Chapter 1-1: Embedded Computing

High Performance Embedded Computing

Soo-Ik Chae

Topics

- Landscape of embedded computing.
- Applications.

The landscape of embedded computing

- Lots of embedded applications require very high performance:
  - Communications.
  - Multimedia.
- Must also meet strict design goals:
  - Real-time performance.
  - Power/energy consumption.
  - Cost.

What is an embedded system?

- An engineering artifact involving computation that is subject to physical constraints
- Two kinds of physical constraints
  - Reaction constraints: deadlines, throughput, and jitters; they originate form the behavioral requirements of the systems.
  - Execution constraints: processor speed, power, and hardware failure rates; they originate form the implementation requirements of the system.
- Reaction constraints are studied in control theory.
- Execution constraints are studied in computer engineering.

Designing embedded systems

- No one architecture (hardware or software) can meet the needs of all applications.
- We need to be able to design a system from the application:
  - Quickly and efficiently.
  - With reliable results.
- The key to embedded systems design is gaining control of interplay of computation with reaction and execution constraints to meet a given set of requirements

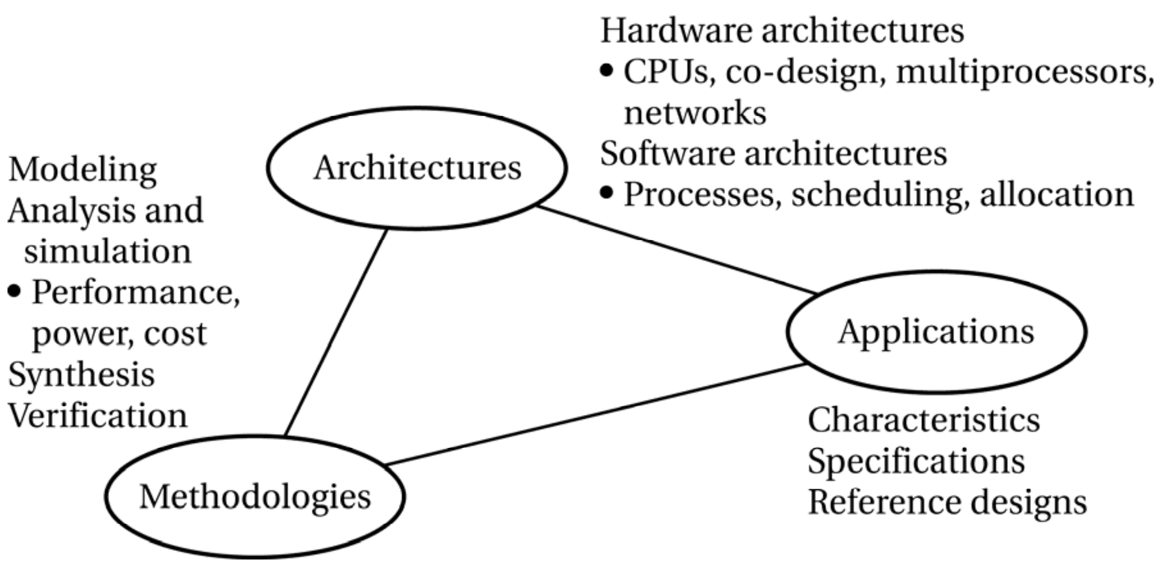
What is system design in general?

- System design is the process of deriving, from requirements, a model from which a system can be generated more or less automatically.
- A model is an abstract representation of a system.
- Software design is the process of deriving a program that can be compiled.
- Hardware design is the process of deriving a hardware description from which a circuit can be synthesized.

What is embedded systems design?

- Embedded systems consist of hardware, software, and an environment, which have in common with most computing systems.
- (Essential difference) Embedded systems involve computation that is subject to physical constraints.
- Powerful separation of computation (software) from physicality (platform and environment), which has been one of the central ideas enabling the science of computing, does not work for the embedded system.
- The design of embedded system requires a holistic approach that integrates essential paradigms form hardware design, software design, and control theory in a consistent manner.

Aspects of embedded system design



Architectures

- Both hardware and software architectures are important.
- The structure of the system determines cost, power, performance.
- Different application requirements lead us to different architectures.

Applications

- You can't design the best embedded systems if you don't know anything about your application.
- You can't be an expert in everything.
  - But a little knowledge goes a long way.
- Domain expertise helps you make trade-offs:
  - Can the requirements be relaxed?
  - Can one requirement be traded for another?

Methodologies

- We must be able to reliably design systems:
  - Start from requirements/specification.
  - Build a system that is fast enough, doesn't burn too much energy, and is cheap enough.
  - Be able to finish it on time.
  - And know before we start how difficult the project will be.
- Methodology = Combine CAD tools and manual steps and codify our knowledge of how to design systems.
- Methodology help us to make large and small decisions.
- ESD encompasses both hardware and software

Methodologies

- Steps in a methodology may be implemented as tools.
- Analysis and simulation tools are widely used to evaluate cost, performance, power consumption.
- Synthesis tools create optimized implementation based on specifications.
- Tools are useful
  - To understand an specific application
  - Productivity (short time-to-market)

Modeling

- A key aspect of methodology is modeling.
  - Work with a simplified version of the object.
  - Abstraction
- Embedded system: complex functionality built on top of sophisticated platforms
- Designers must use a series of models to complete a system design
- Early stages: reasonably accurate simple models
- Later stages: more sophisticated and accurate models

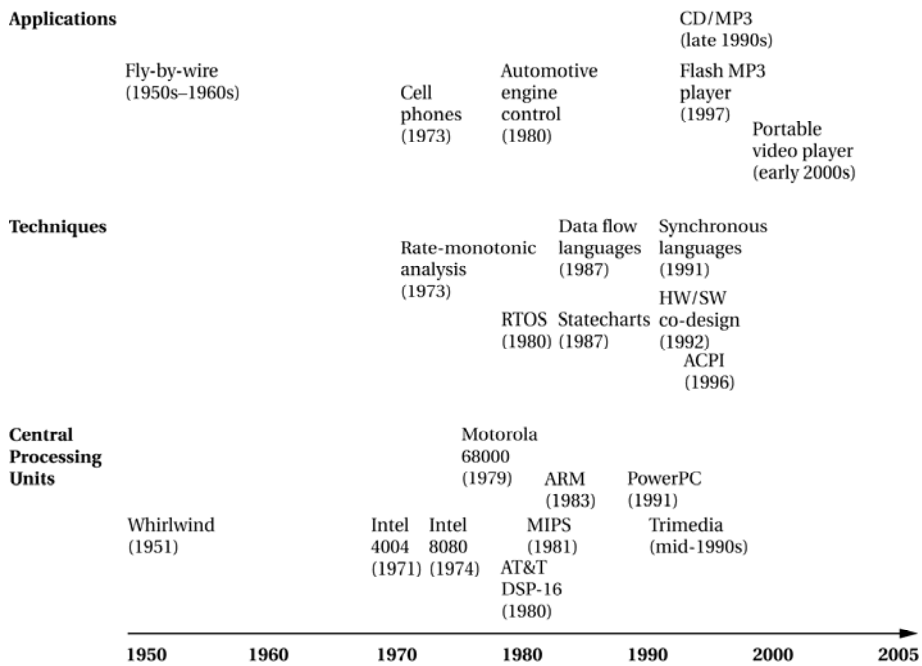
Modeling

- Modeling helps us predict the consequences of design decisions.
- Models help us work faster (once we have the model).
- We can afford to use models if we can reuse them in several designs---methodology relies on and enables modeling.

Disciplines in embedded computing

- Core areas:
  - VLSI, SoC, MPSoC
  - Real-time computing.
  - Hardware/software co-design.
- Closely related areas:
  - Computer architecture.
  - Software engineering.
  - Low-power design.
  - Operating systems.
  - Programming languages and compilers.
  - Networking.
  - Secure and reliable computing.

History of embedded computing

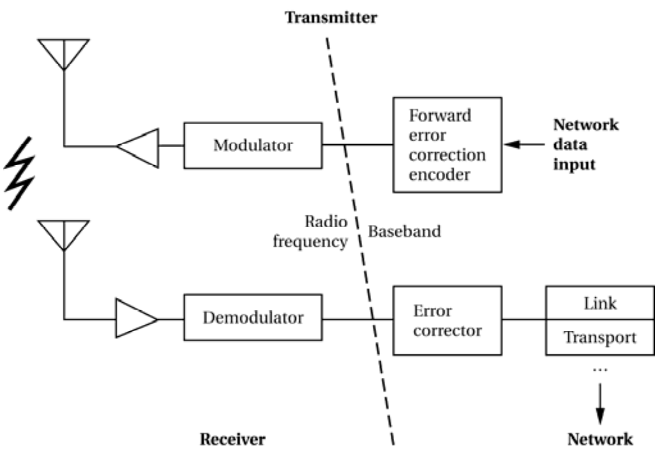


Application Examples

- Radio and Networking
- Multimedia
- Vehicle Control and Operation

Radio and networking

- Modern radio systems carry digital information.
- Perform modulation/ demodulation and error correction.
- May also be closely tied to a networking stack.



Seven layers of the OSI network stack

1. Physical: Electrical, physical.
2. Data link: Access, error control across a single link.
3. Network: End-to-end service.
4. Transport: Connection-oriented service.
5. Session: Control activities.
6. Presentation: Data exchange formats.
7. Application: Interface to end use.

Networks and embedded systems

- An increasing number of embedded systems connect to the Internet.
  - Resource management.
  - Security.
- Many specialized networks have been developed for embedded systems:
  - Automotive.
  - Device control.

Radio and software radio

- Wireless receivers (radios) perform several basic functions:
  - Demodulate the signal.
  - Detection bits.
  - Correct errors.
- Software radio performs at least some of these functions using software on CPUs.
- Software defined radio (SDR) may be all software or a mix of HW and SW.

Tiers of software-defined radio (SDR)

Tier-0: a traditional radio implementation in hardware.

Tier-1: Software Controlled Radio (SCR), implements the control features for multiple hardware elements in software.

Tier-2: Software Defined Radio (SDR), implements modulation and baseband processing in software but allows for multiple frequency fixed function RF hardware.

Tier-3: Ideal Software Radio (ISR), extends programmability through the RF with analog conversion at the antenna.

Tier-4: Ultimate Software Radio (USR), provides for fast (millisecond) transitions between communications protocols in addition to digital processing capability.

Radio operations

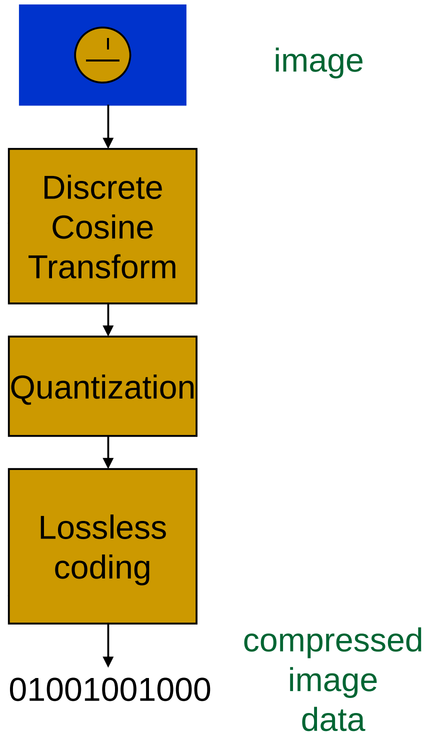
- Modulation:
  - Combinations of modulation variables (frequency, phase, amplitude) form symbols.
  - Symbols may be viewed as a constellation.
- Error correction:
  - Performed on raw bit stream to produce data payload.
  - Basic techniques like parity are often not powerful enough for noisy radio channels.
  - Viterbi method is widely used.
  - Example high-performance codes: turbo coding, low-density parity check (LDPC).

Multimedia

- Image compression: Each image is coded separately.
- Video compression: Takes advantage of correlation between successive frames.
- Perceptual coding: lossy coding, throws away information that will not be noticed.

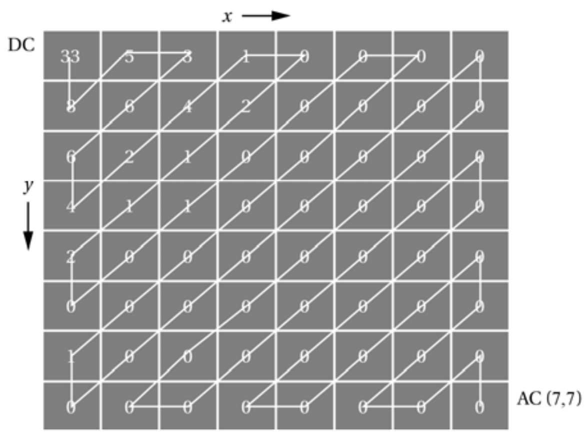
JPEG

- Discrete cosine transform (DCT) performed on 8 x 8 blocks typically, puts image into frequency domain.
- Quantization determines what image data to throw out.
- Lossless coding reduces the size of the representation.



JPEG zigzag pattern

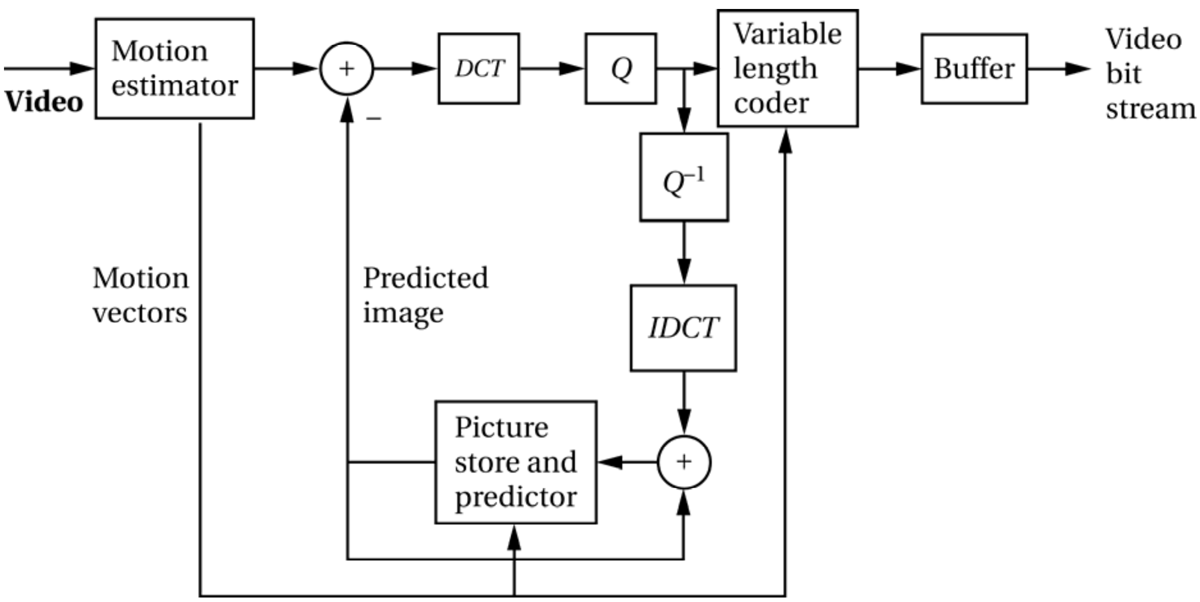
- After quantization, transform coefficients must be sent to lossless coder.
- Sending coefficients in zigzag pattern moves from low to high spatial frequencies.
- High frequency coefficients are more likely to be zero, producing strings that are easier to Huffman code.



Video compression standards

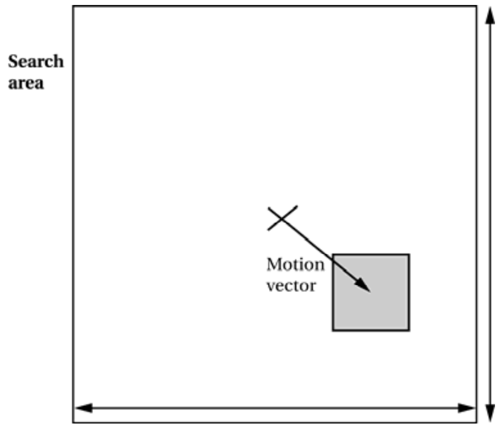
- Makes use of image compression techniques.
- Adds:
  - Support for frame-to-frame coding.
  - Audio streams, data, etc. controlled by a system steram.
- Two major families:
  - MPEG for broadcasting.
  - H.26x for videoconferencing.
- H.264/AVC combines techniques from both traditions.

MPEG-1/2 style compression



Motion estimation

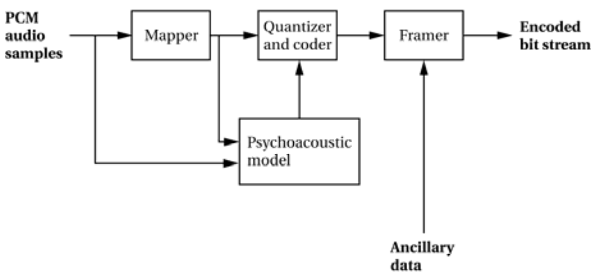
- Motion estimation compares one frame to another frame.
  - Generally performed on 16 x 16 macroblocks.
- Use 2-D correlation to find new position of a macroblock in the other frame.
- Transmit a motion vector to describe motion.



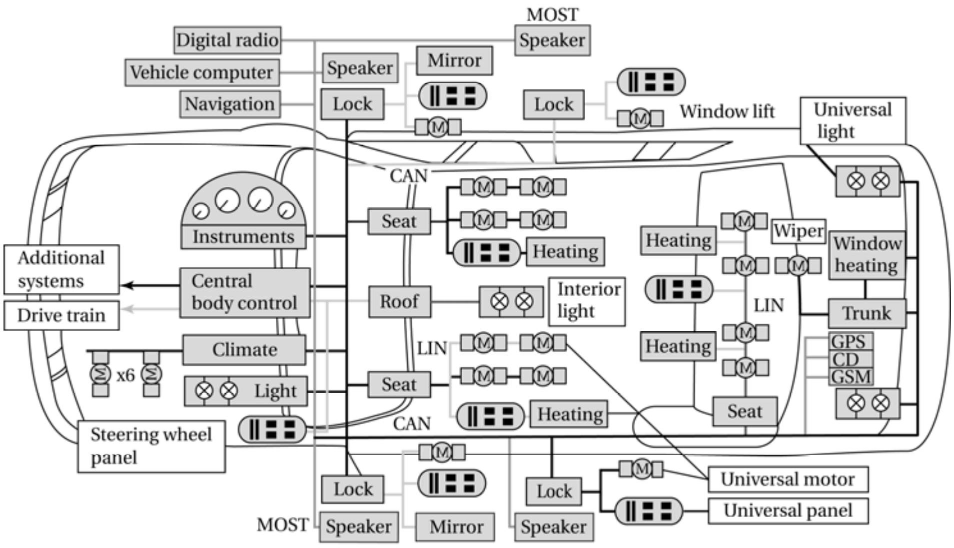
$$SAD = \Sigma_x \Sigma_y | S(x,y) - R(x,y) |$$

Audio encoding

- Perceptual coding models the human auditory system to predict what information can be thrown away.
- Subband decomposition helps improve the compression ratio.
- MP3 = MPEG-1 Audio Layer 3.



Automobiles as distributed embedded systems



CAN	Controller Area Network
GPS	Global Positioning System
GSM	Global System for Mobile Communications
LIN	Local Interconnect Network
MOST	Media-Oriented Systems Transport

Automotive and aviation electronics

- Some functions are safety-critical.
- Must operate in real-time.
- Must fit within power budget (limited by generator).
- Must be lightweight to fit within vehicle weight budget.

Sensor networks

- Used to gather, process data in the field.
- Ad-hoc networks: must set themselves up without intervention of network manager.
- Often battery powered, very tight energy budget.
- Generally wirelessly networked.