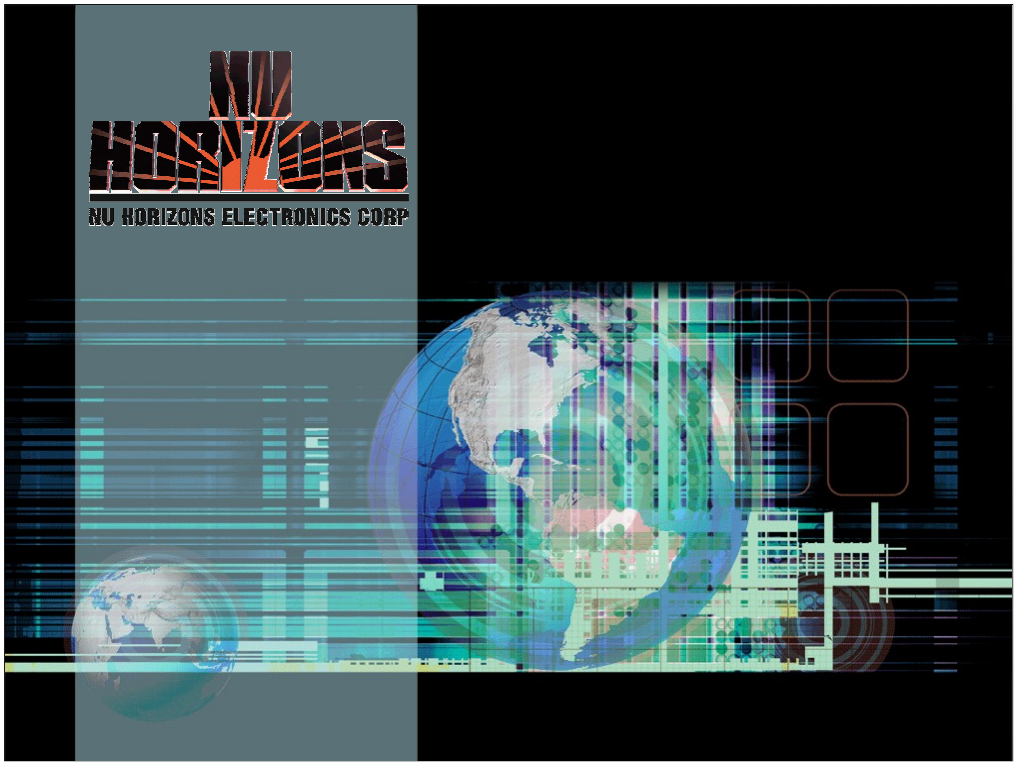


This presentation of “uClinux-on-MicroBlaze” given

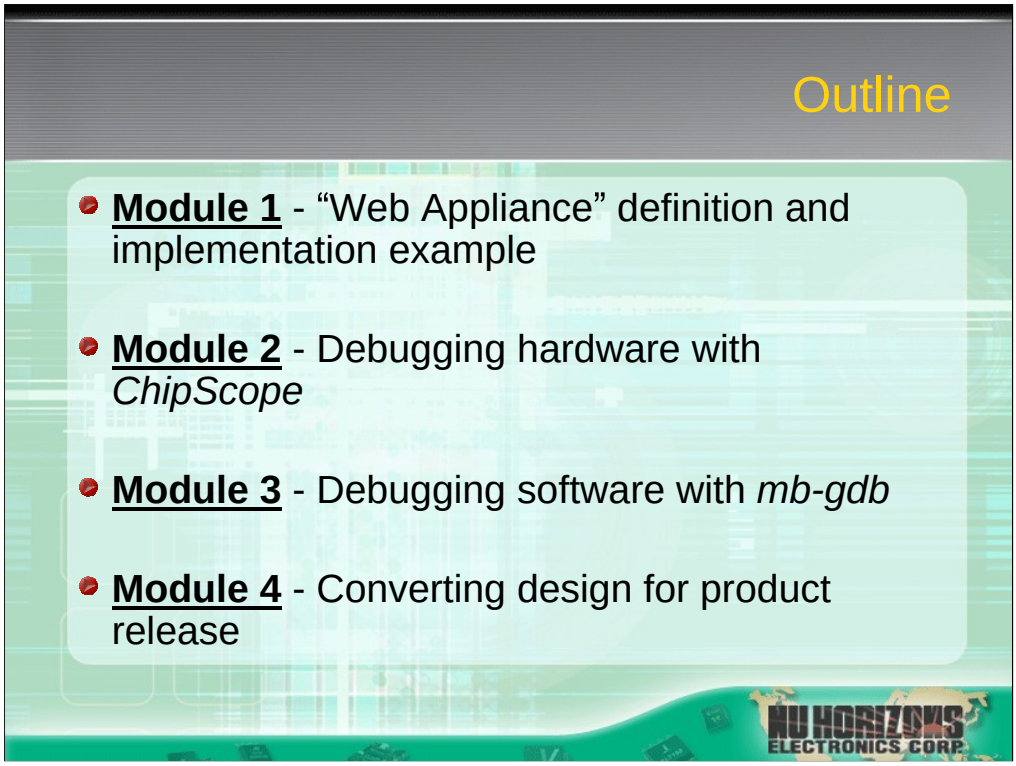
By: David Banas, Xilinx FAE
Nu Horizons Electronics Corp.
2070 Ringwood Ave.
San Jose, CA 95131
<dbanas@nuhorizons.com>

At: Xilinx Learning Center, San Jose

On: Tuesday, May 24th, 2005

μClinux-on-MicroBlaze

Enabling low cost, Spartan3 based Web Appliance design.



The general intended format for this presentation is as follows:

Each module is intended to consume 1 hour, yielding a total time for the presentation of 4 hours.

The intended time format of each module is the same and is as follows:

-45 min.) Material presentation

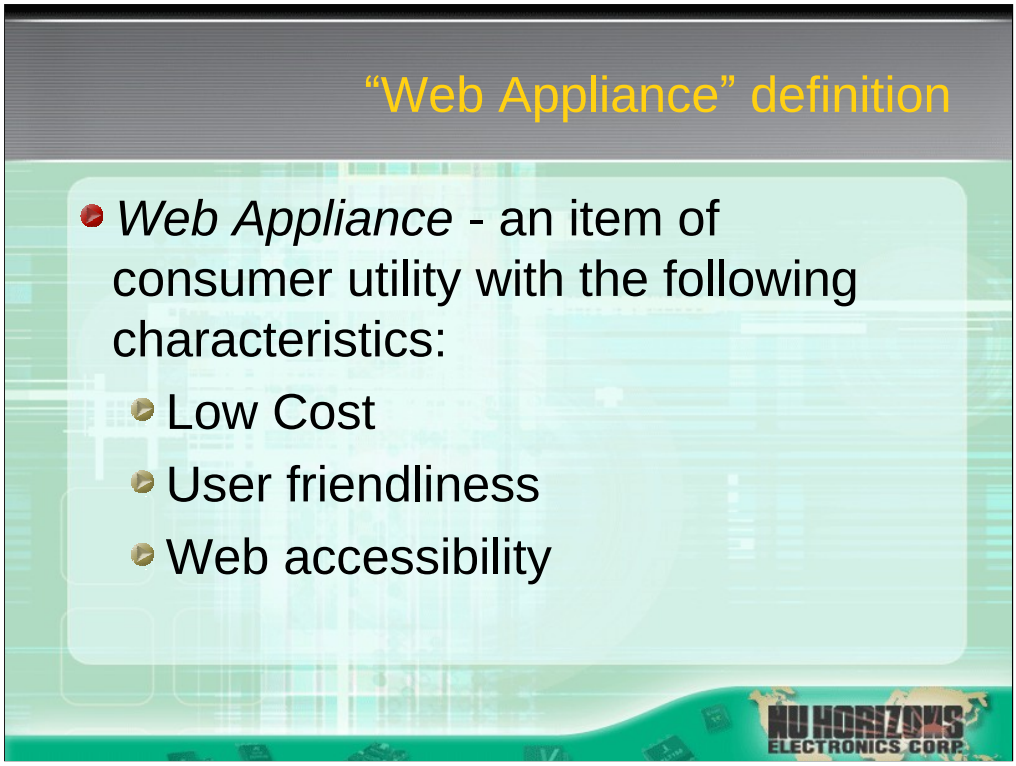
-10 min.) General Q&A

-5 min.) Inter-module break

Module 1 – Title Slide

μClinux-on-MicroBlaze

Module 1 – *"Web Appliance" definition and example implementation*



The target application used as the overall focal point for this presentation is the "Web Appliance", which has the following features and characteristics:

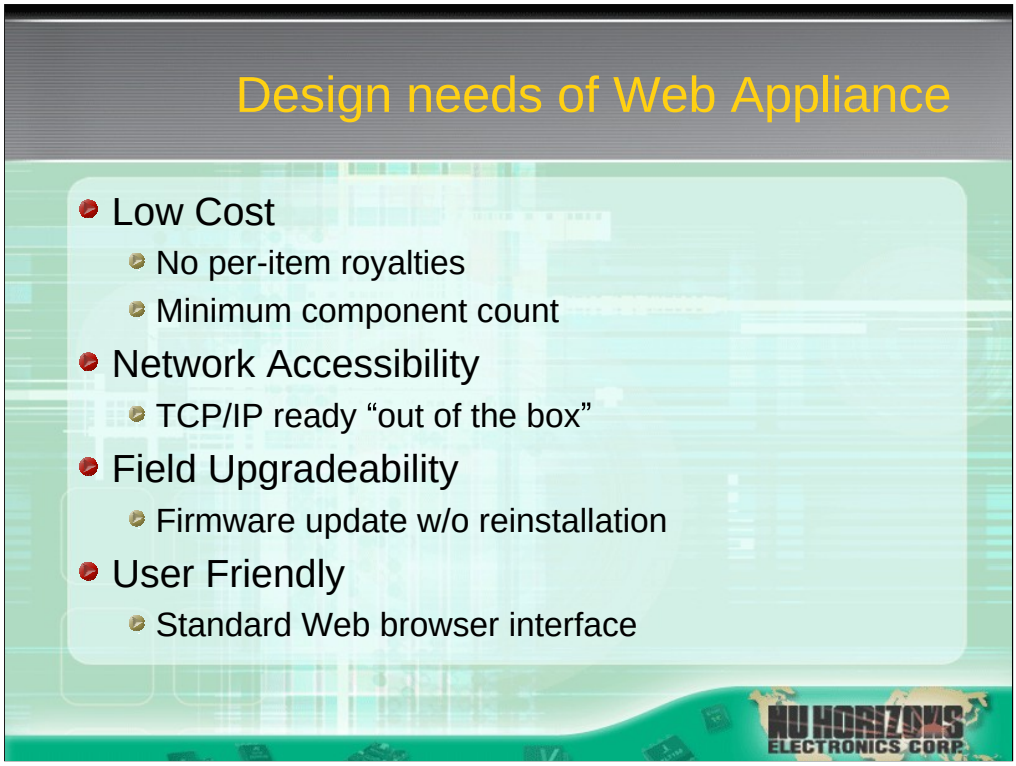
•Low cost consumer device

•Network accessible using ordinary TCP/IP networking protocols over Ethernet, specifically:

> •Web (i.e. – HTTP) server based access and control, allowing the user to interact with the device, perhaps remotely, using a standard World Wide Web browser, such as Internet Explorer or Netscape.

•Field upgradeability in place (i.e. – no removal or de-installation necessary, in order to upgrade embedded software and/or firmware).

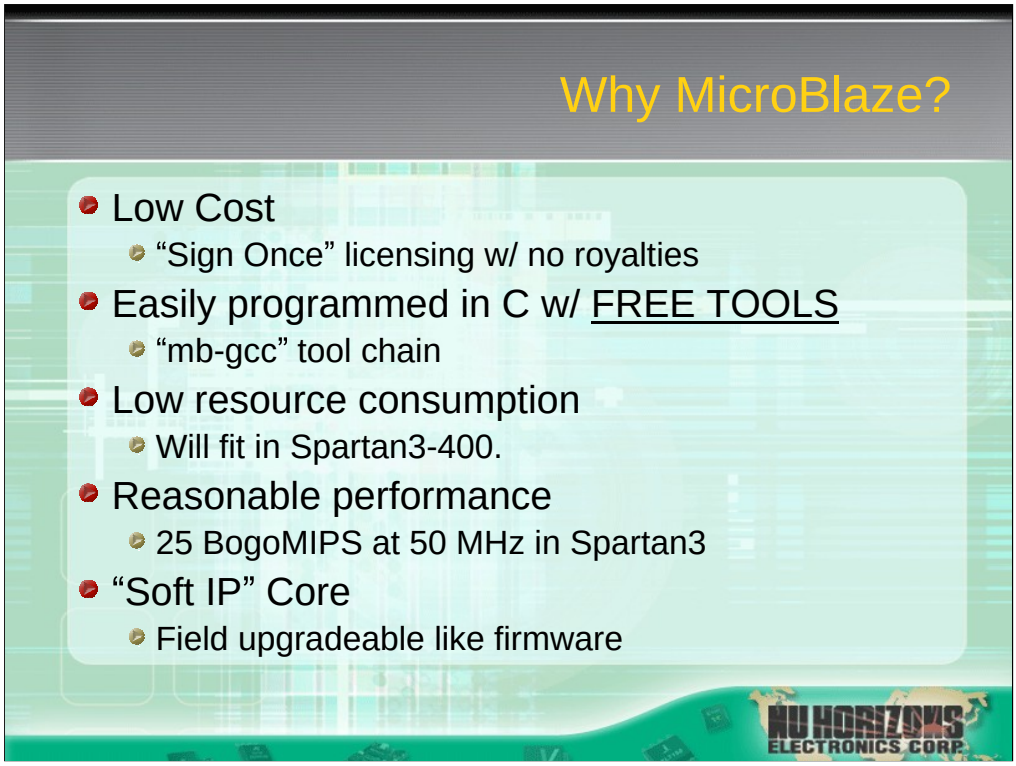
A typical example of such an appliance might be a device, which implements timing and dimming of room lights in a home. The user of this device would be able to control it remotely, from his office at work perhaps, using a standard Web browser, such as Internet Explorer or Netscape.



The device is to be sold into the consumer market space. Therefore, its primary design constraint is low cost.

It is to be accessible via the Web. So, it must have networking capability built in.

It must be “dummy-proof” and, therefore, must NOT require manual user intervention, such as de-installation, when having its software and/or firmware upgraded in the field.



MicroBlaze is a royalty-free processor, which meets the needs of the Web Appliance design requirements, and can be implemented in the least expensive FPGA families, Spartan-3 & 3E.

•Soft IP core, included w/ EDK, is defined in the bit stream containing the FPGA "program".

•32-bit RISC architecture provides approx. 25 BogoMIPS at 50 MHz bus clock frequency in lowest speed grade Spartan-3.

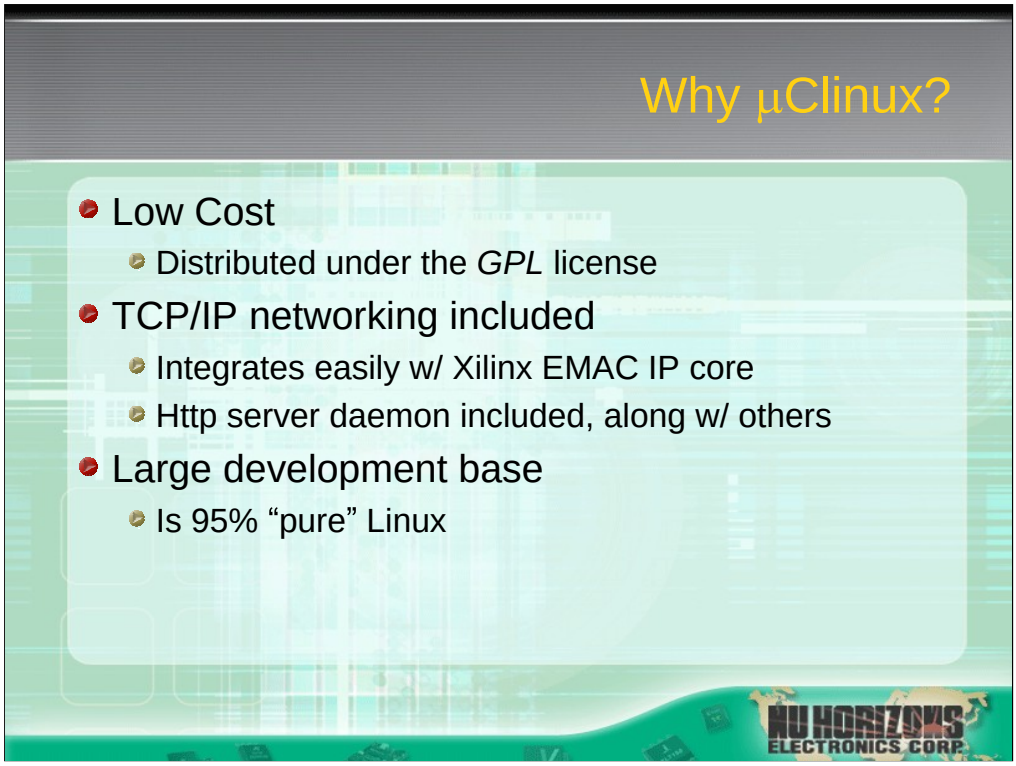
•Will easily support TCP/IP networking at 10 Mbps when run at this speed.

•Boot code is part of the FPGA "program" (i.e. – BRAM init. Contents)

•Kernel and dist. Reside in FLASH, but are run from RAM.

•A custom version of GNU gcc (i.e. – mb-gcc) is available for C compilation of application and boot code.

•Consumes approx. 20% of a Spartan-3 1500.



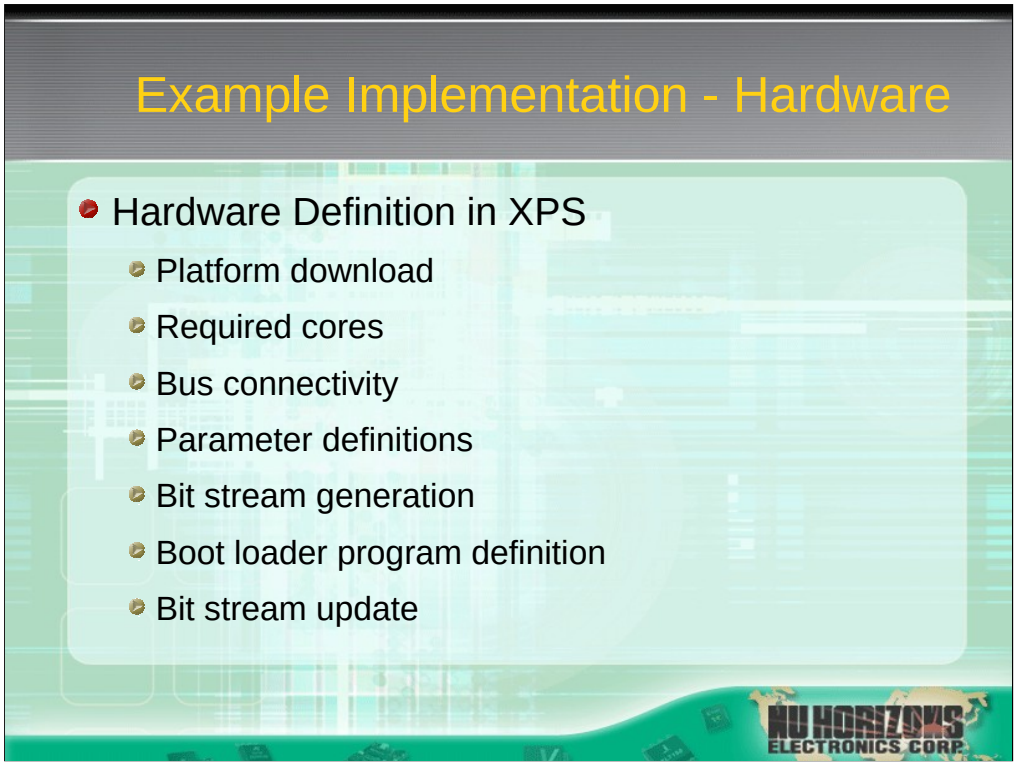
uClinux is a port of the Linux kernel to CPU architectures lacking a Memory Management Unit (MMU), like MicroBlaze.

•It is freely distributed under the GNU General Public License (GPL).

•It includes support for TCP/IP networking, including a HTTP server, by default. (i.e. – No more royalties for embedded network access!)

•Like Linux, from which it is derived, its development base is quite large and active.

•With many eyes all over the World reviewing the code, it tends to achieve a state of robustness rather rapidly.



Some “pre-built” hardware platforms can be downloaded from:

http://www.itee.uq.edu.au/~jwilliams/mblaze-uclinux/Downloads/platforms.html

and can serve as a convenient starting point for your own custom board.

Certain features/cores must be included in the XPS hardware definition, in order to have uClinux-on-MicroBlaze function as a Web appliance.

•Timer with interrupt

•Ethernet MAC core

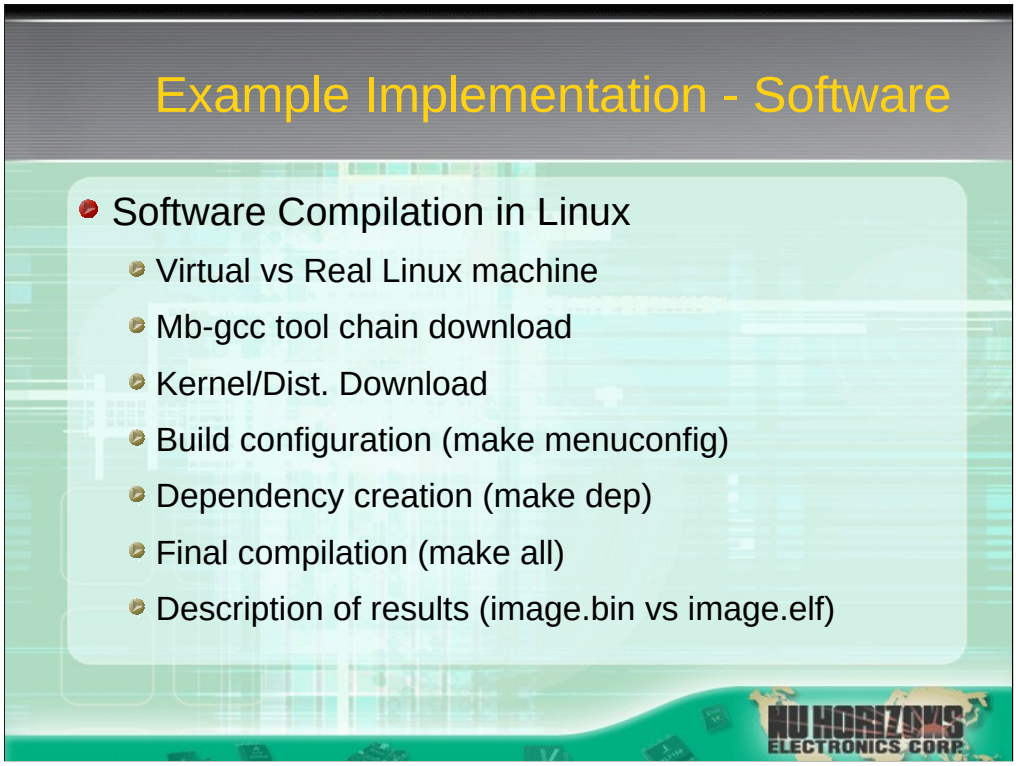
•MDL for debugging and quick download of code

•External memory controller, as uClinux kernel/dist is too large to fit in BRAM alone.

The OPB bus is the main one used for connecting the MicroBlaze to its peripherals. However, there are other busses involved in a complete uClinux-on-MicroBlaze design (i.e. – ILMB, DLMB, & FSL). The best source for seeing how these all must be connected is the list of sample reference platforms given, above.

Certain core parameters must be set correctly, in order to build a successful uClinux-on-MicroBlaze design. Again, the sample hardware platforms listed, above, serve as an excellent starting point for your own custom hardware.

Build libraries, apps and netlist, initialize BRAM contents, and download resultant bit stream to FPGA.



uClinux kernel/dist compilation should be done under Linux, NOT under Cygwin. This may be done in either of 2 ways:

•Using a separate machine running Linux, or

•Running Linux inside a virtual machine on the Windows machine, using a product such as VMware Workstation.

A special version of the GNU development tool suite, made specially for the MicroBlaze processor, MUST be used to compile the uClinux kernel and its associated distribution for running on MicroBlaze. These special versions are easily identified by their “mb-” prefix and can be downloaded here:

http://www.itee.uq.edu.au/~jwilliams/mblaze-uclinux/Toolchain/index.html

The source code for the actual uClinux kernel and its associated distribution can be downloaded here:

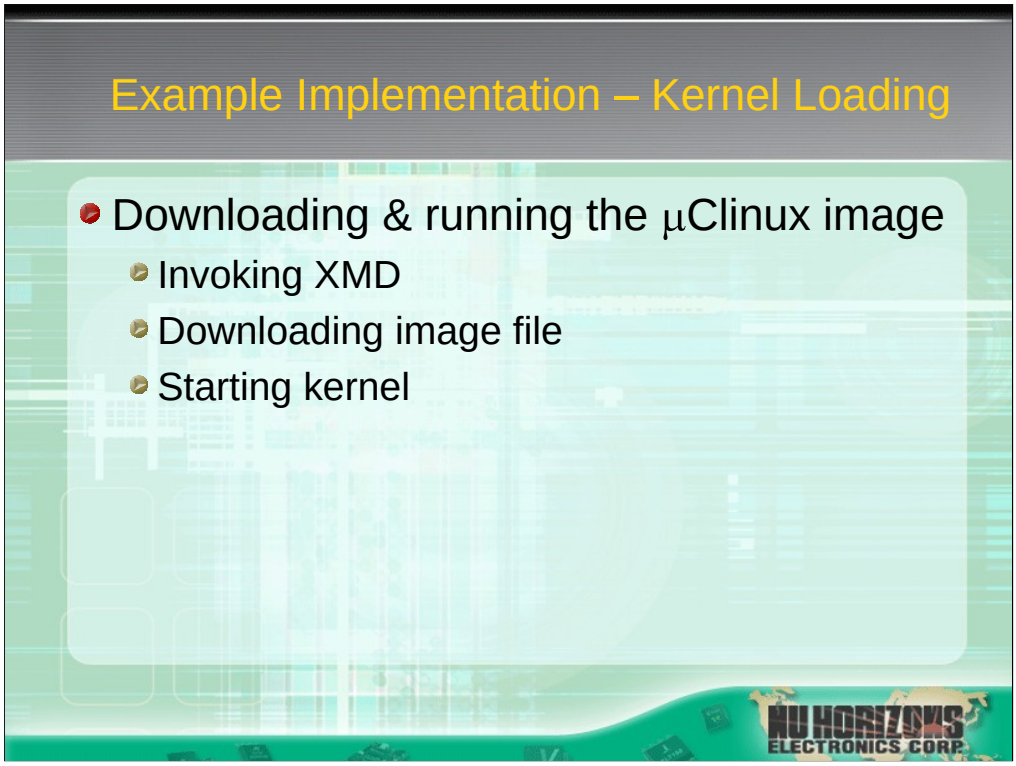
http://cvs.uclinux.org/

General build instructions:

•Use the new “auto-config” flow, in order to automate the transfer of information from XPS to the software build environment.

•Necessary make targets: clean, menuconfig, dep, all

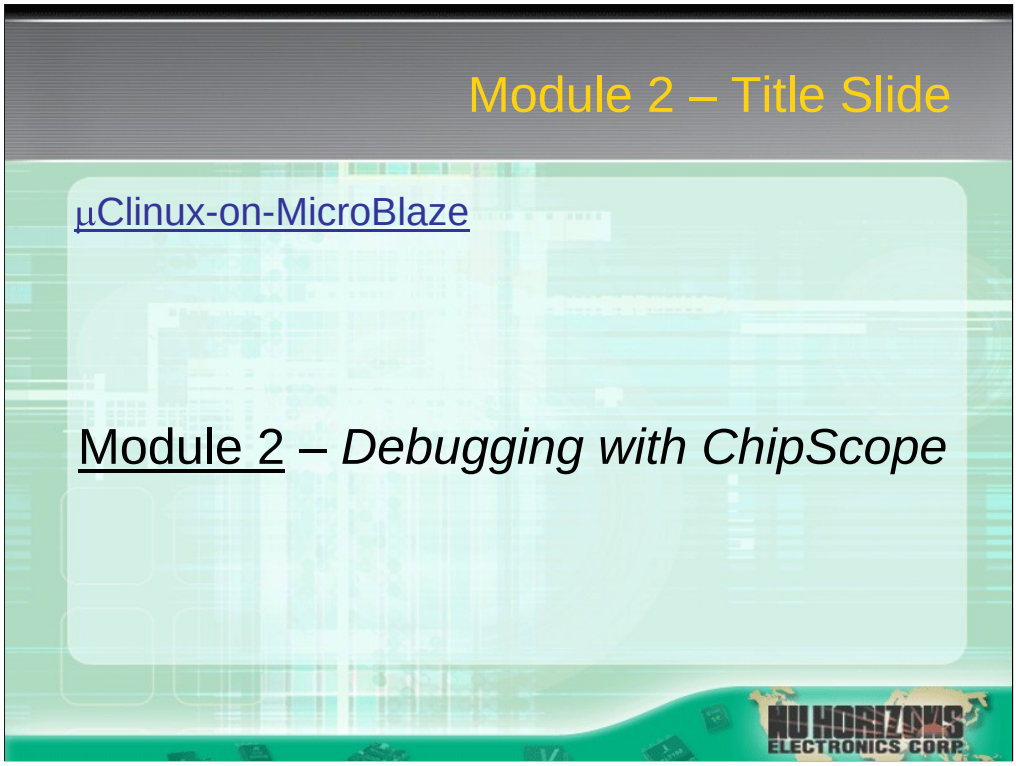
- •Select kernel/user config. change, but don’t change anything, in order to “bring in” auto-config” info.
- •Use “image.bin”, NOT “image.elf”, due to problem with XMD’s interpretation of *.ELF format (i.e. – no romfs support).

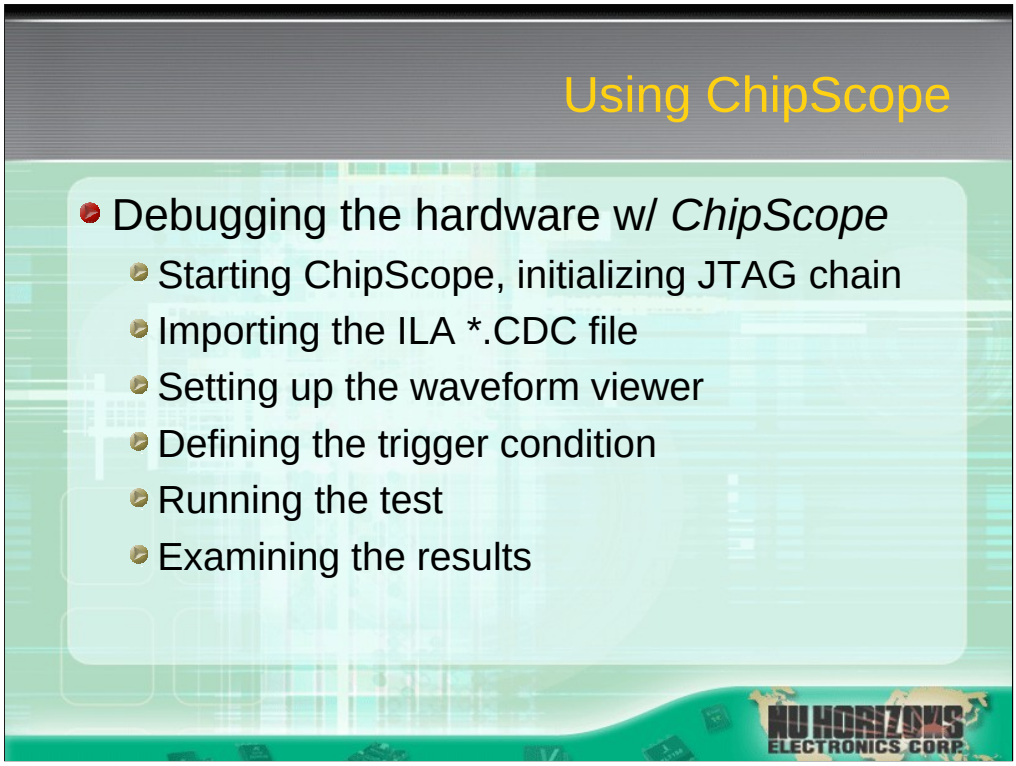


After copying “image.bin” back to the Windows/XPS directory, download it using XMD:

- dow –data image.bin <start>
- con <start> (when NOT intending to debug kernel boot with mb-gdb)

(Presentation of “real-world” uClinux-on-MicroBlaze new board bring up using the Nu Horizons Spartan-3 1500/2000 board as example.)


Module 2 – Title Slide
μClinux-on-MicroBlaze
Module 2 – Debugging with ChipScope
NU HORIZONS
ELECTRONICS CORP.



The *.CDC file gives useful signal/bus name definitions, which can be imported into ChipScope Analyzer, in order to make debugging more convenient. As an example, the *.CDC file for the project being used in this presentation can be found here:

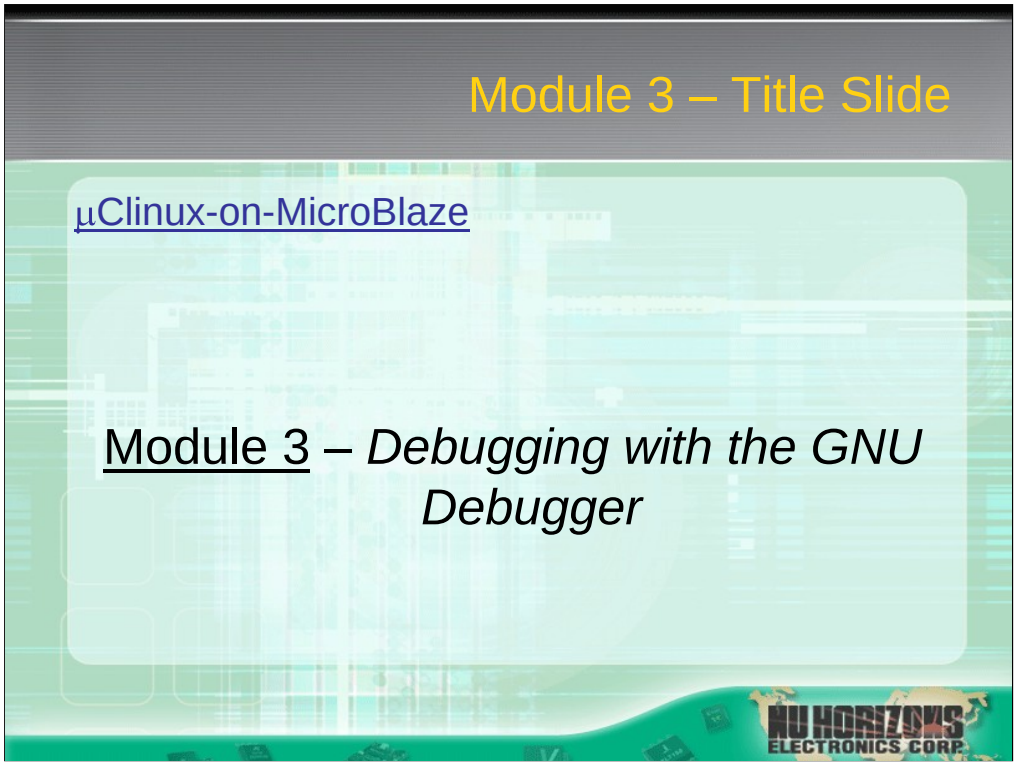
"…\implementation\chipscope_opb_iba_0_wrapper\chipscope_opb_iba_0.cdc"

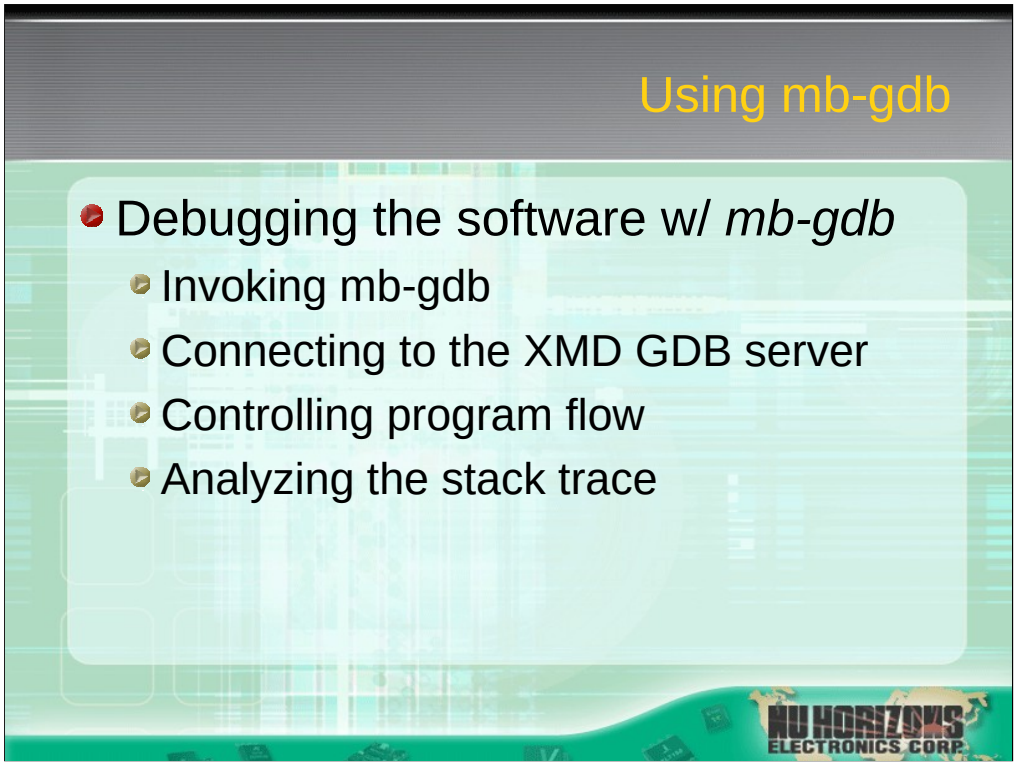
Add signals/busses to the Waveform viewer by selecting and right-clicking on them.

Define the trigger condition as follows:

•Define the necessary match units.

•Define the logical combination of match units that constitutes a trigger event.


Module 3 – Title Slide
μClinux-on-MicroBlaze
Module 3 – Debugging with the GNU Debugger
NU HORIZONS
ELECTRONICS CORP.

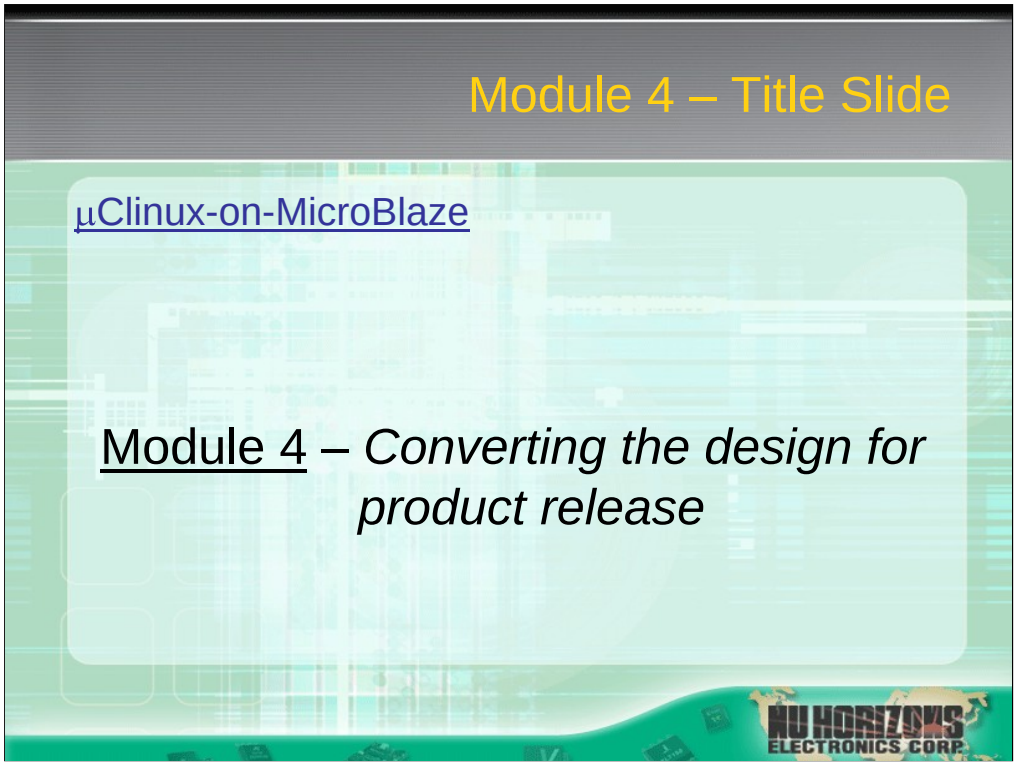


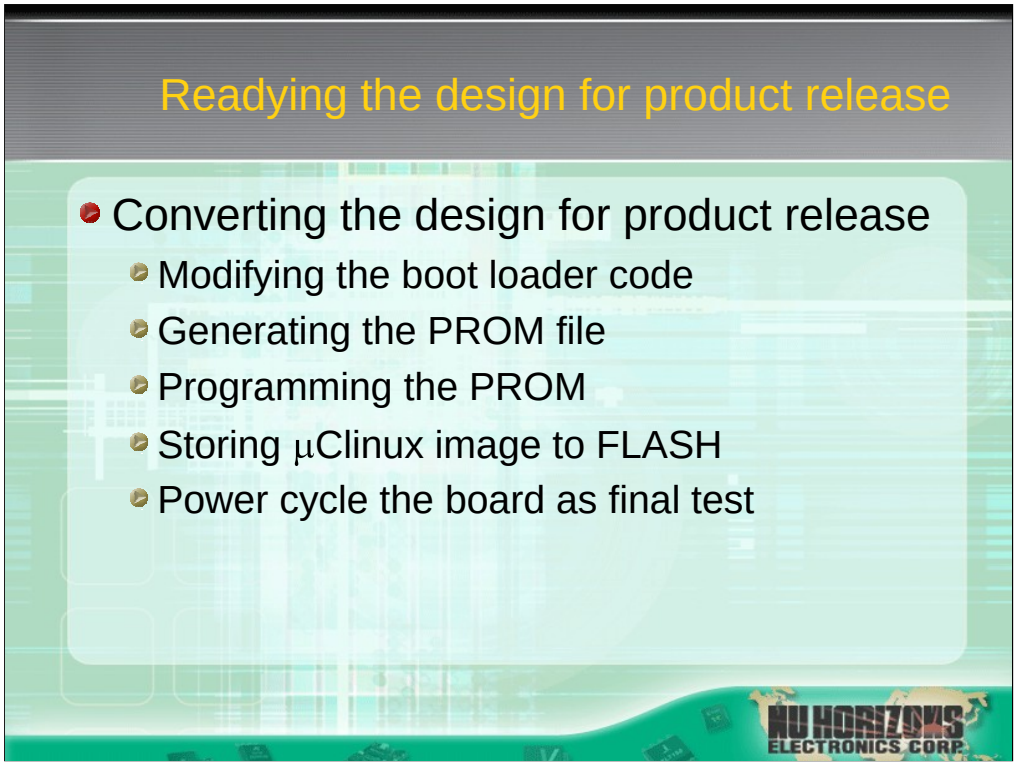
mb-gdb can be used to remotely debug kernel booting.

•After “dow –data…” command, instead of issuing “con <start>” command, return to Linux environment and:

- •mb-gdb image.elf
- •target remote <Windows IP addr>:1234
- •(Begin using mb-gdb as usual.)

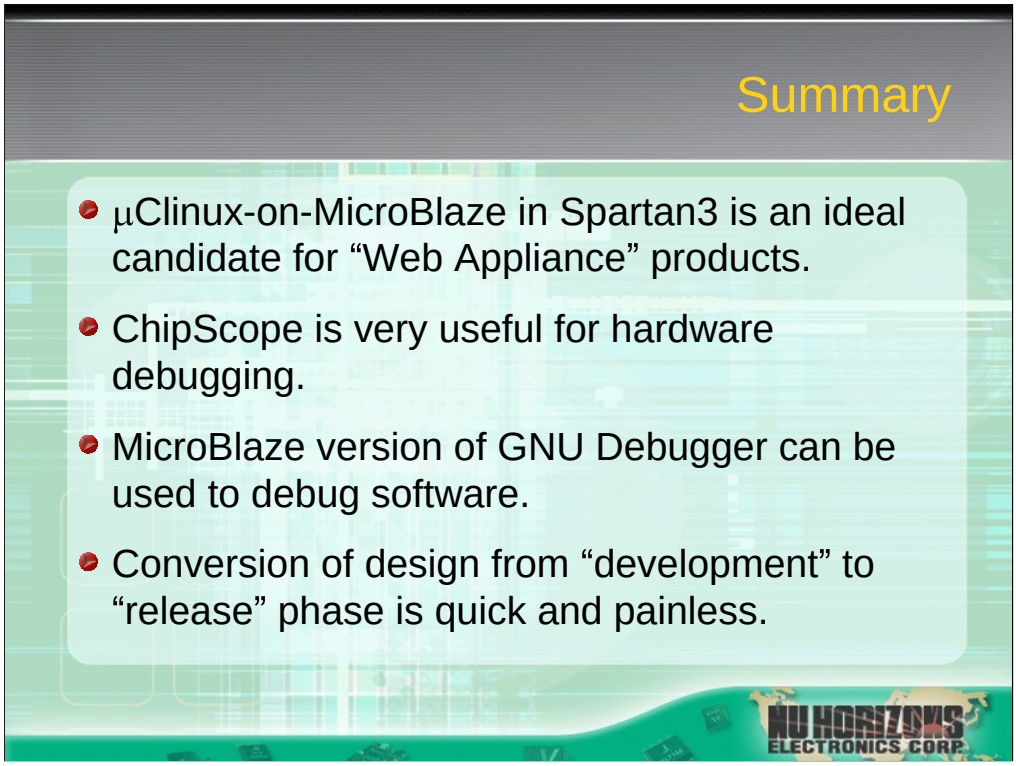
(Single stepping, continuing, and stack trace analysis are demonstrated here.)


Module 4 – Title Slide
μClinux-on-MicroBlaze
Module 4 – Converting the design for product release
NU HORIZONS ELECTRONICS CORP.



Modify the boot loader code to “auto-chose” option #5, which loads the uClinux image from FLASH into RAM and transfers control to the RAM version.

(See Xilinx documentation for remaining steps.)



List of helpful resources:

•http://www.itee.uq.edu.au/~jwilliams/mblaze-uclinux/ - uClinux-on-Microblaze home page, maintained by John Williams of the University of Queensland, Australia.

•http://www.ucdot.org/ - uClinux general (i.e. – not MicroBlaze specific) home page

•http://uclinux.openchip.org/CMS/ - uClinux section of the “OpenChip” public cores development site

•http://www.linuxjournal.com/article/7221 – article in Linux Journal, which introduces Linux programmers to uClinux

•http://www.cyberguard.info/snapgear/tb20020917.html – article on using FLASH memory with uClinux

Acknowledgements

- μClinux-on-MicroBlaze Development Team
  - John Williams – University of Queensland, Australia
  - Greg Miller – Avnet Xilinx FAE, Denver
  - Steve Sanders – μClinux-on-MicroBlaze pioneer
- Xilinx Software Development Team
  - Mb-gnu tool suite
  - Platform Generator modifications (i.e. - auto-config)
- Nu Horizons Management
- Xilinx XDTV Studio Crew !!!!!!!!!!!!
  - Jennifer Sunglao
  - Raul Rocha