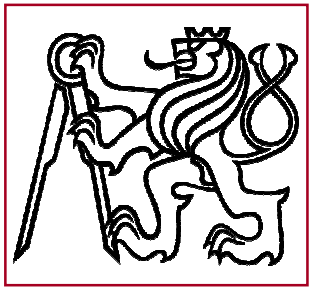
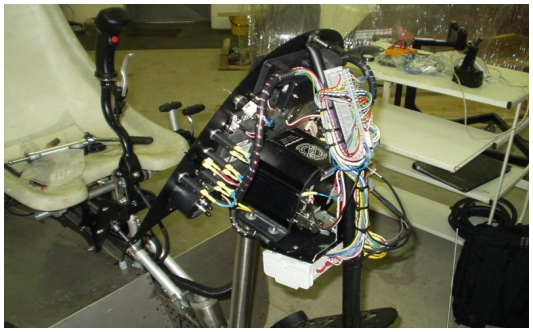
Y35PES

Programming for Embedded Systems

Ondřej Špinka, Pavel Němeček
spinkao@fel.cvut.cz nemecp1@fel.cvut.cz
http://dce.felk.cvut.cz/pes



Embedded System Definition



- An Embedded System is a control system that forms an integral part of the controlled machine
- IEEE definition: A computer system that is a part of a larger system and performs some of the requirements of that system.

General Properties of an ES

An ES should be:

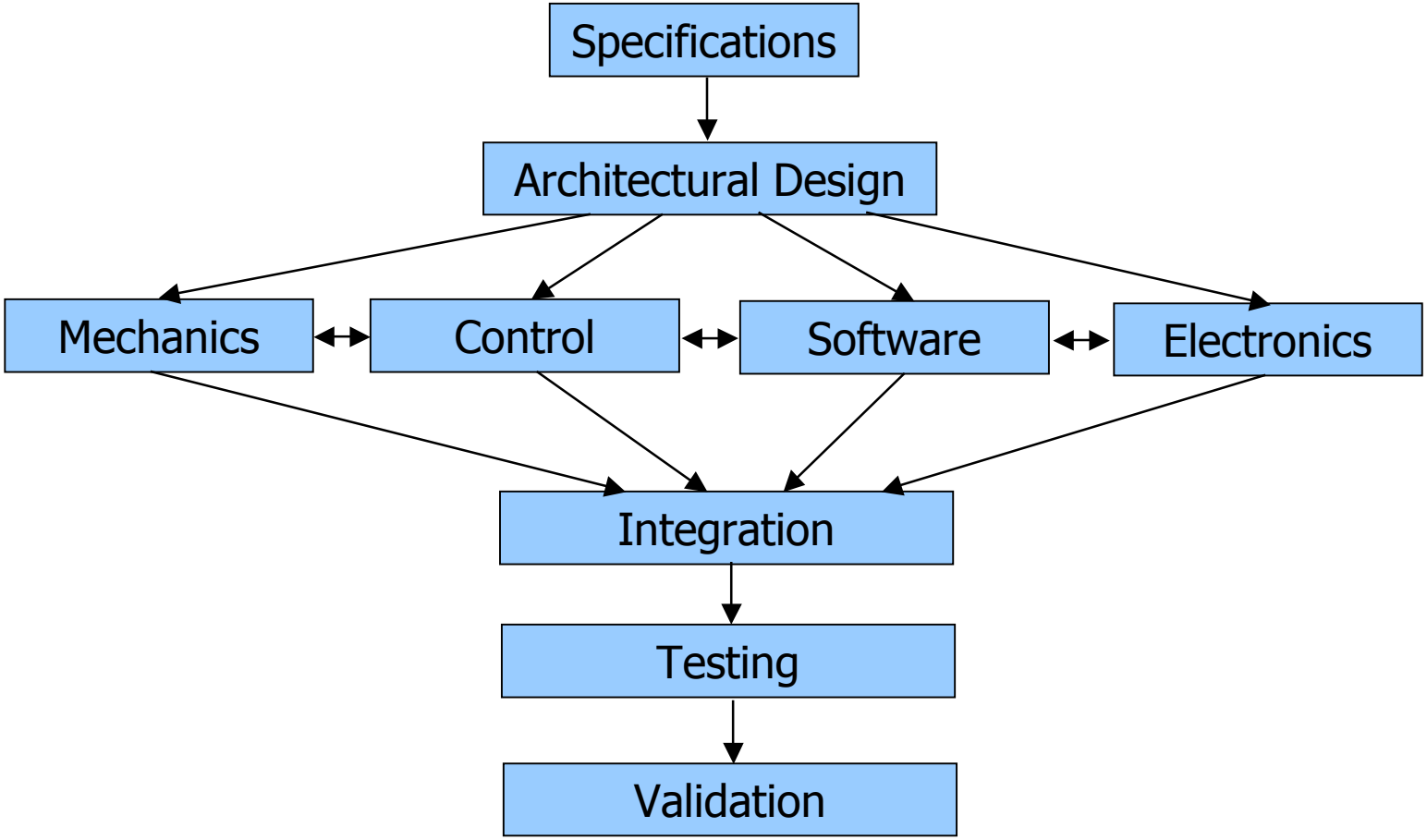
- Rugged and robust
- Reliable and safe
- Compact, lightweight
- Low power consuming
- Affordable

Following Considerations should be taken when designing an ES:

- Functionality and performance (computation power demands)
- Connectivity and dependability
- Real-time constraints
- Demands on reliability and safety (redundancy)
- Modularity
- Tight environment relation (mechanical constraints, temperature range etc.)

ES Design Phases

- Multidisciplinary effort, involving engineers from various fields.



Mechanical and Electrical Demands on ES (1)

As was mentioned at the beginning of this course, the embedded systems form an integral part of the controlled machine, and must often operate in a very harsh environment. Usually there are also very strict demands on the safety and reliability of those systems. This forms some special demands on the design of those systems. Let us start with a very short description of mechanical and electrical requirement an embedded system must usually fulfill.

Mechanical and Electrical Demands on ES (2)

From mechanical point of view, an ES must usually be:

- Mechanically rugged, i.e. it must be able to withstand shocks, vibrations, humidity, various fluid leakages etc.
- Temperature resistant – three categories of ES exist, specifying required temperature range according intended ES usage
  - **Normal range** 0…+70°C
  - **Industrial range** -40…+85°C
  - **Military range** -55…+125°C
- Compact and lightweight

Mechanical and Electrical Demands on ES (3)

Mechanical solutions:

- Robust, yet lightweight housing, often water-resistant, is required
- PCBs are treated with protective lacquer coating
- Big and/or heavy parts must be screwed or sealed to the PCB
- Rugged connectors with locks are used and are often sealed (this is extremely important, as connectors are usually the weakest part of the whole system!)
- Screwed joints are sealed with appropriate screw sealer
- Special wiring is used
- All electrical parts must fulfill required temperature range

Mechanical and Electrical Demands on ES (4)

From electrical point of view, an ES must usually fulfill following demands:

- Accept noisy / un-stabilized power supply
- Low power consumption (especially for battery-operated devices)
- Noise-tolerant on data lines
- Tolerate short circuit and excessive voltage (persistent / momentary) on each input / output pin (very strict norms exist especially for avionics / automotive components)
- Fulfill EMC demands

Mechanical and Electrical Demands on ES (5)

Electrical solutions:

- Internal power filtering / stabilization
- Various standby / sleep modes, switching off currently unneeded devices
- Filtering, photo-coupling of analog / data lines
- Input / output pins protection (pair of anti-parallel diodes to ground / Vcc, protective resistors, photo-coupling etc.)
- Interference elimination techniques are very complex and detailed description of the methodology would be out of scope of this course

Reliability and safety of ES (1)

By system **reliability** we usually understand the probability of the system failure. The system **safety** express the consequences of a failure; If an error is safe, it means that it would compromise the system performance, but it would not lead to grave consequences. Usually, there is a backup / failsafe mechanism in place to take care of a safe error.

We say that a system is **fault-tolerant**, if it can operate properly even when a failure occurs (possibly with some performance loss). **Graceful degradation** (or **controlled degradation**) means that a failure would cause a controlled performance degradation of a system, but would not lead to ultimate crash inevitably.

Reliability and safety of ES (2)

How to maintain system safety?

Basically, we can improve system safety by implementing some fail-safe devices and measures. The most common are:

- **System watchdog** – prevents software hang-ups. It would reset the system in the case of software failure.
- **Error management system** – implementing safety handlers for safe errors and controlled degradation
- **System redundancy** – there is a backup for some or all parts of the system. This backup could stand for the faulty component if needed. The backup could be either **hot** (running all the time) or **cold** (switched on only in case of main system failure). In the most safety-critical systems (e.g. life support systems, critical control systems) there is usually a hot **triple-redundancy** with **majority decision mechanism**.

Reliability and safety of ES (3)

How to maintain system reliability?

The reliability of a system can be improved by thorough testing / modification process. Naturally, both HW and SW influence the overall reliability of the system. Partially, HW and SW might be tested separately; however, most convenient approach is to test the system as a whole. Following two approaches serve to validate and test the system design:

- **Processor-in-the-loop** testing serves to test the SW thoroughly. Inputs of the system are simulated and responses of the system are evaluated by various means (timing, jitter, accuracy…). Also the failures might be simulated in the process. Usually, several test-cases are prepared and batch-run.
- **Hardware-in-the-loop** testing involves the system as a whole. Complete system is connected in exactly the same manner as it would be in the real application. Simulated signals (electrical or mechanical) are acting directly on the inputs/sensors of the tested system. The system response is evaluated in the same manner as by processor-in-the-loop testing. This is a more complex and thorough test compared to processor-in-the-loop, but it is more time-consuming and expensive, and requires special equipment.

Reliability and safety of ES (4)

How to maintain system reliability?

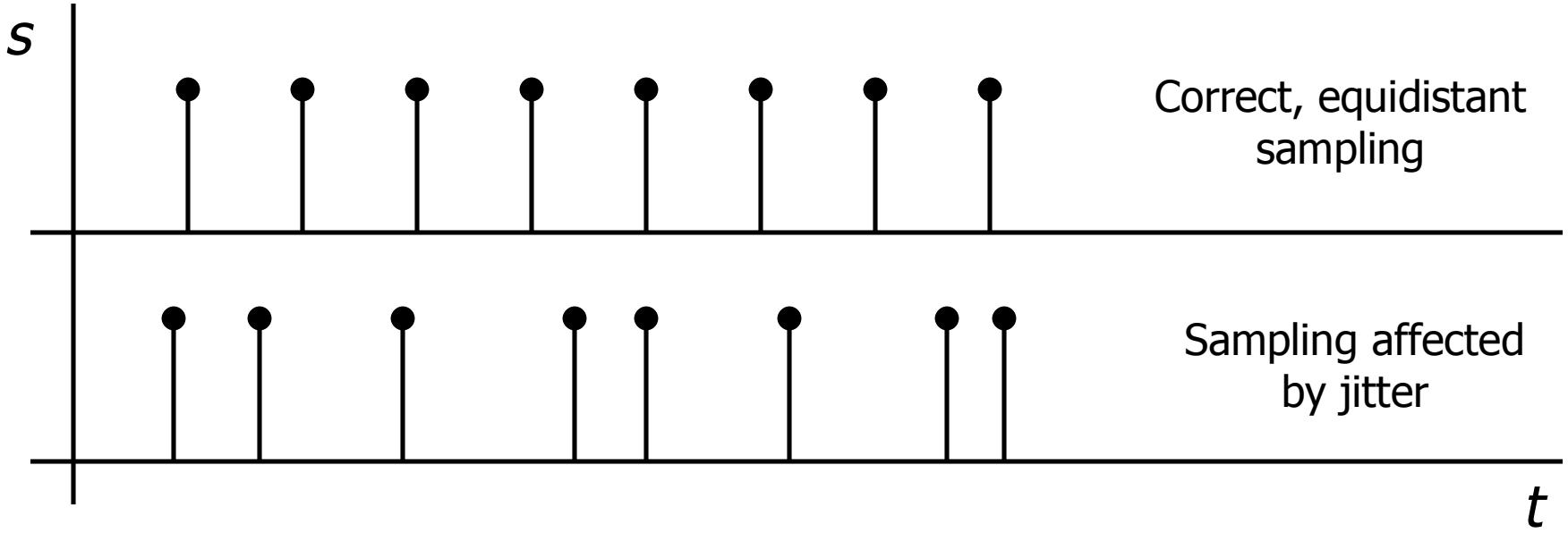
Hardware tests focused on the long-term reliability of the system usually involve some special equipment and are very time-consuming and expensive. Mechanical rigidity of the system might be tested using various test stands, e.g. **vibration test stand** etc. Temperature ranges and humidity resistance are tested in a **climatic chamber**.

Also electrical characteristics, EMC and resistance of the system to electrical shocks are tested. Methodology and more detailed description of those tests are out of the scope of this course.

Real-Time Operations (1)

By the term **real-time response** of a system we mean that the response of the system to an event is **guaranteed** to occur within a certain time.

This is often a natural demand for embedded systems, especially for control and measurement systems. A control/measurement system must ensure to maintain a steady **sampling period**, without variations of any kind (called **jitter**), as any discrepancy in the consecutive sampling times would affect the control algorithm performance.



Real-Time Operations (2)

In other words, correctly designed real-time system should guarantee a response/action time to lie in a relatively tight region. The response/action should be not early, nor late.

Often, the real-time constraints put to certain operations vary. Some of them are very strict (**hard**), requiring a very precise timing, on the other hand some of them might be **soft**, requesting only vague timing demands (for example that the response of the system should not be later than a specified time, without any request regarding the earliness).

Usually, the timing demands on sampling period / control action delivery are hard. On the other hand, the timing demands on communication with other systems might be soft (but might be also hard in certain cases).

Real-Time Operations (3)

What can influence the real-time behavior of a computer system:

In this part, we shall restrict ourselves to **system-less** computer systems only. When any kind of Operating System comes into play the things get a lot more complicated. The analysis of a operating system real-time properties exceeds the scope of this course.

The first natural demand is that the system must have enough computational power to be capable to perform all required operations within the period of a hard real-time task.

Then the interrupts must be taken into account. The interrupts inherently prolong the "normal" program run, but also other interrupts (with lower priority). A thorough analysis of the worst-case event sequence, generating the longest delay, must be performed in order to evaluate the worst-case response of the system. On the other hand, a certain mechanism must also be implemented to prevent excessive earliness (if required), to ensure correct measurement/action delivery.

Real-Time Operations (4)

Sufficient programming techniques:

Because real-time operations are mostly interrupt-driven, we shall talk mainly about interrupts.

- The interrupt handlers should be kept as simple and fast as possible to prevent excessive interrupt latency.
- Interrupt occurrence must be carefully evaluated, worst-case scenarios estimated and consequently correct priorities must be assigned to respective interrupts.
- Most time-critical and jitter-sensitive operations shall be implemented directly using the TPU hardware if possible – i.e. PWM generation, pulse counting / duty cycle / frequency measurement etc.
- If possible, measurement and action computation / delivery should be synchronized by using mutual time source in control systems (timer driven, either external or internal)
- If it is necessary to use multiple timers, those timers should be synchronized somehow if possible (exact techniques how to do that excess the scope of this course)

Lecture Summary – Essential Things to Remember

- An **Embedded System** is a **computer control system** that forms an **integral part of a controlled machine**.
- There are three temperature range categories for ES: normal, industrial and military.
- By system **reliability** we usually understand the probability of the system failure. The system **safety** express the consequences of a failure.
- Basically, two complex approaches exist to test an ES thoroughly: **processor-in-the-loop** (SW test) and **hardware-in-the-loop** (complex).
- By the term **real-time response** of a system we mean that the response of the system to an event is **guaranteed** to occur within a certain time. Real-time constraints might be **soft** or **hard**.
- Unwanted variations in the **sampling period** are called **jitter**.