

foss.in/2010 - Fedora Embedded and QEMU/ARM hands on

Saleem Ansari

Fedora

17th December 2010

- 1 Agenda
- 2 Fedora Embedded SIG – Special Interest Group
- 3 Some background
- 4 A basic hands-on with QEMU/ARM
- 5 Fedora ARM Infrastructure
- 6 Questions?

Agenda

- Fedora Embedded SIG – Special Interest Group
- Some background knowledge
- A basic hands-on with QEMU/ARM
- Fedora ARM Infrastructure

Fedora Embedded SIG – Special Interest Group

- Goals - <http://fedoraproject.org/wiki/SIGs/Embedded>
 - Create high quality packages of cross-compilers and related tools
 - Provide packaging guidelines
 - Make distribution releases on sub architectures
- Targeted platforms/arches: ARM, AVR32, AVR, Blackfin, TIGCC, RTEMS, others
- For this talk we will focus on ARM architecture

Why Fedora on ARM architecture? Multiple reasons!

- 2007: ~ 98% of one billion mobile phones sold each year use at least one ARM processor.
- 2009: ~ 90% of all embedded 32-bit RISC processors are ARM processors.
- 2010: ~ 5 Billion ARM9 processors have been shipped so far
- ARM processors to enter server markets - Marvell's announcement
- ARM processor- Eagle
 - based on Cortex A15 architecture, it has hardware virtualization support!
- OLPC is a one big project which interested to use Fedora ARM port

Some background

- Platform / processor selection for a Linux/embedded project
 - at-least 32bit processor
 - processor should have a MMU
 - non-MMU based processors can use ucLinux
 - however you have to modify applications too
 - floating point unit
 - which optional and can be emulated by libraries

Why do we need a boot loader?

- There is a lot of work done by boot loader.
- Kernel assumes that a lot of system initialization is already done:
 - configuring memory sub-system.
 - loading kernel image at the correct memory address
 - loading initial RAM disk (optional)
 - initializing boot parameters
 - obtaining the ARM Linux machine type
 - initializing register variables for entry into kernel
- Writing code for all this is:
 - highly error prone, repetitive, cumbersome and boring
 - a readymade bootloader generally just fits in

Why do we need Linux Kernel?

- Linux, of-course, is Open Source
- Vendor Independence
- POSIX compliance
- Varied Hardware Support
- linux kernel has many device drivers already built by someone
 - provide us with well defined device interfaces
 - does the grunt work of hardware level interfacing with the devices
 - driver works on another hardware with usually very few changes

About QEMU

- QEMU is an emulator for popular CPU
 - x86, ARM, PowerPC, Sparc32/64, MIPS and ColdFire
 - AVR32 support is in progress at <http://repo.or.cz/w/qemu/avr32.git>
 - It also simulates peripheral devices, basically a complete System on Chip.
- What makes QEMU special for doing embedded systems development?
 - No need of actual hardware for prototyping.
 - Its greener- no need of any power source.
 - Many projects already use QEMU
 - Android, MeeGo, etc.

Hands-on. We will discuss the following:

- Development setup up
 - ARM toolchain, QEMU etc.
- What happens at system reset?
 - Initialization.
 - Bootloader.
 - Kernel.
 - Initial root filesystem.

System setup

- Fedora ARM Toolchain
 - ARMv5 only -
<http://fedoraproject.org/wiki/Architectures/ARM/CrossToolchain>
- QEMU/ARM
- mkimage - tool to create boot images
- other tools
 - bridge-utils (optional)
 - koji

What happens when an ARM based board is powered ON?

- Understand the memory map for RAM and ROM
- Startup code
 - First instruction
 - Vector table
 - Stack initialization
 - Relocating the RAM to a different address
 - Running from RAM

Hands on:

- Target hardware - VersatilePB - emulated QEMU
- Lets make a minimalistic linux based system on ARM
- Following steps are involved:
 - Compile and configure U-Boot boot-loader
 - Compile and configure Linux Kernel
 - Create a root file-system
 - We will use Busybox for shell command interface
 - Shove all of them into a one big chunk to be flashed onto memory

What we learn from this exercise?

- Its a great learning experience
- However its not productive in the long term.
- It doesn't scale to thousands of software packages.
- Fedora ARM port solves this problem:
 - Proven packaging technology already in use in many of the most popular distros
 - Reduce development time and time to market for your next device

Fedora ARM Infrastructure

- Seneca Centre for Development of Open Technology (CDOT) at the School of Computer Studies at Seneca College, Toronto.
- http://zenit.senecac.on.ca/wiki/index.php/Fedora_ARM_Secondary
- http://zenit.senecac.on.ca/wiki/index.php/Fedora_ARM_Koji_Build
- Latest news:
 - <http://fedora-arm.blogspot.com/>
 - <http://paulfedora.wordpress.com/>
- Current status page
 - <http://arm.koji.fedoraproject.org/status/>
- Recent video by Paul Whalen

Bulding packages for Fedora ARM

- Create or fix an existing SRPM package
- Schedule a build on ARM Koji to create RPM
- Check and fix if necessary
- ARM Koji web interface
- More on Howto use koji
 - <http://fedoraproject.org/wiki/Koji>

Contributing to Fedora/ARM

- fedoraproject.org/wiki/Architectures/ARM/HowToQemu
- Mailing list
 - <https://admin.fedoraproject.org/mailman/listinfo/arm>
- IRC - #fedora-arm on irc.freenode.net
- Issues not yet resolved:
 - https://bugzilla.redhat.com/show_bug.cgi?id=ARMTracker
- More info at:
 - http://fedoraproject.org/wiki/Architectures/ARM/Team_and_Develo

Current state of affairs in Fedora/ARM

- Fedora 13 release for ARM will hopefully be near Xmas.
- The default toolchain is still for ARMv5 and doesn't support hardfp so many of the modern ARM CPUs aren't optimally used
- Floating point support in the Fedora/ARM cross toolchain:
 - Discussion is on for which of soft/softfp or hardfp to select for upcoming ARMv7 toolchain
- OLPC (currently the only one eager to use Fedora ARM for a large scale project)
 - they seem to be aiming to jump from F-11 to F-14, for their next release
 - it's not clear if that includes XO-1.75 (the OLPC's ARM target).

Agenda
Fedora Embedded SIG – Special Interest Group
Some background
A basic hands-on with QEMU/ARM
Fedora ARM Infrastructure
Questions?

Questions?

Thank you!

- tuxdna at #fedora-arm on irc.freenode.net