Yocto Embedded Linux training

Pierre Ficheux (pierre.ficheux@smile.fr)

December 2023

Training goals

- Understanding Embedded Linux principles (0.5 day)
- Understanding Yocto / OpenEmbedded principles (2.5 days)
- Writing sample recipes based on OSS standards
  - Autotools
  - CMake
  - Linux drivers
  - Device Tree
- Customizing existing Yocto recipes
- Using Yocto tools (SDK, Devtool, ptest, testimage, etc.)
- Building a Yocto “IoT” device !

Prerequisite

- Basic UNIX/Linux knowledge (using the shell !)
- Basic C language knowledge

Training environment

- Practical work on QEMU/ARM emulator
- The training is based on Yocto 3.1 (Dunfell, LTS)
- The Linux environment is Ubuntu 18.04 (VirtualBox)
- The training duration is 3 days, i.e. 21 hours course (7h/day)

Embedded Linux section (ARM board)

- GNU/Linux reminders
- GPL/LGPL licenses
- Introducing the cross-compilation
- Cross-compiling the Linux kernel
- BusyBox
- Using a “build system”
  - Pros and cons
  - Main tools (Buildroot, Yocto/OpenEmbedded, etc.)

Yocto section (QEMU/ARM)

- History (OpenEmbedded → Yocto)
- Main concepts : BitBake, metadata, layers, inheritance, etc.
- Creating the *core-image-minimal* image for QEMU/ARM
- Generated directories (`deploy` and `work`)
- Tuning and optimisation (`local.conf` / `bblayers.conf`)
- Creating a test layer
- Writing recipes (`.bb` files)
- Package management (OPKG)
- Autotools / CMake classes
- Static and dynamic dependencies
- Layer priority, extending recipes (`.bbappend`)
  - Applying patches
  - Using configuration fragments
- Device tree integration

Yocto section (QEMU/ARM)

- Kernel recipes and modules (the “module” class)
- Custom images and “packagegroup” class
- Testing a custom image with NFS-Root
- Creating a custom distribution (aka “distro”)
- Building and using the cross-toolchain (SDK/eSDK)
- Using “Devtool”
- Using SysvInit and systemd
- Using CI (“ptest” and “testimage”)

Provided software components

- The user name is “stage” and the password is “stage0”
- Use `sudo` for root access
- Everything is located in the `LE_Yocto` directory on the desktop

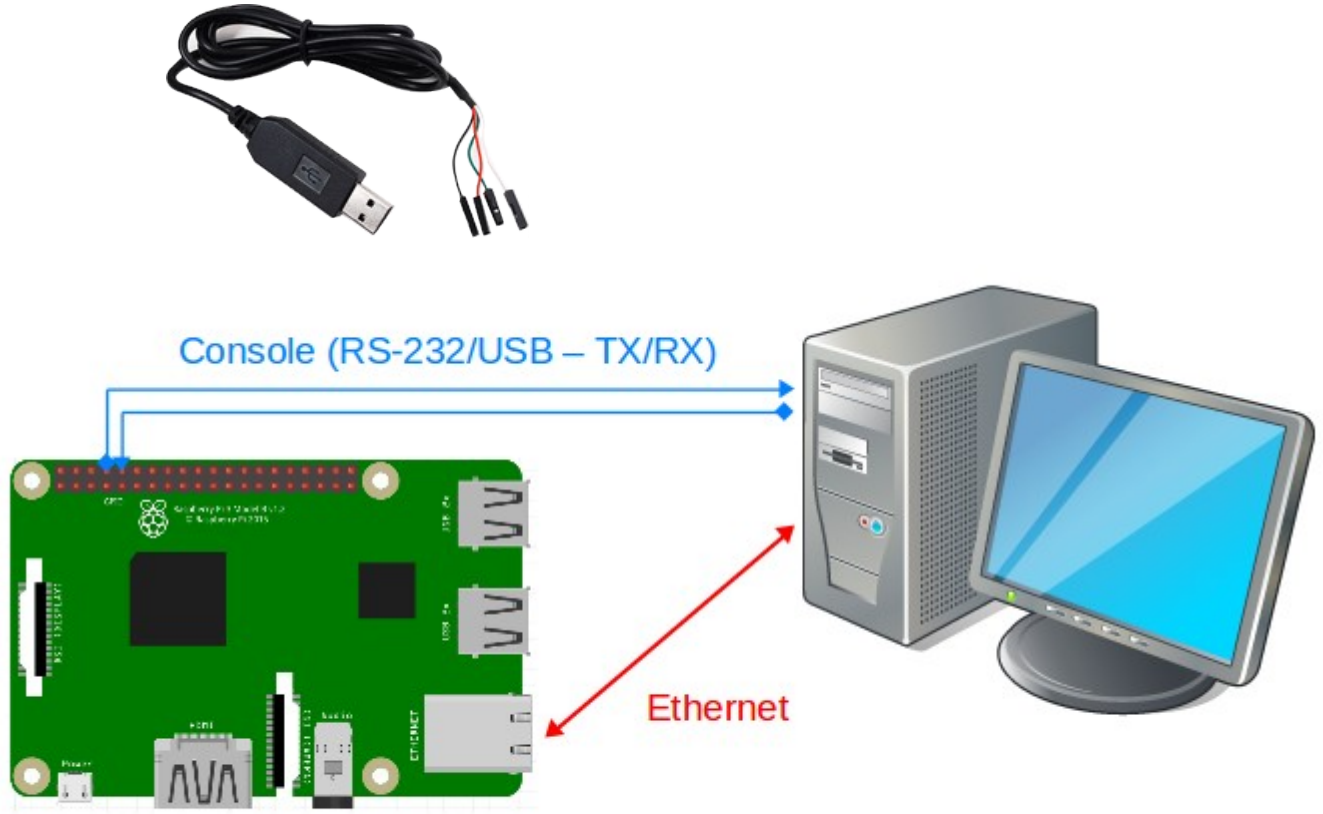
  `slides` = training slides

  `PW` = practical work

  `poky` = Yocto sources and binaries

  `misc` = training layers

Training configuration (Pi)



Training configuration (QEMU)