Embedded Systems Lab 3 - HS 2020
28.10.2020

Andreas Biri

Lab structure

- **Goal of today's lab:**
  - Gather hands-on experience with an embedded **operating** system.
- **Agenda:**
  - Wednesday   16:15 - 18:00   Introduction (recorded) and questions
  - Friday   16:15 - 18:00   Questions & Answers
- **Available assistants:**
  - Andreas Biri - TA
  - Adrian Schneebeli - SA
  - Luca Rufer - SA

Lab structure

- **Interactions:**
  - **Exercise Zoom:** Questions can be asked throughout the lab in this room by raising your hand. Please feel free to write in the chat in case we overlook your question.
  - **Help Zoom:** Student assistants are available throughout the session for 1-on-1 meetings under the Zoom Meeting ID 917 6971 5701 (labs only).
  - **Matrix Chatroom:** Questions that are relevant for everyone can be asked in the Matrix chatroom where the responsible assistants can answer as quickly as possible.
  - **In-person:** Students can come to ETZ D96 to ask questions in person.

Goals

- Introduce a **real-time operating system (RTOS)**

- Learn what **tasks** are and how to create them

- Get to know **task states** and priorities

- Implement inter-task communication using **queues**

- Learn how to handle **critical sections**

Introductory questions - Question 1

Why do we need an operating system?

☐ Because it simplifies programming

☐ Because it allows us to truly run programs in parallel (i.e. at the same time)

☐ Because it makes our program execution more efficient

Introductory questions - Question 1

Why do we need an operating system?

- ✓ Because it simplifies programming
- ✗ Because it allows us to truly run programs in parallel (i.e. at the same time)
- ✗ Because it makes our program execution more efficient

# Introductory questions - Question 2

What makes using tasks tricky?

☐ They can potentially interfere with each other through shared resources

☐ They are very hard to program

☐ They can create dead-locks and introduce dependencies

Introductory questions - Question 2

What makes using tasks tricky?

- ✓ They can potentially interfere with each other through shared resources
- ✗ They are very hard to program
- ✓ They can create dead-locks and introduce dependencies

Tasks
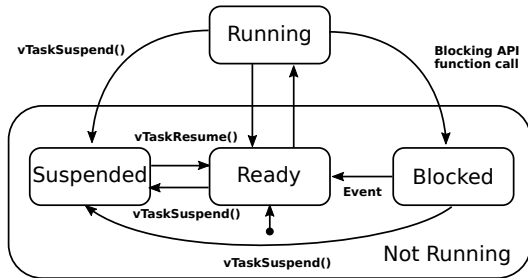
- OS allows splitting up the application into independent tasks

- Tasks can be individually put into new states depending on their behaviour

- Priority signals scheduler which task to prefer

Critical sections

- **Pre-emption:** Tasks can be interrupted by other events

- **Time slicing:** Scheduler shares processing time between tasks

- However, this might stop time-critical interactions (e.g. with peripherals, such as when reading from a sensor)

- **Mutex:** Guarantees _exclusive_ access to a shared resource

Related chapters from the lecture

- Ch. 5: **Embedded Operating Systems**

- Ch. 5: FreeRTOS Task Management $\rightarrow$ Task 1
  - Task functions
  - Creation of tasks
  - Task handles and deletion at runtime

- Ch. 7: Resource Sharing $\rightarrow$ Task 2
  - Example of shared resource conflict
  - Mutual exclusion

Related chapters from the lecture

- Ch. 5: FreeRTOS Timers and FreeRTOS Task States → Task 2
  - Delay
  - Blocked, Ready, Running States
  - Events

- Ch. 7: Communication and Synchronization → Task 3
  - Communication between tasks
  - Queues

FreeRTOS

- The lab is self-contained but cannot cover everything about FreeRTOS

- Supplementary material can help to understand the content and go beyond
    - https://www.freertos.org/
    - FreeRTOS Reference Manual v9.0.0.
    - Book: "Mastering the FreeRTOS Real Time Kernel"

Closing question

What is the length of a `uint32_t` variable?

☐ 4 bytes

☐ The same length as a `long` variable

☐ `sizeof(uint32_t)`

Closing question

What is the length of a `uint32_t` variable?

- ✓ 4 bytes
- ✗ The same length as a `long` variable
- ✓ `sizeof(uint32_t)`

Introduction is over

- The assistants are now available **until 18:00** to answer questions.
    - **Zoom:** Either ask in this channel or use the Zoom Meeting ID 917 6971 5701 to talk individually with an assistant.
    - **Matrix-Chatroom:** Ask a question in the chatroom so other students can also profit from the response (or respond even faster).
    - **Email:** For individual questions, you can also reach me under `abiri@ethz.ch`.
- On **Friday from 16:15 - 18:00**, we will also be available for questions.

Happy coding!

Questions?

Andreas Biri

ETZ G75

+41 44 632 08 73

abiri@ethz.ch