems are arithmetic related
  - Choice of the number representation has large impact
- Topics
  - Basis: representation of integers
  - Fixed-point formats, floating-point formats
  - Characterization and use of formats in DSP designs
  - Basic component of DSPs: Multiply-accumulate unit (MAC unit)
  - **Neural network quantization** (new!)



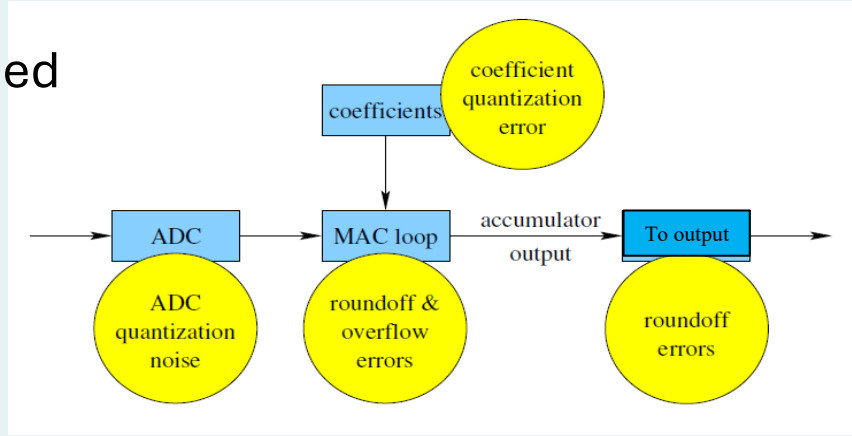


2. Implementing fixed-point digital filters

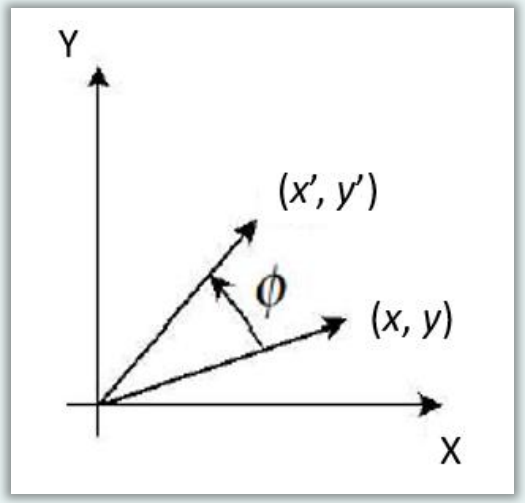
- Motivation
  - Typically, algorithm designer creates a reference model, that works in a floating-point format
  - HW/SW designers convert this to a design that utilizes the computational primitives of the target platform
  - Look at the basics and challenges of the conversion



- Topics
  - DSP system design flows: (1) analytic (2) simulation/tool-based
  - Matlab Fixed Point toolbox
  - FIR and IIR filter implementation issues
  - Error sources in implementation



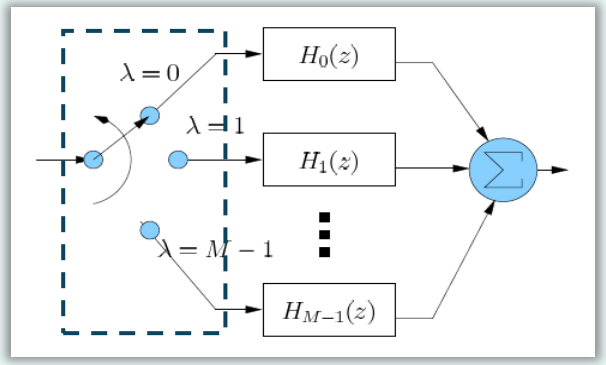
3. Cordic & DCT



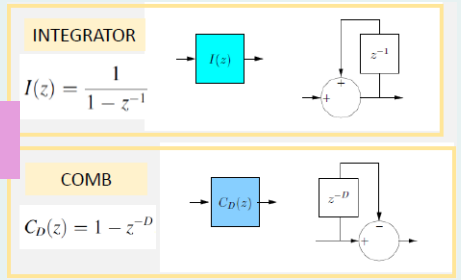
- Motivation
  - CORDIC - a tool for fixed-point function evaluation
    - evaluating trigonometric functions, also hyperbolic and linear
    - vector rotations: compact (de)modulators, transform butterflies
    - advantage: avoid use of full multipliers
  - DCT: the most common transform along with the FFT/DFT
    - used in many standardized video and audio codecs
    - CORDIC is a key tool in fast HW/SW implementations of DCT
- Topics
  - CORDIC explanation: Givens transform
  - Unified CORDIC for different types of functions
  - DCT and its implementation using CORDIC
  - 2-D DCT and JPEG coding

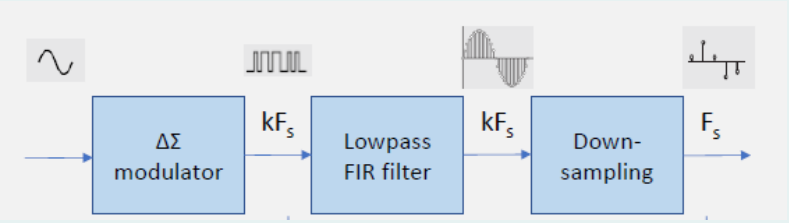
4. Multirate processing

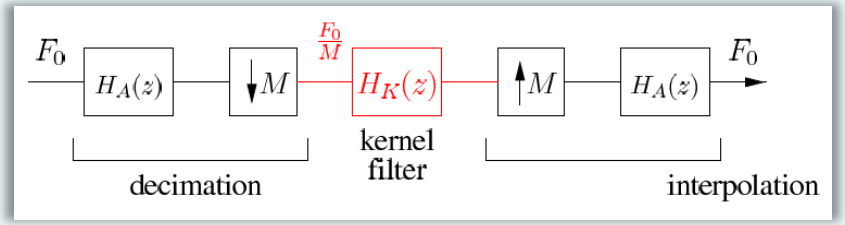
- Motivation
  - Multirate processing is common in DSP
  - Tools for reducing the number of aritmetic operations
- Topics
  - basis: polyphase decomposition and noble identities
  - rate changers: efficient decimation & interpolation, CIC filters
  - oversampling A/D conversion
  - narrowband FIR filter implementation



CIC

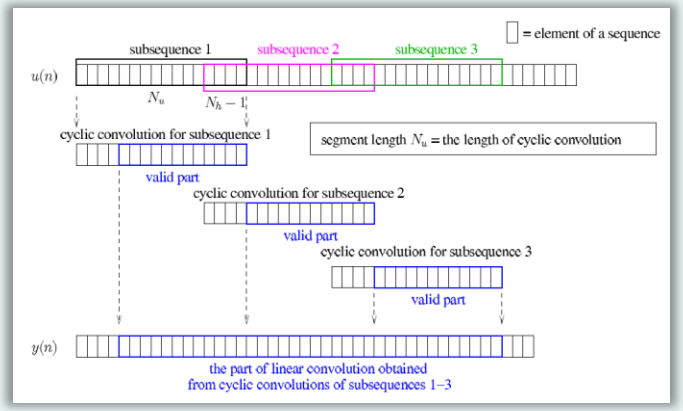


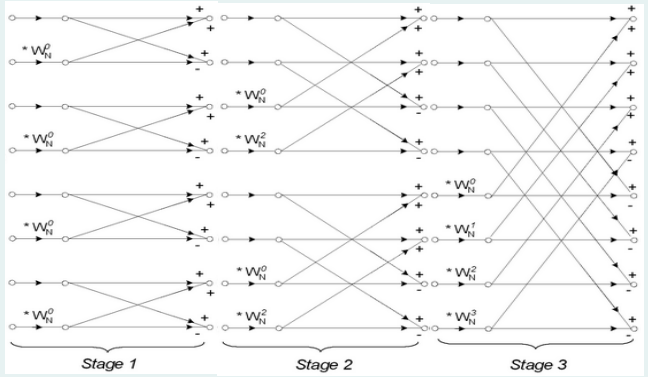


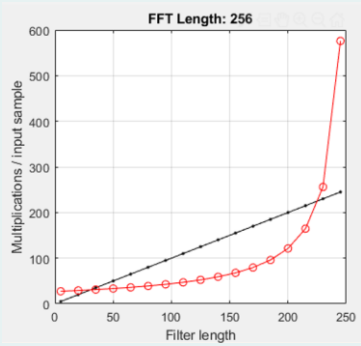


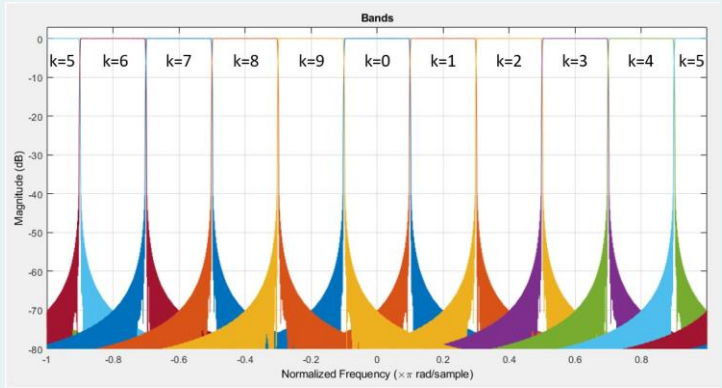
5. Frequency-domain processing

- Motivation
  - Convolution and correlation are common operations in DSP
  - Look at techniques that allow their implementation in the **frequency domain**
  - Many other applications also involve processing in frequency domain
- Topics
  - Sectioning methods (overlap-add, overlap-save)
  - **Comparison**: frequency domain versus time domain
  - Fast Fourier Transform (FFT)
  - Modulated filter banks (another multirate application)



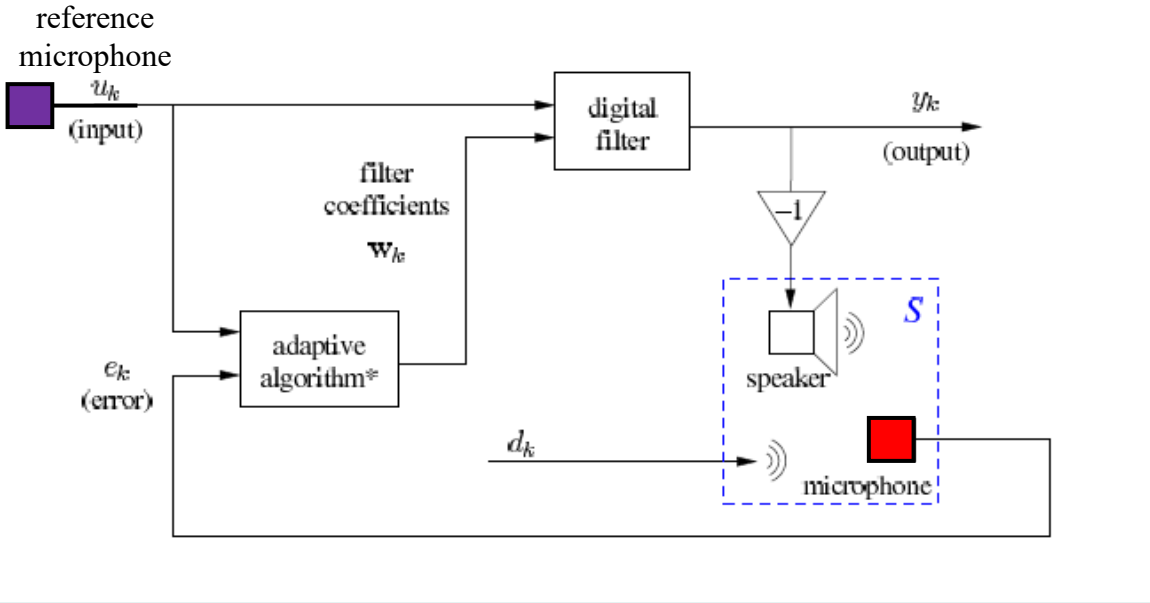






6. Adaptive filtering

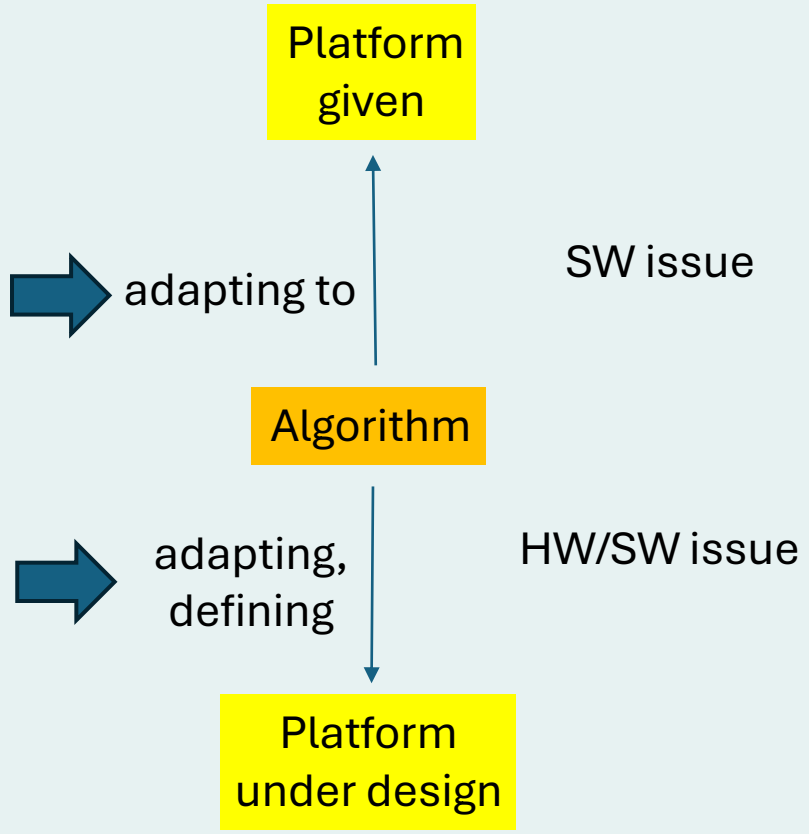
- Why included?
  - One of the most common operations in signal processing systems
  - Has many kinds of useful applications
  - Algorithms related to machine learning
  - Learning basics of Matlab's Simulink for building and running simulations
- Topics
  - Adaptive filter structure
  - Application examples
  - Common adaptive algorithms: LMS, RLS
  - Simulink tool



Example. Active noise cancellation

Summary

- This course takes a step towards digital signal processing implementations by considering them from the viewpoint of algorithms
  - How the basic algorithm becomes implementable on a resource-constrained platform?
  - How sufficient performance (e.g. SNR, stability) in the final implementation is confirmed?
  - HW/SW codesign perspective: algorithm analysis can help to define what kind of computational primitives are needed in HW
- The course can be considered as a continuation to the Digital Filters course
  - Extended discussion + new concepts
  - More experience on using associated tools in Matlab



Next …

- Lecture on Thursday 10:15, hall L5 - About number formats
- Warm-up:
  - Brush-up file on number format conversion available under **Readings** in Moodle; related exercise also available there.
  - File **intro1.pdf** covers topics to be discussed in the first part of the course
- Handout of Design Task #1 will be available on Thursday at 8:00
  - Lecture next Monday provides more background and hands-on
- Do group registration in Moodle
  - For seeking a partner, you may use **Discussion, Q&A forum** in Moodle front page