[redacted]

Graphical User Interfaces for Embedded Linux

Objectives

- Understand the different layers associated with the graphic interface system
- Learn how to use the Linux frame buffer
- Discover the hardware abstraction layers
- Learn to use the main graphical toolkits (widgets) taretting embedded Linux
  - QT
  - FLTK
  - WxEmbedded
  - MiniGUI
  - ...
- Learn how to link commands to GUI elements

Labs are conducted using the System Workbench for Linux - Basic EditionIDE.

Hardware

- A Linux PC for two trainees, with an ARM based target board
- Course material
- CDROM with documentation and tools used during the trainin

Prerequisites

- Knowledge of Linux user level programming (D0 - Linux user mode programmingcourse)
- Knowledge of embedded Linux (D1 - Embedded Linux with Buildroot and Yoctocourse)

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.
- At the start of each session the trainer will interact with the trainees to ensure the course fits their expectations and correct if needed

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 by quizzes offered at the end of various sections to verify that 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

The various system layers used by GUIs

- Kernel drivers: frame buffer, direct access, ...
- Hardware abstraction layers: X, GGI, DirectFB, ...
- Graphical toolkits (widgets): QT, GTK, FLTK, ...
- Window manager: matchbox, sawfish, ...
- Desktop environments: GNOME, KDE, ...

The Linux frame buffer

- Selecting frame buffer support when building th ekernel
- Generic VGA support, VESA
- Provided features

The hardware abstraction layers

- X: standard server and embedded-specific versions (tiny-X, Kdrive, Nano-X)
- GGI: generic multiplatform GUI (Linux, Windows, MacOS/X ,...)
- DirectFB: hardware acceleration above the frame buffer

The graphic toolkits (widgets)

- Trolltech's QT
- GTK (Gimp ToolKit)
- WxEmbedded
- FLTK (Fast Light ToolKit)
- MiniGUI
- Nano-X (ex MicroWindows)
- Standard "desktop" versions and embedded-specific versions
- Graphical GUI design tools: QT designer, Glade, FLUID...

Linking commands to GUI items

- Using fork and exec
- Pipe communication
- Integration with widgets

Embedded GUI distributions for Linux

- The "familiar project"
- GPE Palm Environement
- Qtopia
- Pixil

Renseignements pratiques

Inquiry : 3 days