

Embedded and Ambient Systems

2022. 11. 22.

Portable code, virtualization



Mérés-technika és
Információs Rendszerek
Tanszék

Portable code

- A certain functionality can be reused in several environment, e.g.:
 - Different HW
 - Different compiler
 - Different operation system,
 - Different library functions
- Portability requirements of a code:
 - Only slight HW dependence is allowed
 - Contains only few HW specific parts (as few as possible)
 - Exploits only minimal specialties of a certain compiler or SW environment (IDE)
- Advantages:
 - Most of the functions are implemented in SW -> code reuse is important
 - Development time can be reduced, code parts can be reused with small overhead
 - Less amount of errors: a tested, frequently used code should be reused
- Structured programming (see: robust- and structured programming lectures)

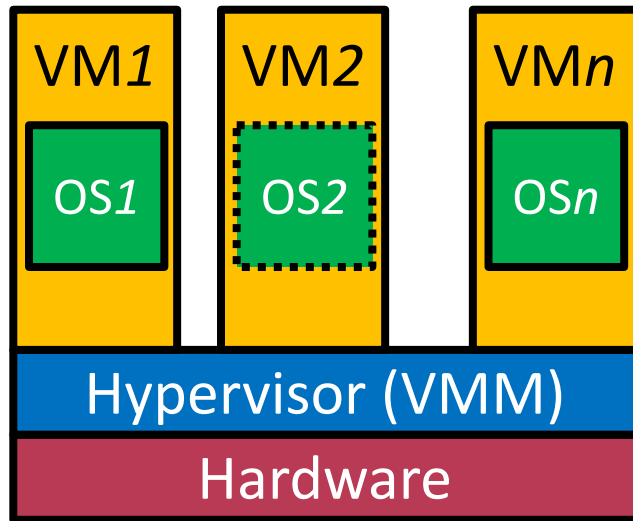
Portable code

- Operation systems (OS):
 - A large step toward portability
 - The application program uses only the functions of the OS, low-level HW handling remains hidden
 - Tailoring of the OS to a certain processor (porting) is done by someone else (e.g. processor manufacturer)
 - Proliferation of Linux in embedded systems:
 - Supports real-time operation
 - Scalable
 - Low-level programming is possible

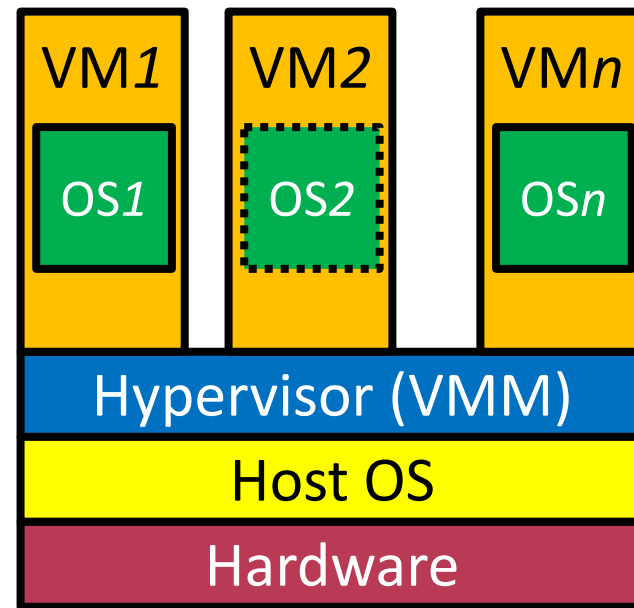
Virtualization

- The application SW shall be run on the processor via a uniform platform
 - In a PC environment a good example is Java (JVM: Java Virtual Machine), that offers access to the resources of the computer independently of the OS
- Proliferation of visualization can be seen not only in PC but also in embedded environment
 - Processors become more and more powerful
 - Wide spectrum of the processors
 - Several applications
 - Time to market is an important issue

Model of operation



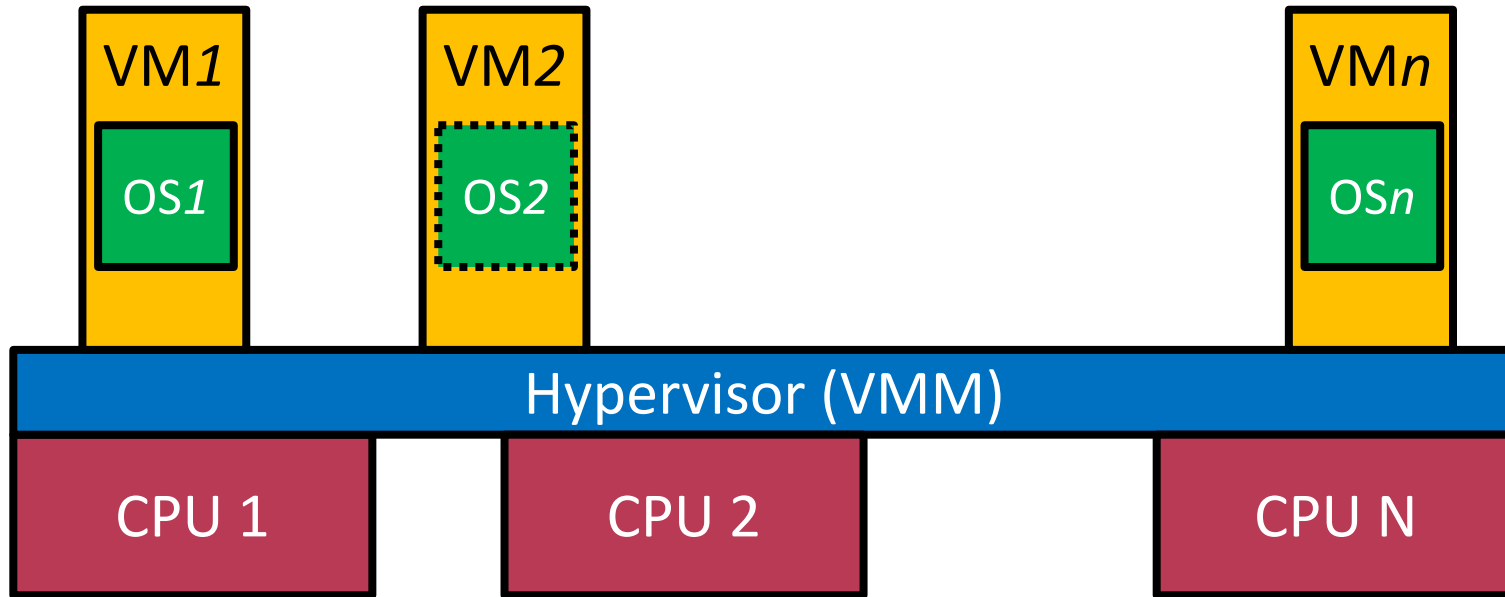
Type 1: hypervisor runs directly on HW



Type 2: hypervisor runs on a host OS

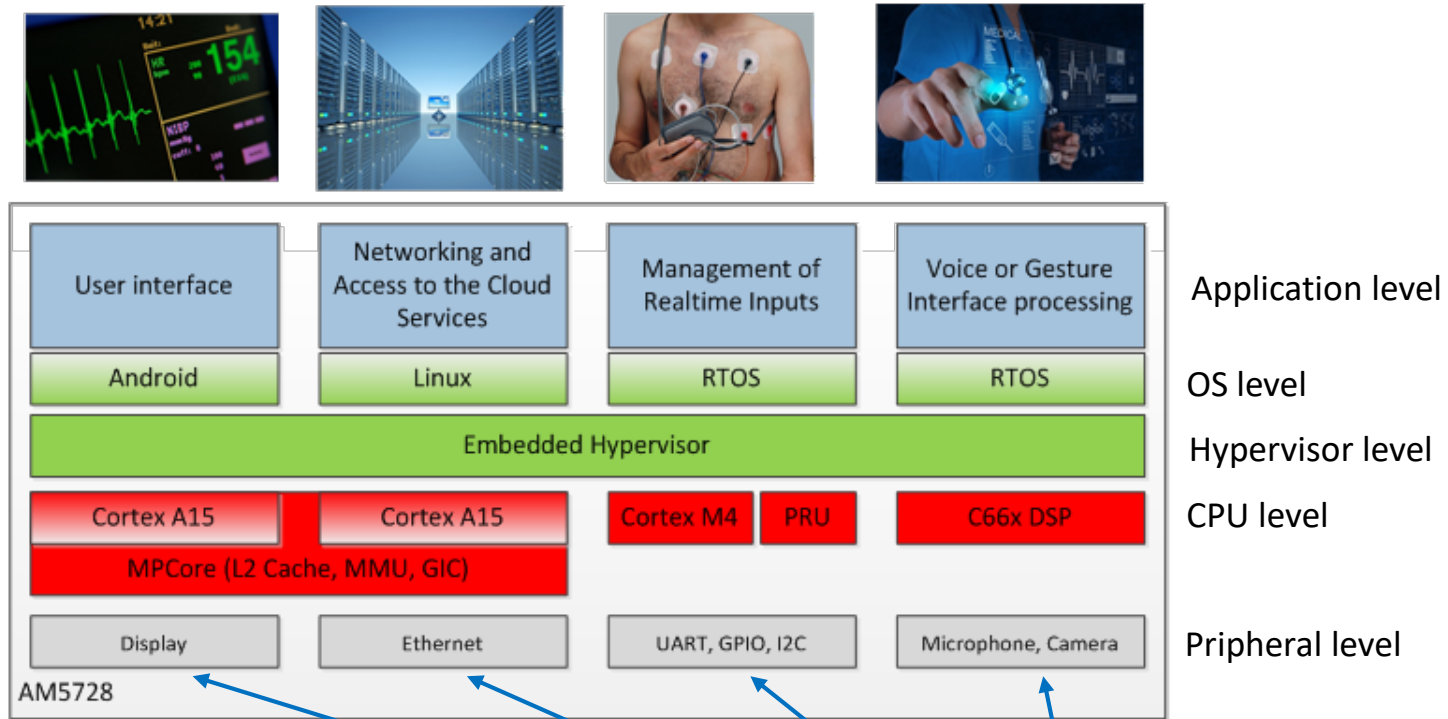
- Hypervisor = virtual machine monitor (VMM)
 - Supervises the virtual machines and program components
 - Offers a supervised access to the system resources
- VM: Virtual Machine
 - May run an operation system or a simple code

Extended model (e.g. SoC)



- Code can be run on *or tasks can be distributed among* more than only one CPU platform
- Different processors may be dedicated to different tasks (e.g. communications, signal processing, graphics)

Extended model (e.g. SoC)



- Code can be run on *or tasks can be distributed among* more than only one CPU platform
- Different processors may be dedicated to different tasks (e.g. communications, signal processing, graphics)

Hypervisor

■ Tasks:

- Distribution of resources
- Efficient communications between virtual machines
- Scheduling
- Interrupt distribution
- Energy management

■ Requirements:

- Security
 - SW component must not get out of the surveillance of the hypervisor →
→ HW-based memory protection
- Small size not to consume large memory
- Rapid execution of tasks not to influence real-time operation

Virtualization

- More than one OS can run in a separated manner:
 - Protection to avoid one system affect an other system
 - The safety-critical and non-safety-critical tasks shall be handled separately
- Protection:
 - Memory: virtual address ranges, no direct memory access is allowed
 - Timing: hypervisor has an own timer, no application has the right to run arbitrarily long
 - OS services
 - HW resources: access is possible only via hypervisor
 - Unreliable codes shall be run in a separated manner (sandbox principle)

Disadvantages of virtualization

- Disadvantages:
 - Several abstraction layer -> larger uncertainty in timings
 - Increased resource consumption:
 - Memory
 - Processor time
 - More difficult to assure resource distribution (e.g. communications)
 - Virtualization platform must be free of errors and malfunctions to provide reliable operation for system components rely on the platform

HW-based support

- The entire functionality of protection and separation cannot be assured by only SW, therefore HW-based support is needed
- HW-based support:
 - Different privileges inside the processor (e.g. special registers that can be accessed in hypervisor mode)
 - Memory management