

# Welcome to NCLUG

20.July.2004

Embedded Linux

T.Michael Turney

In November 2001, Jack Ganssle wrote a column entitled “Is Embedded Linux a Bust?” where he discusses his doubts about Linux's ability to gather any interest in the embedded world. The main problem with this article (and that of many others, I should add), is that it makes no difference between Linux's viability as an embedded OS and the viability of the “embedded Linux” market.

Karim Yaghmour,  
author of Building Embedded Linux Systems

Above quote is from article posted on O'Reilly  
dated, 9.June.2003  
<http://linux.oreillynet.com>

# What's My Line

- Contestant number 1
  - Hi, my name is T.Michael Turney and I am an Open Source Ambassador
- Contestant number 2
  - Hi, my name is...
- Our panel has a question...
  - What is an Open Source Ambassador?
- Contestant number 1
  - I'm glad you asked...

# Open Source Ambassador

- Believes in the philosophy of Open Source Software
- Understands that any software engineering project can benefit from being exposed to OSS
- Possesses the skills to clear most hurdles early in the project chronology
- Desire to evangelize the message

# My Linux Pedigree

- October 2000
  - Hired by Monta Vista Software as an FAE to help sell HardHat Linux
- May 2002
  - Authored App-Note on kernel debugging utilizing the Abatron BDI-2000 JTAG unit
- August 2002
  - Start working on first project as an embedded Linux consultant

# Current Open Source Projects

- DDD customization to form tighter integration with BDI-2000 JTAG hardware debugger
- Kernel.org port to WRS 8260 hardware platform
- Maintainer of [www.recipes4linux.com](http://www.recipes4linux.com)

# The Inside Story

- As market pressures force us to:
  - Deliver more complex products
  - Deliver more frequently
  - Deliver sooner
- Something has to give, one design team can't write all the code that is required in a typical product today

# The Inside Story continued

- If you accept the premise that your design team will not create all of the software that goes into the product you are working on
- You will either
  - Evaluate commercial software to license and use
  - Evaluate Open Source Software to license and use
- This presentation is intended to give you insights into how to evaluate Linux as an option for future embedded design projects.



# Embedded Linux

- **Different flavors of Linux**
- **Host System Issues**
- **Kernel Porting Issues**
- **zImage vs. vmlinux**
- **Debugging Issues**
- **Real-time vs. Non Real-time**
- **Embedded Distributions**

# Different Flavors of Linux

- Desktop vs. Embedded
- Components, by Architecture
  - Kernel
  - File System
  - X-Dev Tools

# Understanding a Linux Distribution

- Desktop Distribution (e.g., RHL 7.3)
  - Kernel (version is important)
  - Core filesystem components
  - Bootloader (lilo, grub)
  - Hardware probe (kudzu)
  - Optional packages (RPM files)
- What did I leave out?

# Understanding a Linux Distribution

- Embedded Distribution (e.g. MVL 2.1)
  - Cross-Development tools (X-Dev)
  - Kernel (version is important)
  - NFS Development filesystem
  - Tool to help build deployed filesystem
  - Missing
    - Bootloader
    - Hardware probe

# Linux, by Architect

- From [www.kernel.org](http://www.kernel.org)
  - Linux was first developed for 32-bit x86-based PCs (386 or higher). These days it also runs on (at least) the Compaq Alpha AXP, Sun SPARC and UltraSPARC, Mot 68K, PPC, PPC64, ARM, SH, IBM S/390, MIPS, HP PA-RISC, IA-64, DEC VAX, AMD x86-64 and CRIS.

# Linux Web Resources

- The following are referenced at kernel.org for some common embedded architectures
  - X86 : [www.kernel.org](http://www.kernel.org)
  - PPC : [www.penguinppc.org](http://www.penguinppc.org)
  - MIPS : [www.linux-mips.org](http://www.linux-mips.org)
  - ARM / StrongARM / Xscale :  
[www.arm.linux.org.uk](http://www.arm.linux.org.uk)

# Embedded Linux

- **Different flavors of Linux**
- **Host System Issues**
- **Kernel Porting Issues**
- **zImage vs. vmlinux**
- **Debugging Issues**
- **Real-time vs. Non Real-time**
- **Embedded Distributions**

# Host System Issues

- In the lab, a typical embedded Linux system boots thusly...
  - Power-cycle hardware
  - Firmware initializes HW and sets up a `bd_t` structure with info for Linux kernel, and presents command line prompt to user (serial console)
  - User sets up FW variables to point to development host and issues `tftp` command
  - Host downloads embedded kernel
  - Target kernel boots, NFS-mounts root filesystem and presents login prompt on serial console



# Host System : TFTP

- By definition, an embedded Linux system has to boot an embedded Linux kernel
- The kernel can be resident on the board in flash (deployed system) or downloaded each time the board boots (lab system)
- Most COTS development boards come shipped with FW that can accept a TFTP download
- Linux host has to have TFTP enabled and embedded kernel in /tftpboot directory

# Host System : NFS

- The Linux development host should be able to serve up an NFS-mount root filesystem to the target.
- In `/etc/exports` file, very important to include `no_root_squash`
- Multiple active development targets require multiple root filesystems
- `$USER` or other mechanism can be employed, e.g.
  - `/exports/rootfs/$USER`

# Host System : DHCP

- Not absolutely required
  - (see .../Documentation/nfsroot.txt)
- Is very helpful with multiple developers and multiple targets
- Allows IP addresses on development intranet to be associated with target MAC addresses and root filesystem mount points and kernel download files

# Host System : FTP

- Not absolutely required
  - If target kernel is configured to support NFS, a development area on host can be mounted by target, for access to development application
  - Without target kernel NFS support, some mechanism must exist to move application files between host and target, especially if target does not require reboot between test runs
  - FTP is convenient mechanism to support this requirement.

# Demo 1

- On host, show...
  - /exports/rootfs
  - /etc/exports
  - /etc/dhcpd.conf
  - /etc/hosts
  - /tftpboot
- Boot RPX-Lite board (PPC/823) with Monta Vista Linux distribution
- Boot Opto22 LinuxBrain

# Embedded Linux

- **Different flavors of Linux**
- **Host System Issues**
- **Kernel Porting Issues**
- **zImage vs. vmlinux**
- **Debugging Issues**
- **Real-time vs. Non Real-time**
- **Embedded Distributions**

# What kernel tree do I use?

- kernel.org is x86 centric
- Each main embedded architecture has its own tree, with separate maintainers
- At any instant in time kernel.org is out of sync with these architecture trees
  - Sometimes it is transient, a feature in one of the trees has not yet been migrated to the other tree
  - Sometimes it is by design, e.g., not all patches accepted into the PPC tree will be accepted by Linus into the x86 tree

# What kernel should I use?

- This is probably the first question a “do-it-yourselfer” should ask?
- Sometimes there is no choice
  - An x86 project should probably use kernel.org
  - A PPC project may require device drivers or capabilities not in kernel.org
- Is your project on the “bleeding edge”?
- Do I need to use 2.6 or can I use 2.4?
- A good first step is to grab the latest kernel.org tarball and play with ARCH



# Firmware Requirements

- My desktop doesn't require firmware why does my embedded system?
  - The firmware used by your desktop is called BIOS
  - The load function is pointed to by the MBR and is typically either GRUB or LILO
- Anybody starting a do-it-yourself project should square away the FW first
- If the HW is proprietary or the provided FW doesn't support Linux, look for an Open Source FW project, e.g. U-Boot

# Getting Started 101

- When porting the kernel to new HW, don't include the kitchen sink
- Configure a minimalist kernel, which only needs to include networking and serial port support for onboard hardware devices.
- Once the minimal kernel is up and running, start adding things
- Expect to spend coding time in .../arch tree and .../drivers tree
- On PPC many supported boards fill in bd\_t in .../arch/ppc/boot/simple/embed\_config.c

# Embedded Linux

- **Different flavors of Linux**
- **Host System Issues**
- **Kernel Porting Issues**
- **zImage vs. vmlinux**
- **Debugging Issues**
- **Real-time vs. Non Real-time**
- **Embedded Distributions**

# zImage vs. vmlinux

- zImage is a compressed vmlinux with a loader wrapped around it
- When you run 'make zImage' vmlinux is created, as normal, and zImage can be found somewhere in the .../arch tree
- Regardless of debug flags enabled, the compressed kernel image is stripped as is the zImage file
  - A 30 MB kernel file (with symbols) can be less than 1 MB as a zImage file

# zImage vs. vmlinux

- Basically, zImage does...
  - Uncompresses kernel image
  - Relocates kernel image
  - Can provide kernel command-line arguments
  - Starts executing the kernel

# Embedded Linux

- **Different flavors of Linux**
- **Host System Issues**
- **Kernel Porting Issues**
- **zImage vs. vmlinux**
- **Debugging Issues**
- **Real-time vs. Non Real-time**
- **Embedded Distributions**

# Debugging Tools

- Kernel Debugging
  - kgdb (and other kernel patches)
  - Abatron BDI-2000 (and other JTAG emulators)
  - Dynamic modules
- Application
  - gdb
  - ddd

# Kernel Debugging : kgdb

- Kgdb and other kernel patches
- Kernel is configured to include debug handlers, rebuild and use modified kernel
- Communicates through serial port to gdb session running on different computer
  - *No problem in embedded world, this is how it always is*
- Downside
  - Kernel is bigger
  - Communication channel support
  - Can be flaky



# Kernel Debugging : JTAG

- Abatron BDI-2000 and similar tools
- Connects to target through JTAG port
- Connects to development host
  - Serial connection for configuration
  - Network connection for data traffic
- Similar functionality as gdbserver on target for gdb debug session

# And the Survey says...

- Hardware assist tools will be a battleground for traditional embedded tools vendors
- WindRiver is still a player here, through acquisitions, and will find a niche, probably on the high-end
- Other companies will find their niches
- This is an arena where the vendor has to find a way to charge for something other than software, and they will

# David vs. Goliath

- Abatron BDI-2000 JTAG emulator
  - Favored by embedded kernel developers for many years.
  - Cheap, easy to configure, easy to use
  - And it works with Linux (MMU-aware)
- WindRiver Vision/Click and Vision/Probe
  - Windows-only GUI (Vision/Click) \*
  - Automag'ic configuration of target
  - Latest versions are Linux compatible
  - \*One architecture supported by Eclipse/Linux

# One Differentiating Point

- The Abatron BDI-2000 and other JTAG devices of its ilk, require the kernel to be aware of its existence
  - Typically this means a possible configuration parameter and certainly code changes to support (CONFIG\_BDI\_SWITCH)
  - What it really means is that the firmware of the JTAG unit and the kernel have to be in-sync
- The tool (BDI-2000) is made aware of the target environment through an ASCII configuration file

# One Differentiating Point...

- A selling point of the WRS solution is the automatic configuration of the target
  - The kernel is not modified
  - The tool has to be made aware of the target Linux environment through FW variables and well-documented user steps
- The tool is aware of the target environment through the Vision/Probe firmware

# Kernel Debugging and Dynamic Modules

- There are two reasons to use a kernel debugger
  - Bringing the kernel up on a new target board
  - Debugging device drivers
- A hardware assist tool should allow you to debug modules as easily as kernel or driver debug

# Application Debugging

- The only standard is gnu gdb
- GUI front-ends for gdb abound, including DDD
- Remote serial protocol is supported
  - Interface to kgdb on remote target
- gdbserver can perform same function across network interface or serial interface
- Some target environments are rich enough to run gdb locally
  - Target resources used, not host
  - Ddd not typically available

# Demo 2

- Let's take a 10 minute break while I setup to do a hardware debug demonstration
- Show ARCH in ../Makefile
- Show vmlinux vs. zImage
- Show embed\_config.c
- Show kernel debug with JTAG and bdiDDD



# Embedded Linux

- **Different flavors of Linux**
- **Host System Issues**
- **Kernel Porting Issues**
- **zImage vs. vmlinux**
- **Debugging Issues**
- **Real-time vs. Non Real-time**
- **Embedded Distributions**

# Real-time vs. Non Real-time

- Micro-kernel between Linux and hardware
  - RTLinux
  - RTAI
- MVL w/pre-emptible kernel patch
- 2.6 w/pre-emptible kernel patch merged

# Is your system Real-time?

- Dr. R. Callison at Boeing Phantomworks provides a good working definition...
  - In a hard real-time system, the value (utility) of a computation falls to zero at a given deadline
  - In a soft real-time system, that value (utility) declines at some known or unknown rate

# Micro-Kernel

- Software abstraction layer between hardware devices and Linux kernel
- Linux kernel runs as background or lowest priority process/thread/task of micro-kernel
- Linux kernel is patched (e.g. disable sti/cli)
- Interface mechanisms exist between micro-kernel tasks and Linux processes/threads

# Micro-Kernel...

- RTLinux
  - [www.fsmlabs.com](http://www.fsmlabs.com)
  - Commercial product
  - Patent
- RTAI
  - [www.aero.polimi.it/~rtai](http://www.aero.polimi.it/~rtai)
  - Opensource project
- Adeos nano-kernel
  - Now used by RTAI to avoid patent infringement

# And the Survey Says...

- Don't be surprised to see competition here
- There are already plenty of alternatives
- Relatively easy point-of-entry for traditional RTOS vendors
  - Must be able to show differentiation at this level
  - Kernel is relatively easy to extend with modules, which can be binary-only, i.e., locking you into a vendor's solution suite (e.g. TimeSys)

# Pre-emptible Kernel Patch

- Timeline
  - 7.September.2000
    - LinuxDevices.com article announces Monta Vista Software is working on patch, based on 2.4 SMP technology
  - Summer.2001
    - MVS releases HHL2.0 with pre-emptible kernel patch available on x86 targets
  - 8.January.2002
    - LinuxDevices.com interview with pre-emptible kernel patch maintainer, Robert Love

# Pre-emptible Kernel Patch...

- Timeline
  - 10.February.2002
    - LinuxDevices.com article on pre-emptible kernel patch being merged into tree v.2.5.4-pre6, by Linus Torvalds
  - Summer.2002
    - MVS releases MVL2.1 with pre-emptible kernel patch available for many targets
    - Robert Love becomes MVS intern



# Textbook Example of the Open Source Community

- Private company creates proprietary technology based on GPL
- Technology is generally released to Open Source community through patch
- Hacker in the community champions the patch and becomes the “owner”
- Patch “owner” hired by original company

# Linux Kernel 2.4

## Performance Numbers

The following were statistics used by a commercial company  
To promote an older distribution

	Stock 2.4	Preempt 2.4	Micro Kernel
Interrupt Latency	80 – 150 us	80 – 150 us	30 – 50 us
Task Response Latency	20 ms	< 200 us	< 100 us

# Embedded Linux

- **Different flavors of Linux**
- **Host System Issues**
- **Kernel Porting Issues**
- **zImage vs. vmlinux**
- **Debugging Issues**
- **Real-time vs. Non Real-time**
- **Embedded Distributions**

# Commercial Distributions

- From a recent VDC survey of embedded...
  - BlueCat
  - Redhat
  - Embedix
  - Monta Vista
  - Red ICE
  - RT-Linux
  - Timesys Linux
  - Tynux
  - Other

# Non-Commercial Distributions

- Wolfgang Denx ELDK
  - Denx is owner of U-Boot project
  - Denx is VERY active on various mailing lists
  - Embedded Linux Development Kit is what Monta Vista's HHL Journeyman Edition used to be : a FREE “professional quality” embedded Linux distribution
  - You get what you pay for

# Can your embedded distribution do this ?

- Monta Vista Software v2.1 on an embedded x86 platform can configure, build and run the following Open-Source software, without any modification...
  - playcd
  - aumix
  - sox
  - cdda2wav