Writing drivers with power management support

Objectives

- Configure the Linux kernel for optimal power management.
- Understand how Linux manages the power
  - To write drivers integrated in this scheme
  - To optimize low power modes
- Install and use power-management utilities

Labs are conducted on target boards, that can be:
Atmel ARM9-based boards, with Lauterbach JTAG probes.
We use the last linux kernel, as supported by the board.

Course Environment

- Theoretical course
  - PDF course material (in English) supplemented by a printed version.
  - The trainer answers trainees' questions during the training and provide technical and pedagogical assistance.
- Practical activities
  - Practical activities represent from 40% to 50% of course duration.
  - Code examples, exercises and solutions
  - One PC (Linux ou Windows) for the practical activities with, if appropriate, a target board.
    - One PC for two trainees when there are more than 6 trainees.
  - For onsite trainings:
    - An installation and test manual is provided to allow preinstallation of the needed software.
    - The trainer come with target boards if needed during the practical activities (and bring them back at the end of the course).
- Downloadable preconfigured virtual machine for post-course practical activities
- At the start of each session the trainer will interact with the trainees to ensure the course fits their expectations and correct if needed

Prerequisite

- Good practice of C programming on Linux
- Basic knowledge of Linux kernel and driver programming (see our D3 - Linux Driverscourse)

Target Audience

- Any embedded systems engineer or technician with the above prerequisites.

Evaluation modalities

- The prerequisites indicated above are assessed before the training by the technical supervision of the traineein his company, or by the trainee himself in the exceptional case of an individual trainee.
- Trainee progress is assessed in two different ways, depending on the course:
  - For courses lending themselves to practical exercises, the results of the exercises are checked by the trainer while, if necessary, helping trainees to carry them out by providing additional details.
  - Quizzes are offered at the end of sections that do not include practical exercises to verifythat the trainees have assimilated the points presented

- At the end of the training, each trainee receives a certificate attesting that they have successfully completed the course.
  - In the event of a problem, discovered during the course, due to a lack of prerequisites by the trainee a different or additional training is offered to them, generally to reinforce their prerequisites,in agreement with their company manager if applicable.

Plan

First Day

Reminders on kernel programming

- Reminders on kernel module development
- Kernel objects

Exercise: *Writing a kernel module creating and using kernel objects and sets*

- The sysfs file system

Exercise: *Interacting with a kernel module through a kernel object and the sysfs file sytem*

Power-management : APM and ACPI

- The APM standard.
- The ACPI standard.

Exercise: *Writing a user program to shut down the system.*

Power-management, suspend and hibernate

- Freezing tasks
- Suspend in RAM
- Suspend on Disk
- Hibernate
- Debugging power management

Exercise: *Writing a kernel module reacting to suspend in RAM and system wakeup*

Second Day

Power management in drivers

- Device states vs System state
- Putting devices in low-power modes
- Power-management operations in drivers
- Controling system state changes from drivers

Exercise: *Adapting a device driver to put the device in suspended state when unused*

Cpu frequency governors

- The cpufreq user interface.
- The various governors available.

Idle Cpu governors

- The Idle task.
- The Idle device concept.
- CpuIdle governors.

Renseignements pratiques

Inquiry : 2 days