 by Dake Liu: dake@isy.liu.se

Dake Liu

TSEA26
Design of Embedded DSP Processors

Unit 1: Introduction

by Dake Liu: dake@isy.liu.se

Introduction to the course and the staff

Self introduction

Dake Liu Professor，
dake@isy.liu.se
Phone 281256
Datorteknik

Oscar Gustafsson Docent

Erik Bertilsson PhD

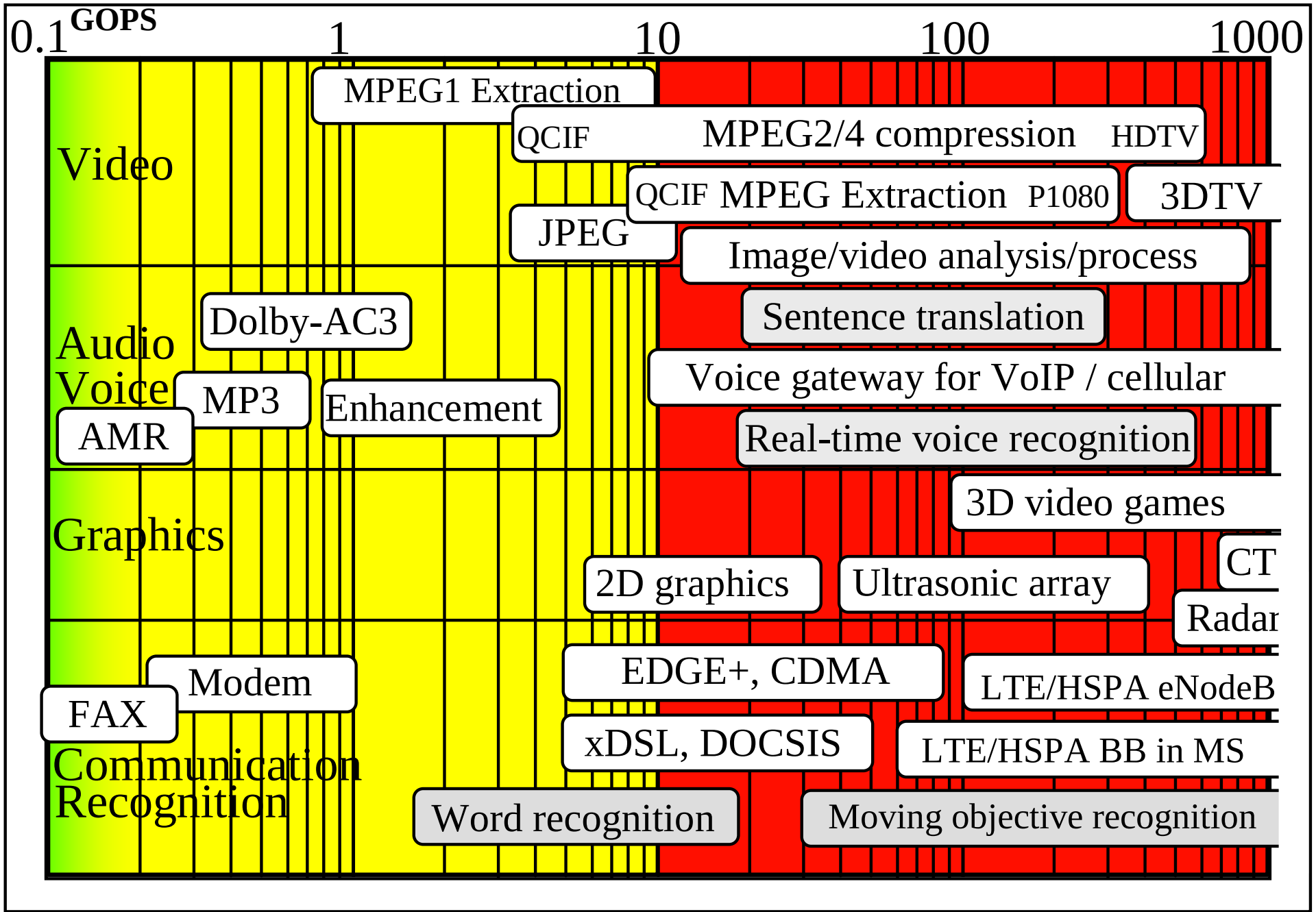
http://www.da.isy.liu.se/courses/tsea26/

 by Dake Liu: dake@isy.liu.se

ASIP and why ASIP

- 3 embedded digital circuits: MCU/**ASIP**/ASIC
- Application Specific Instruction-set Processor
  - Designed dedicated for an application domain
  - Reach ASIC performance & CPU flexibility in the domain (sounds too academic, we'll know why & how)
  - Difficult to design, get huge benefits
  - Not for CPU, ASIP is for DSP/GPU/NPU

Market requirements on embedded computing



 by Dake Liu: dake@isy.liu.se

Moors law saturated, performance requirements on computing is higher! Innovative architecture is the only solution

1. General computing

a) Homogeneous SMP：same CPU，flexible， saturated

b) Heterogeneous：Allocate tasks to suitable processors, e.g. SSE

c) Deep heterogeneous：CPU (FSM)+GPU (vector)+FPGA (P-h)

2. Embedded computing

a) ARM/MCU：one ARM to multi ARM，to architecture IP

b) ARM + ASIC accelerator：Huge NRE cost and short life time

c) ARM + GPU + DSP + ASIP: The future embedded syste,

by Dake Liu: dake@isy.liu.se

When CPU, ASIP, and ASIC

CPU

- All general applications
- x86，ARM，
- High end, Very high design cost
- Strong SW ecological support

ASIP

- For volume embedded applications
- Low SW eco requirement
- Performance in an application domain

ASIC

- A function module not progrmmable
- Performance
- High design cost & Short product life time

The great history of our division, our spin-off company offered 20% global BB-ASIP IP for mobile phones!

The goal of the course

- The course will be challenging and rewarding
- To learn industrial (warning) design experiences
  - Learn (application specific) processor design
  - Focusing on efficient HW microarchitecture design
  - Learn firmware design basic skills
- After completing the course we will
  - Design a simple processor, know embedded systems, DSP implementations, DSP processor architectures.

 by Dake Liu: dake@isy.liu.se

You can do sth after the course

- Design an application specific instruction set or map functions to hardware
  - Code analysis, instruction set specification, SW-HW co-design
- Design a processor microarchitecture
  - Datapath, Data access and memory, control path, and peripherals
- Be able to write efficient firmware for
  - Quality computation kernels (parallelization, finite precision etc)
- have knowledge of firmware development toolchain
- have knowledge of processor integration and verification.

The scope and the backgrond

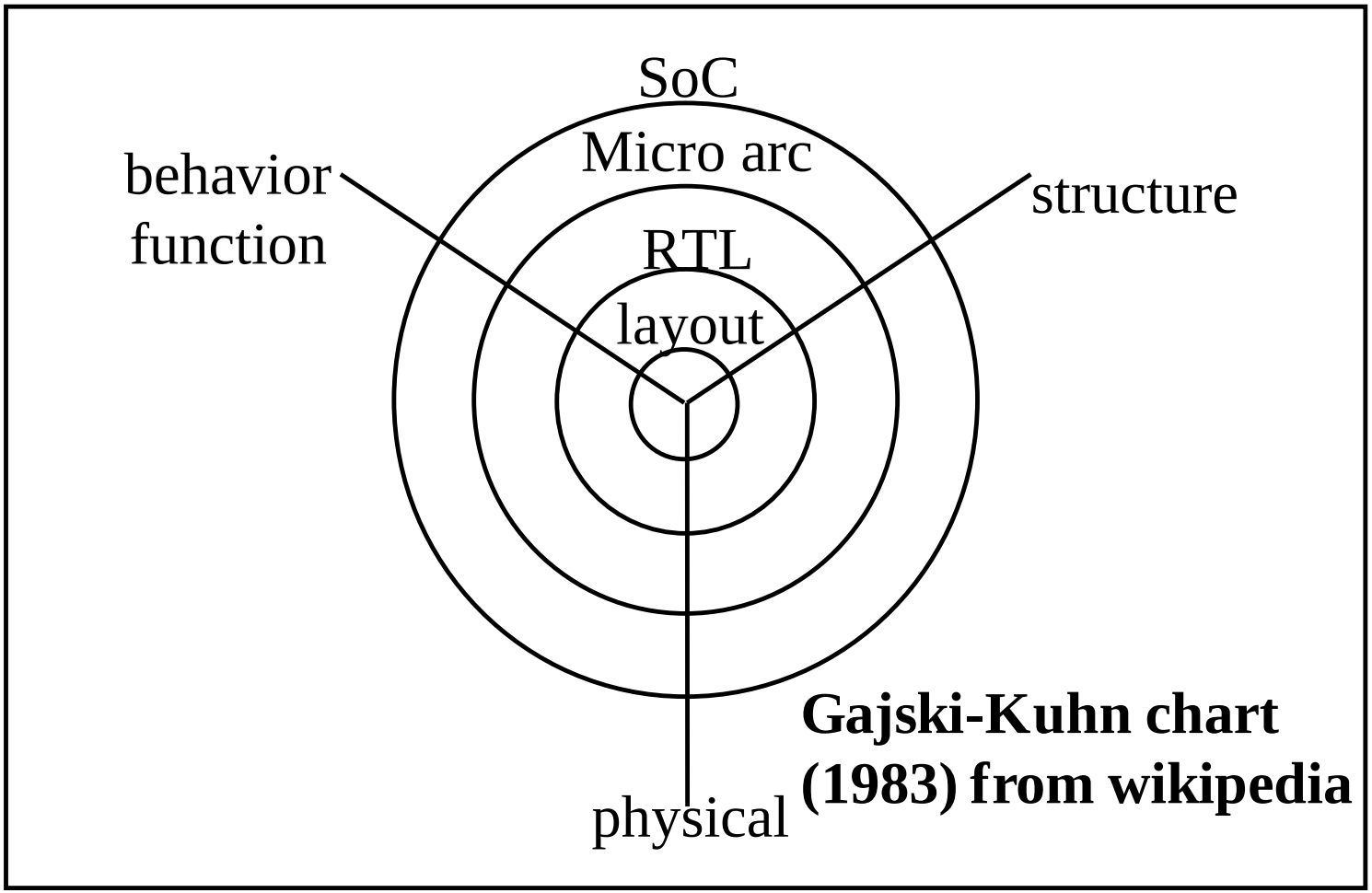
- It is about implementation, we learn methods
- Literature and references
  - My book, Andreas Ehliar contributed very good slides
  - Synopsys ASIP Designer (to know how to synthesis)
- Background knowledge:
  - Logic circuit (Digitalteknik)
  - Processor fundamentals (Datorteknik)
  - DSP fundamentals
  - Programming skill C, ASM, VHDL or Verilog

For special international students

- Diplomatically speaking, if you don't have the pre-requisites, you will have gained a lot of knowledge when passing this course. . .
  - Students have passed this course before, without having all of the prerequisites, but they probably had to work pretty hard to do so.
  - Talk to me during the break if you have questions.

 by Dake Liu: dake@isy.liu.se

HW Design methodology



 by Dake Liu: dake@isy.liu.se

The method to learn the course

- **Divide & conquer**
  - **Refine & simulate**
    - **We will use Daniel Gajski Y-Chart**
    - **What whatever HW design and spec, we describe behavior, architecture, and phy requirements**
  - **Describe & synthesis**
    - **We will not teach ASIP synthesis**

 by Dake Liu: dake@isy.liu.se

Summarize what/how to learn

Skills

Concepts

	20% System understanding	50% Plan HW schematic	5% HW coding	15% FW coding	10% Integration verification
Finite precision 10%	Just enough quality	Where/what	Sat/rnd	Gain strl	Corner cases
Micro architecture 10%	Functions to map	Sharing	sharing	------	Balance
Register file 10%	Write conflict	Critical path	Fanout	Life time	Fanin fanout
ALU: Arithmetic & Logic 10%	HW sharing	Reuse skill	IP code	precision	corner
MAC: MUL and ACC 10%	MAC/LALU/MLU	Reuse skill	IP code	Use MAC	corner
Memory and data access 10%	Modulo	Pipeline	pipeline	D-allocate	IP coding
Program flow control 10%	PC and I-decoder	I-decoders	PC	C-hazard	Pipeline
Assembly coding tools 10%	Behavior/arch SIM	------	------	debug	Verification
Firmware plan & design 10%	Bit/mem/cycle	------	------	plan vs code	SW v.s. HW
Survey of Different ASIP 10%	Efficient VPU	Tool limited	critical	Kernels	

The way to learn/reach TSEA26

- Based on Y-chart
  - Focusing on specification / description
  - Behavior, Architecture, Phy constraint
- Based on the 2D what/how table
  - To fill in the table step by step
  - 2D-table, a good way to spec/describe

 by Dake Liu: dake@isy.liu.se

Education process and the Examination

- Fö: 11 Lectures, from numbers to an ASIP
- Le: 6 Tutorials, follow me step by step
- La: 4 Labs, design a processor by filling in
- Exam
  - Written examination 3ECTS
  - Laboratory assignments 3ECTS

TSEA26 staff

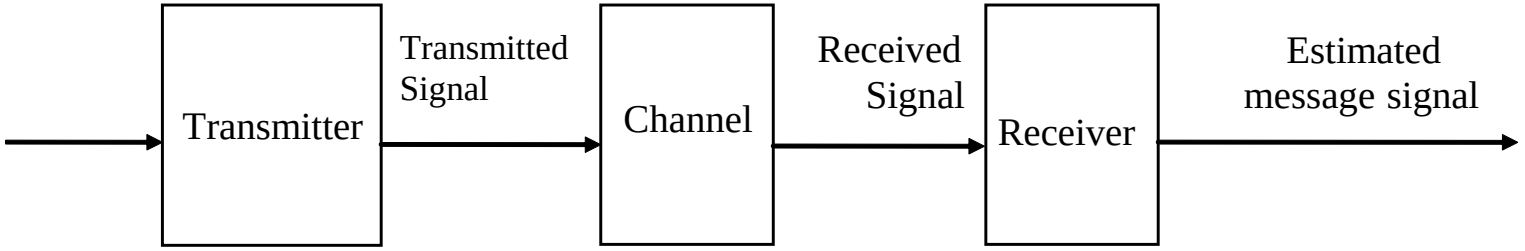
- Lectures/examiner: Dake Liu
- Tutorial: Oscar, Dake
- Labs: Oscar, Eric
- Course homepage: http://www.isy.liu.se/edu/kurs/TSEA26/
- About the Lectures, Tutorials, The labs

by Dake Liu: dake@isy.liu.se

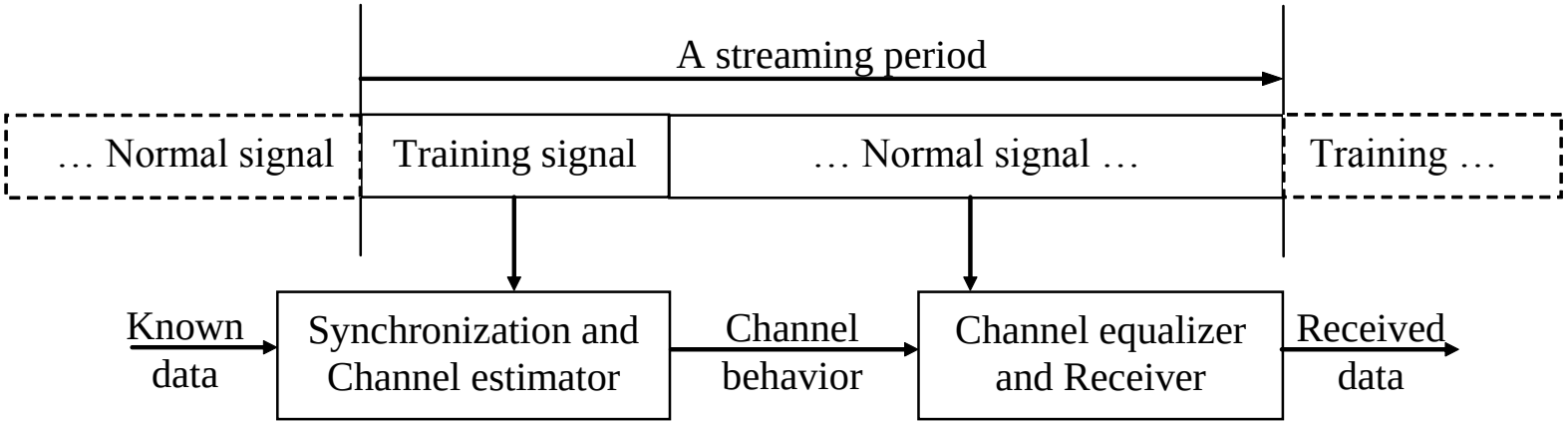
Motivation to learn ASIP design

 by Dake Liu: dake@isy.liu.se

Application example: Communication



Streaming signal between a transceiver pair

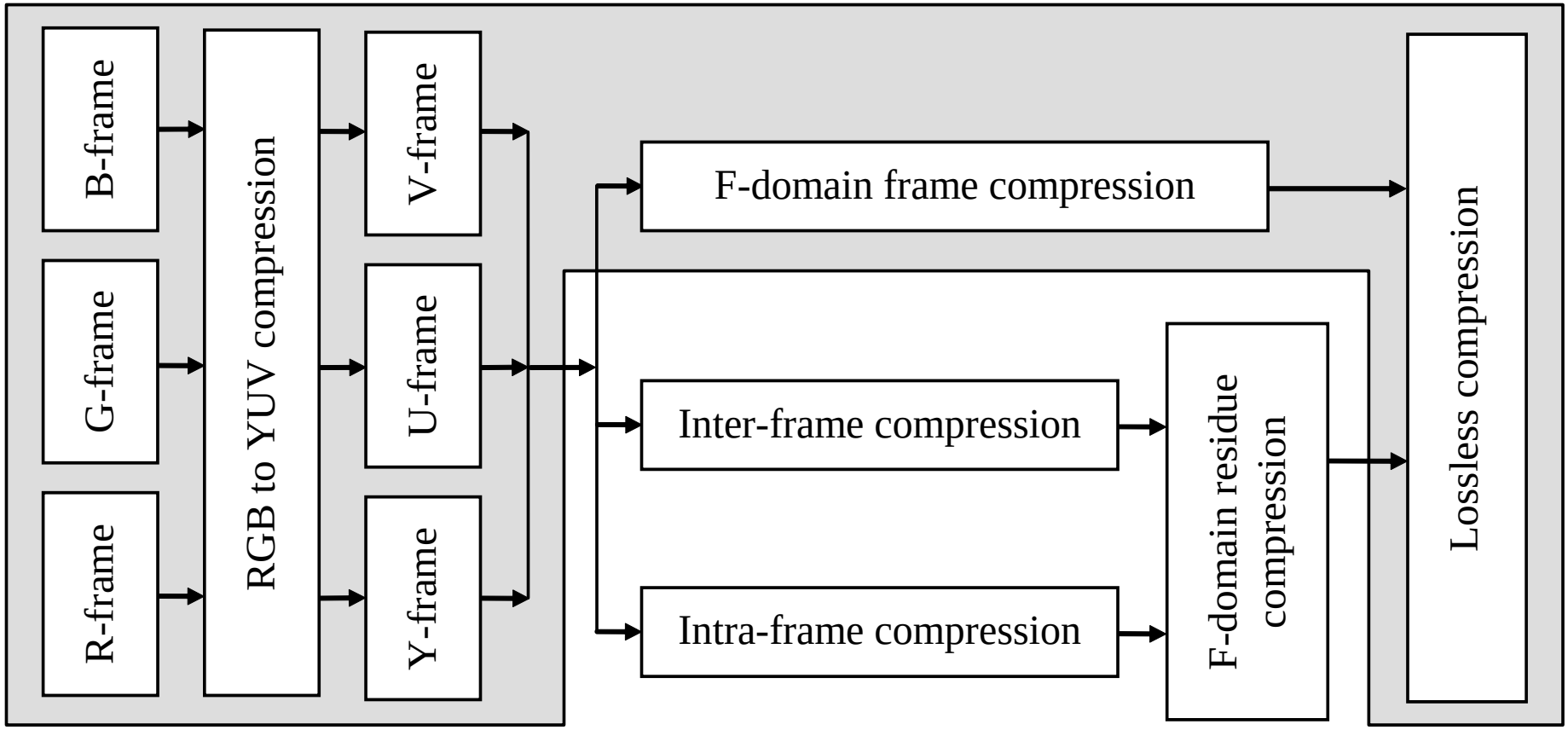


Recovering data from noise channel

We need 100 times x86 performance and 1/100 x86 power and cost

 by Dake Liu: dake@isy.liu.se

Application example: Image and video compression



We need 10 times x86 performance and 1/200 x86 power and cost

 by Dake Liu: dake@isy.liu.se

Application example: DSP on a general processor

- When you have a computer, run software on it
- Video audio player, image viewer based on SIMD: Single Instruction Multiple Data
  - SSE (Intel), AltiVec (IBM), NEON (ARM), ePUMA (PhD)
- You do not want to design a video player using a 6000$ x86 Xeron E7 (a joke)
  - You need to design an Application Specific DSP for high volume and long life time product

 by Dake Liu: dake@isy.liu.se

We thus need embedded system

- A computer system with dedicated functions within a larger mechanical or electrical system, often with real-time computing constraints
- 98% of microprocessors, are manufactured as components of embedded systems.

ASIP

- Application Specific Instruction-set Processor
  - A processor optimized for a certain kind of application domain
  - Instruction set optimized for a certain application
  - Accelerators for very demanding parts of the application
  - Low power, high performance, low cost
- The focus of this course

 by Dake Liu: dake@isy.liu.se

ASIC

- Application Specific Integrated Circuit
  - Lowest cost (in high volume)
  - Lowest power
  - Highest performance
- In practice:
  - Very high development cost
  - Long development time
  - May reduce a product lifetime, induce another chip design with huge NRE cost

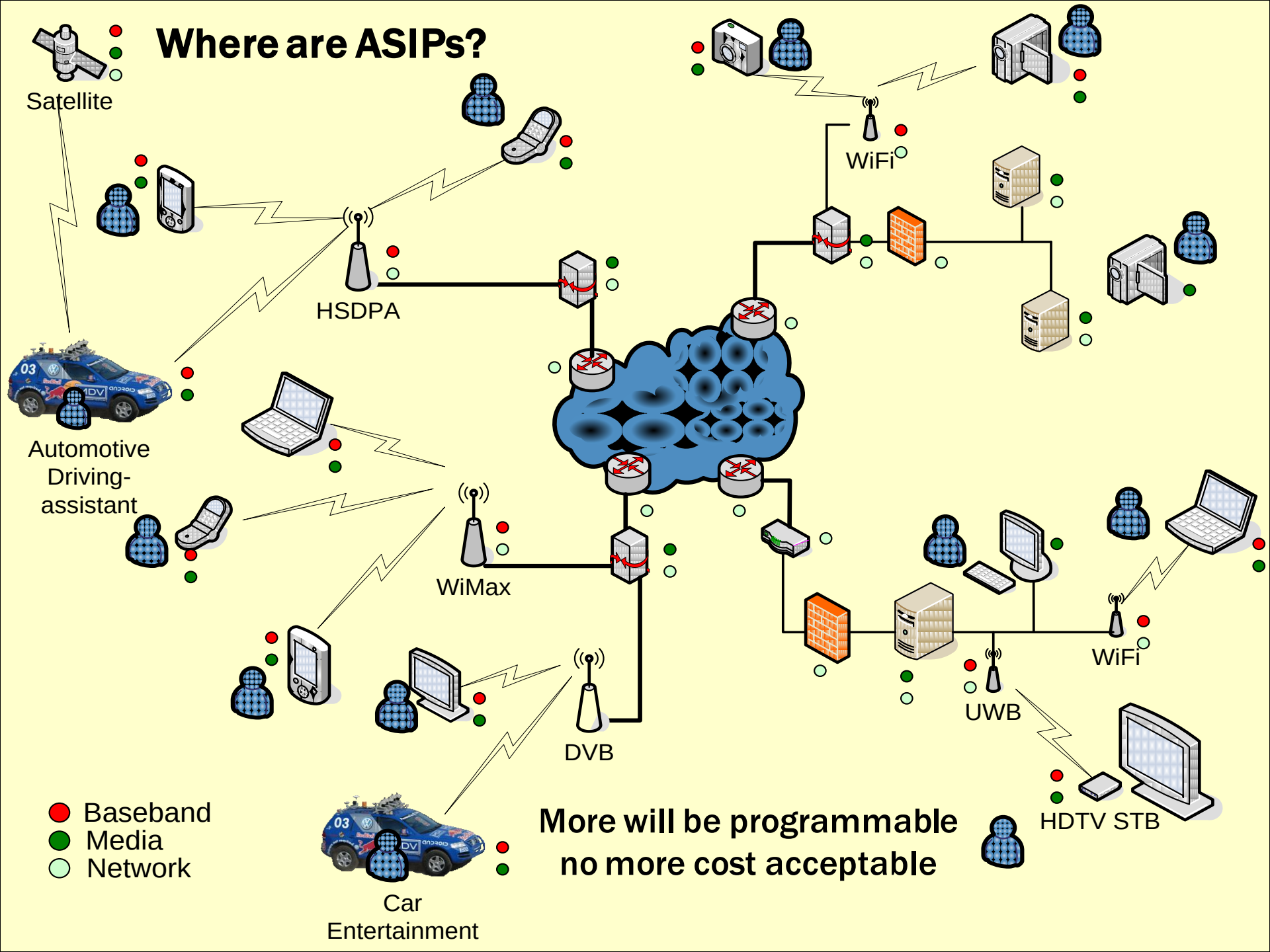
by Dake Liu: dake@isy.liu.se

The future

Any applications, when functions can be predicted and the volume is high, it will be implemented on ASIP

Classical highend embedded system consists of ARM + ASIC accelerators, the future one will be ARM+ ASIP

Where are ASIPs?



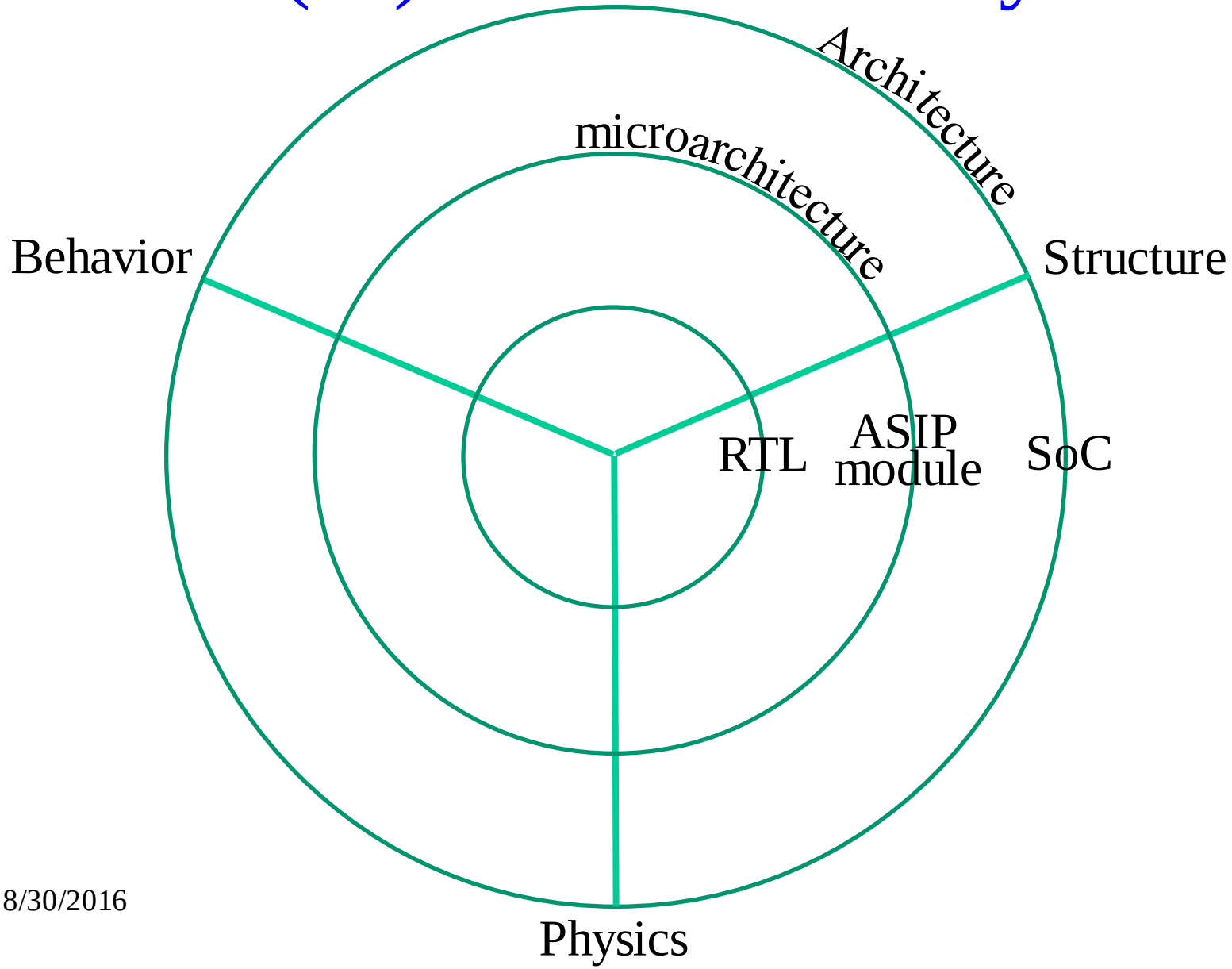
Let us learn it!

by Dake Liu: dake@isy.liu.se

ASIP fundamentals

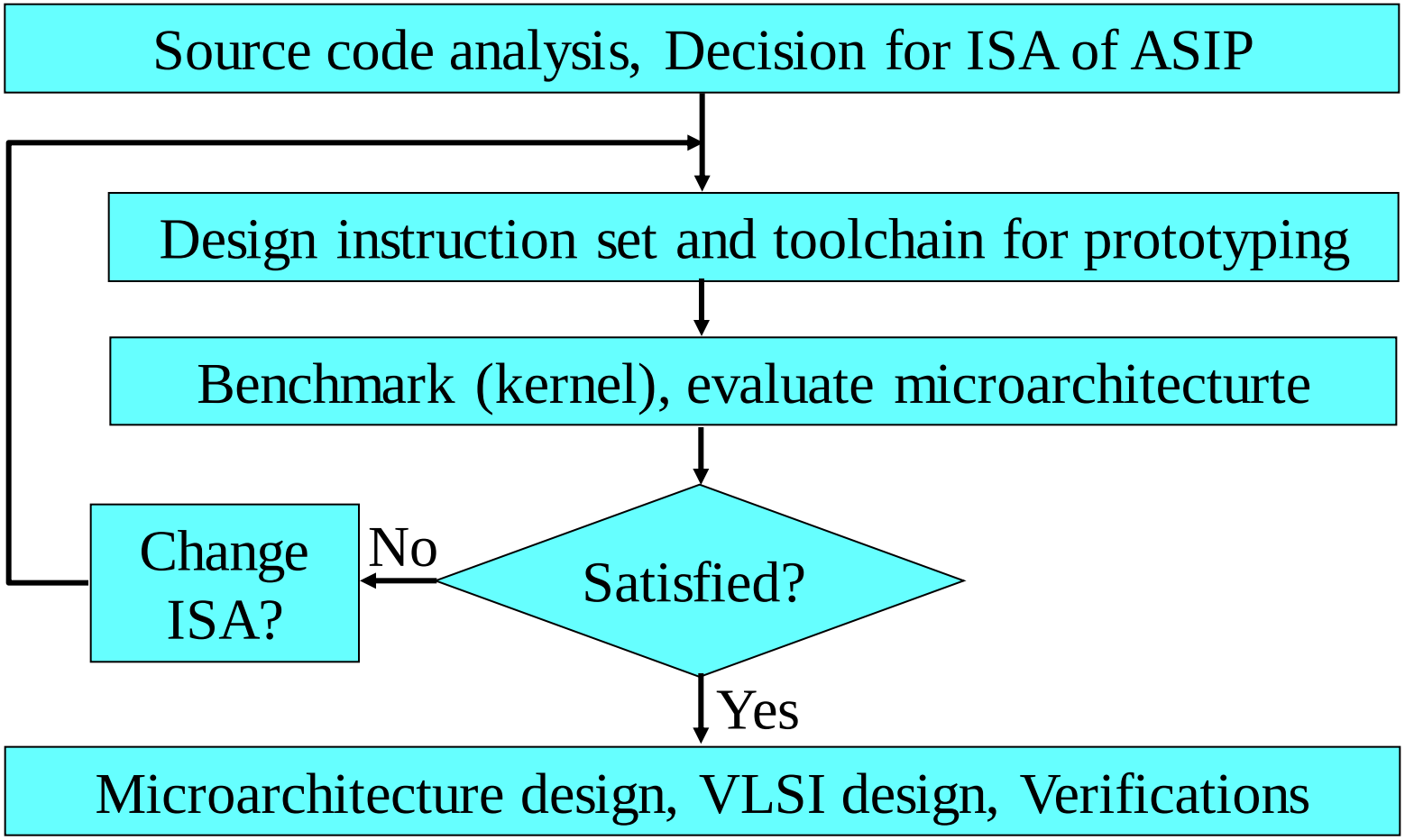
by Dake Liu: dake@isy.liu.se

ASIP (IP) in an Y-chart system



 by Dake Liu: dake@isy.liu.se

ASIP design flow



 by Dake Liu: dake@isy.liu.se

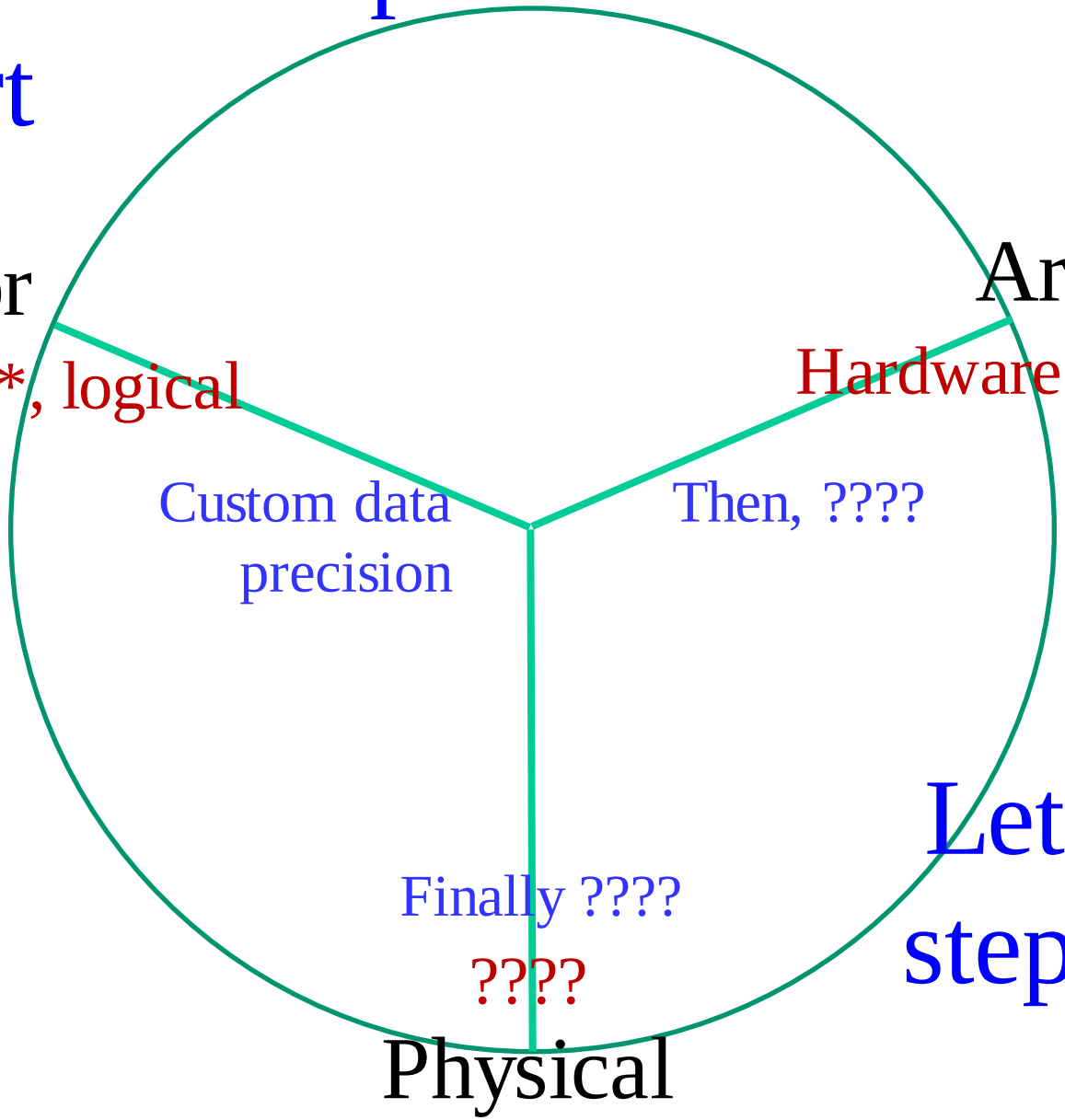
To design an efficient ASIP, we need

1. Application specific datapath and data types
   - Deep understanding of data types /corner cases
   - Data types / finite data precision (audio example)
2. Application specific memory access
   - Deep understanding of parallel data features
3. Application specific program flow
   - Deep understanding of control complexities

by Dake Liu: dake@isy.liu.se

Look into Datapath

Y chart



Let us do it step by step

What is data type and why to discuss on it

- Data type defines implements & uses of data
  - Standard data types: Integers, floating-points, booleans, characters, alphanumeric strings
  - Custom data type for ASIP: sufficient precision, low cost, and minimized run time.
- Skills to handle low cost data type via trade offs of low cost HW and high quality FW.
  - Going through and to learn the course TSEA26

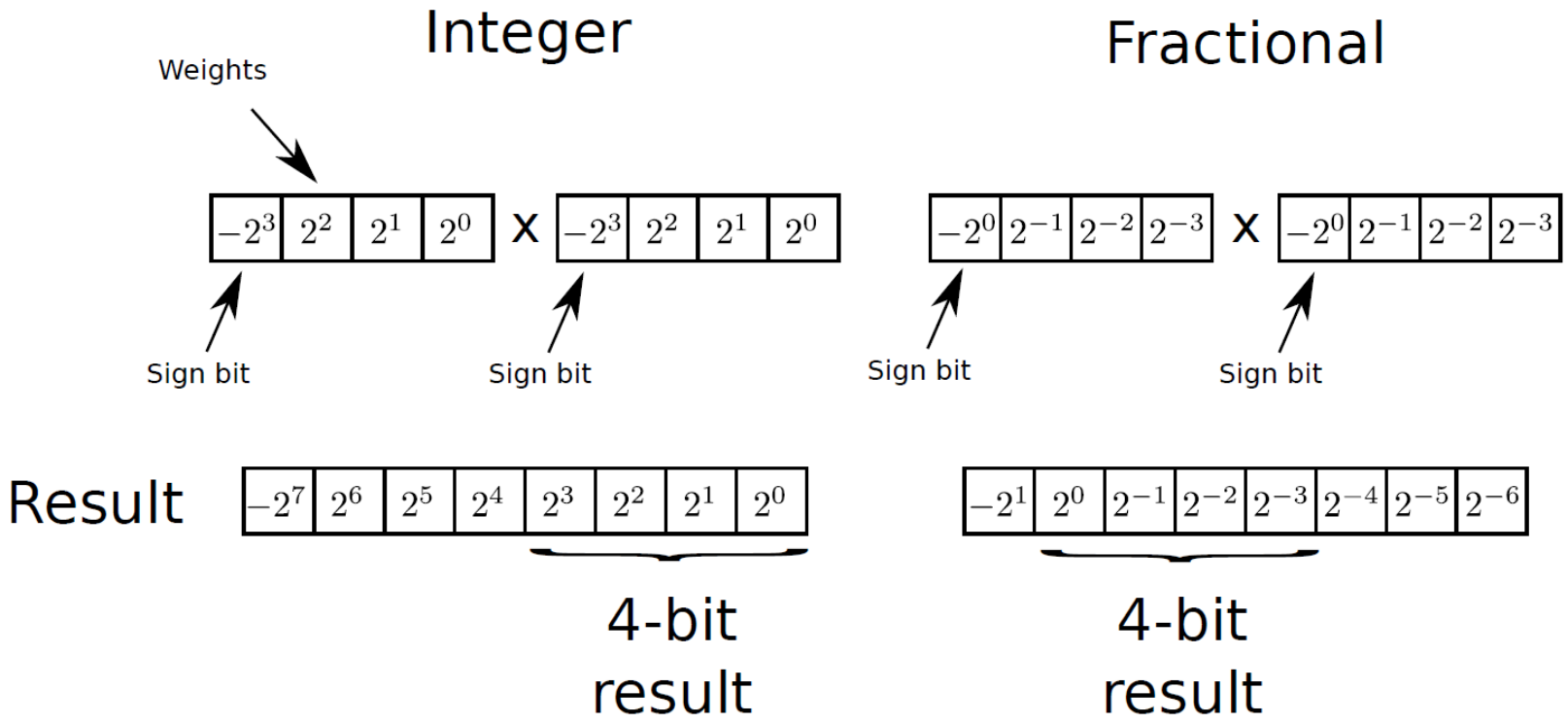
 by Dake Liu: dake@isy.liu.se

Custom data precision: Fixed point numerical representation

- What is fixed point numerical representation?
- Integer or fractional representation
  - Dominates DSP field
  - Integer: Between -2^{n-1} and $+2^{n-1}-1$
  - Fractional: Between -1 and $+1-2^{-n+1}$
  - (Where n is the number of bits)

 by Dake Liu: dake@isy.liu.se

Fractional vs integer multiplication



- Integer: Overflow is a real concern
- Example: $0111 \times 0111 = 00110001$
  - Integer: $00110001_2 = 49_{10}$
  - Integer (truncated): $0001_2 = 1$ (Fatal error)
  - Fractional: $00.110001 = 0.765625$
  - Fractional (truncated): $0.110_2 = 0.75_{10}$ (Pretty close to correct result)

 by Dake Liu: dake@isy.liu.se

Precision

- The distance between the smallest values that can be represented using a certain number format
- Example:
  - 16 bits fractional numbers, the precision of the data is
    $0{:}000\text{-}0000\text{-}0000\text{-}0001_2 \approx 0.000305_{10}$

 by Dake Liu: dake@isy.liu.se

Dynamic range (of a digital signal)

- The ratio *between* the largest range a certain number format can represent *and* the precision.
- E.g., the dynamic range of a 16 bit fixed point number format is 65535/1
- Commonly measured in dB. Example:
  - $20 \times \log_{10}65535 \approx 96$dB. Each extra bit adds 6dB

 by Dake Liu: dake@isy.liu.se

Quantization error (of digital signals)

- The numerical error introduced when a longer numeric format is converted to a shorter one
  - E.g: 8FFFFFFFF → 8FFF (fractional)
  - Quantization errors occur during A/D and D/A conversions are not discussed in this course.

$$x = \sum_{i=1}^{b} \alpha_i 2^{-i} + \sum_{i=b+1}^{c} \alpha_i 2^{-i} = Q_T[x] + E_T$$

  - E_T will be cut off as truncation / qualtization error

 by Dake Liu: dake@isy.liu.se

Drawbacks of Fixed Point

- Sometimes it is not possible to separate dynamic range and precision
- Higher firmware design costs (E.g., when using a Matlab model as a reference)

 by Dake Liu: dake@isy.liu.se

Floating Point

- Numbers are represented by a mantissa (m), an exponent (e), and a sign bit (s)
- value = $-1^{S} \times m \times 2^{e}$
- IEEE-754 Single precision floating point number (32 bits)
- NaN table and implementation issues

 by Dake Liu: dake@isy.liu.se

Floating Point

(IEEE 754-2008)	Sign	Exponent	Significand field	Total bits
Half	1	5	10	16
Single	1	8	23	32
Double	1	11	52	64

 by Dake Liu: dake@isy.liu.se

Why not Floating Point?

- When precision is more important than dynamic range
- Complicated data path, longer critical path, higher silicon cost for arithmetic units
- Harder to reason about the stability of calculations
  - Example: $x + y + z \neq z + y + x$

Let us review what did we do

- If you understood, congratulations!
- If you do not understand, find out the reason?
- What is ASIP, why design it?
- What is microarchitecture? How to design?
- How many kinds of DSP implementations?
- We already have the 1st step: custom data types.
- To think when you are home: Microarchitecture design following Y-chart

 by Dake Liu: dake@isy.liu.se

ASIP on markets (mostly in SoC as IP)

ASIP	Applications	IP/year	$/year
SDR baseband	Handset, base station	1B	3-5B
ISP for image and video	Handset, video, camera	1B	1-2B
Video codec	Handset, survaillence	5亿	2-4B
Storage	SSD、M-cards	>100M	~500M
Gateways	Gateway, home gateway	>50M	~500M
Network processors	ISP、router	>10M	~100M
Industrial control	Motion and motor control	100M	300M
Robots	Vision, control	10M	30M
IoT	Communication, sensing	50B	50B
Deep learning	Server, terminal	?	?
Defense DFE	Baseband, sensing, ISP	?	?
Defense AP	Recognition, decision	?	?
……Video application, VR, AR, medical, toys, commercial, and much more……			

Self reading after the lecture

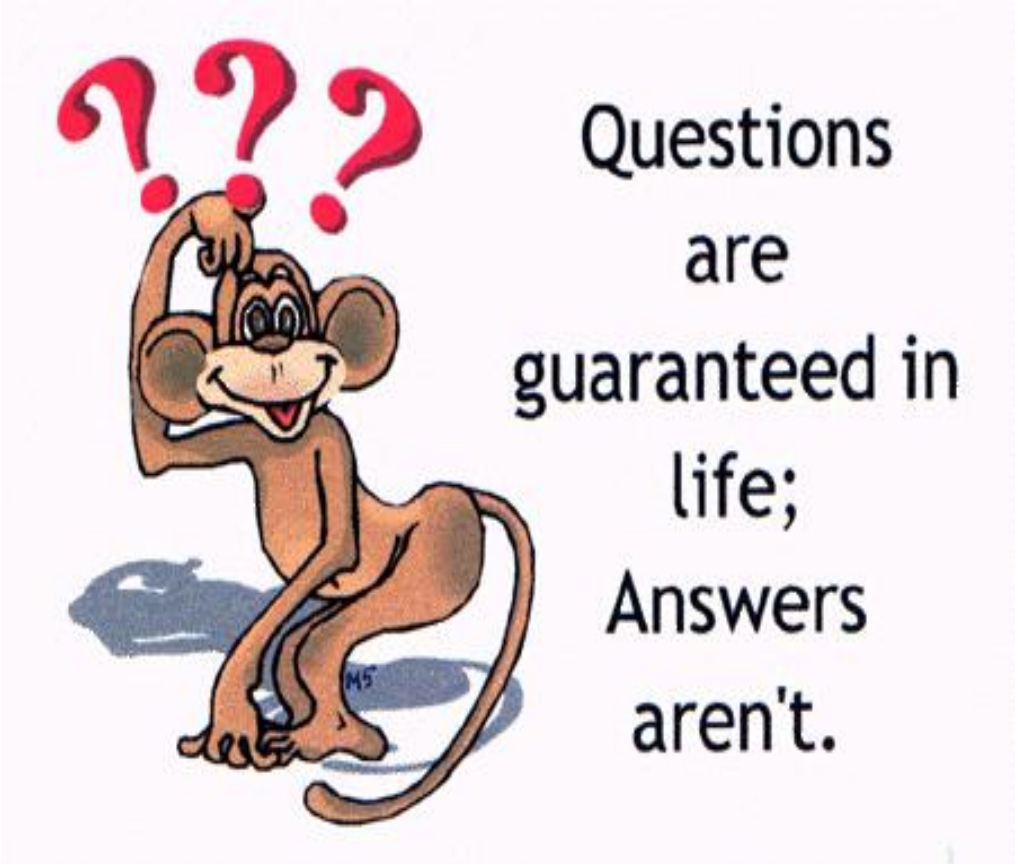
- Pick up related knowledge in the pre-requirement list
- Chapter 1
- Integer part in Chapter 2
- Think about: How to reach the best data quality via FW coding on low cost fixed point processor

 by Dake Liu: dake@isy.liu.se

Exciting time now! Let us discuss

- **Whatever you want to discuss and related to HW**
- **You will have the chance after each lecture (Fö), do take the chance!**
- **Prepare your Qs for the next time**

 by Dake Liu: dake@isy.liu.se



Welcome to ask any questions you want to

- I can answer
- Or discuss together
- I want to know what you want

Dake Liu, Room 5xx coridoor B, Hus-B, phone 281256, dake.liu@liu.se