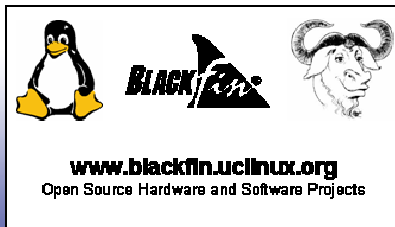The World Leader in High Performance Signal Processing Solutions

uClinux

ANALOG
DEVICES

What's so special about Linux?

- ◆ **Multiple choices vs. sole source**
- ◆ **Source code freely available**
- ◆ **Robust and reliable**
- ◆ **Modular, configurable, scalable**
- ◆ **Superb support for networking and Internet**
- ◆ **No runtime licenses**
- ◆ **Large pool of skilled developers**

What is uClinux?

- ◆ **Open Source means no purchase cost and royalty free**
- ◆ **Huge base of (free) developer support**
- ◆ **Full component driver support for serial, Ethernet, host USB, etc.**
- ◆ **Single click selection will configure necessary Linux kernel, libraries, and applications**
- ◆ **Cross-compilation tools**
- ◆ **Available now for BF-531/2/3, BF-535, BF-537**

Differences between Linux and uClinux

◆ **Memory Management**

- **The most significant difference is the lack of a MMU in uClinux**

- **uClinux removes the MMU requirement by implementing a flat memory model (a 1-to-1 virtual-to-physical mapping)**

- **On uClinux, memory is limited to whatever physical memory is available at a given time to a program**

What is uClinux? (cont′)

- ◆ **Linux vs. uClinux**
  - ● **Most of binaries and source code for the kernel have been rewritten to tighten-up and slim-down the code base**
  - ● **The uClinux is much, much smaller than the original Linux 2.0 kernel, while retaining the main advantages of the Linux OS: stability, superior network capability(a full TCP/IP stack), and excellent file system support (Only NFS, ext2, MS-DOS and FAT16/32)**
    - ◆ common Linux API
    - ◆ uCKernel < 512 KB
    - ◆ uCKernel + tools < 900 KB

What is Real Time?

"A real time system is one in which the correctness of the computations not only depends upon the logical correctness of the computation but also upon the time at which the result is produced. If the timing constraints of the system are not met, system failure is said to have occurred."

- Donald Gillies

"Real time in operating systems:
The ability of the operating system to provide a required level of service in a bounded response time."

- POSIX Standard 1003.1

Hard vs. Soft Real Time

- ◆ **Hard**
  - ● *guaranteed worst-case* response times
  - ● absolutely, positively, first time every time
- ◆ **Soft**
  - ● Kinda, sorta, usually

Linux and Real-Time

♦ **Is Linux 2.4 Hard Real-Time?**

- **In a nut shell : NO**, BUT...Why**?**

- **(Monolithic Kernel) The Linux kernel uses coarse grained synchronization, which allows a kernel task exclusive access to some data for long periods. This could delay the execution of any real-time task that needs access to that same data.**

- **(Non Preemptable) The Linux kernel does not preempt the execution of any task during system calls. If a low-priority process is in the middle of a system call and a message is received for a real-time process, the message will unfortunately be held in the queue until the system call completes, despite its low priority.**

Linux and Real-Time

◆ **(Resource Lock) Linux makes high-priority tasks wait for low-priority tasks to release resources. For example, if any process allocates the last network buffer and a higher priority process needs a network buffer to send a message, the higher priority process must wait until some other process releases a network buffer before it can send its message.**

◆ **(Priority Scheduling) The Linux scheduling algorithm will sometimes give the most unimportant and "nicest" process a time slice, even in circumstances when a higher priority process is ready to execute.**

Real-Time Work on Linux

- **Today Linux can provide Soft Real-Time out of the box.**
- **Two Approaches to Real-Time Linux**
  - **Modify Linux to include a Real-Time scheduler.**
    - Fixes the problems discussed previously.
  - **Put Regular Linux on top of a Real-Time operating system (RTOS).**

Real-Time Work on Linux

- **Several *Vendors* are working on Real-Time solutions:**
  - MontaVista has a Real-Time kernel that they are trying to get accepted by the Linux Community.
  - RTLinux provides a Real-Time kernel uses Linux a thread of the Real-Time OS.
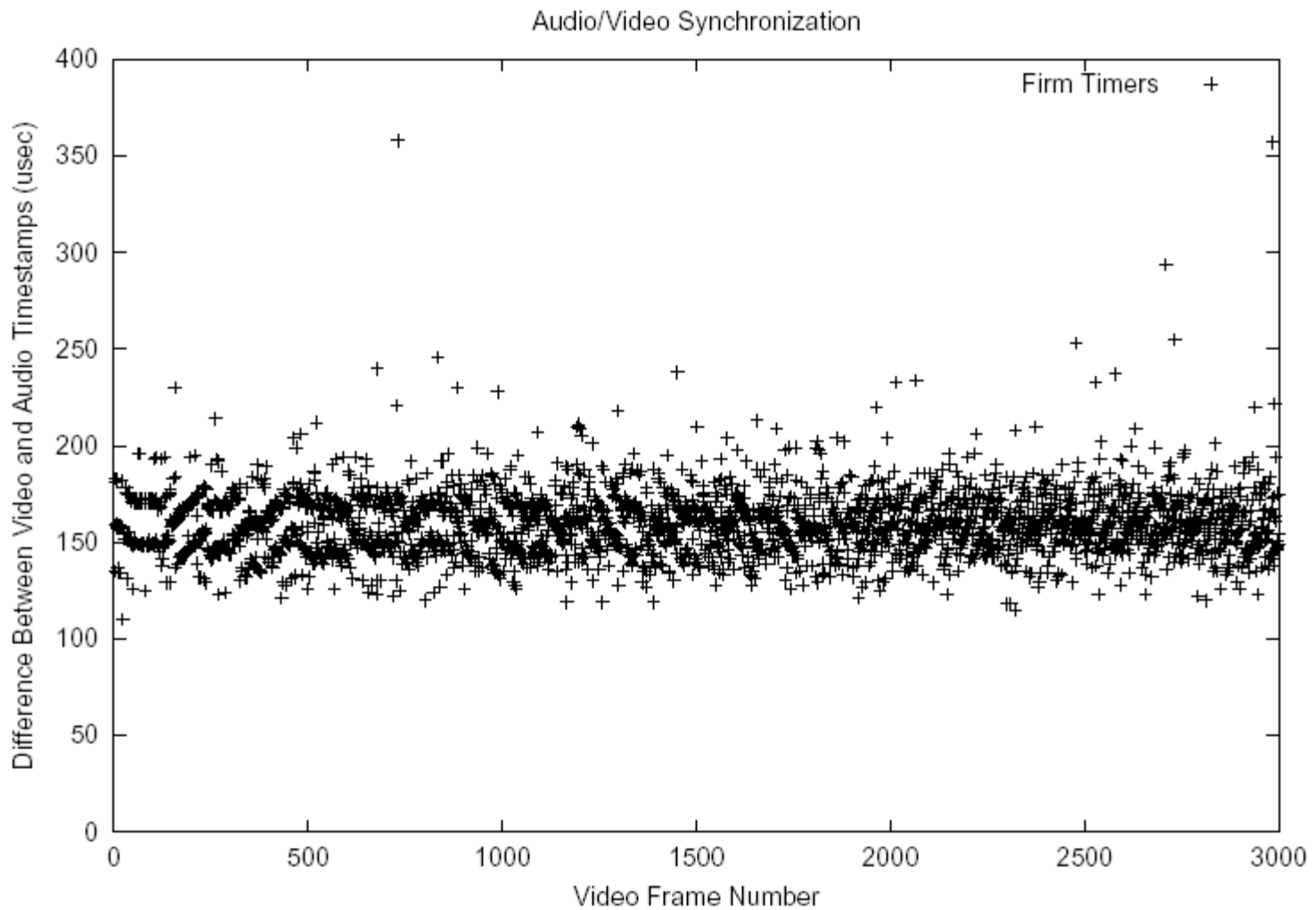 - Lineo has implemented the Real Time Application Interface (RTAI) for Linux.

- **Several Open Source Development teams are working on Real-Time solutions:**
  - Fixes have gone into the 2.6 kernel

2.4 kernel real time performance
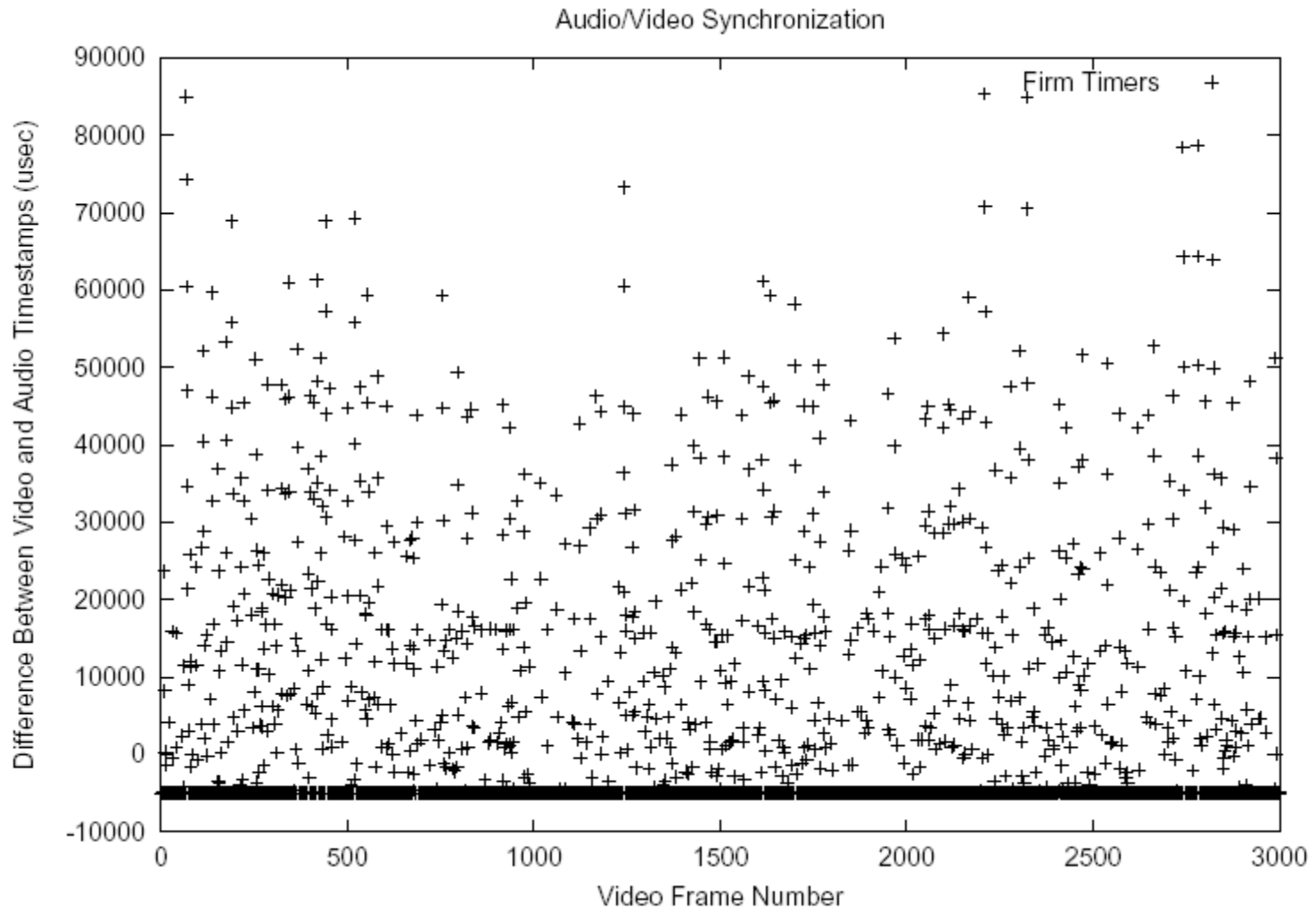
2.6 kernel real time performance

uClinux Development

- – Cross Development Tools
- – Libraries
- – Debug Concept / JTAG Emulators / Debugger
- – Initialize Debgger / Intialization Script

uClinux Development (cont')

Cross Development Tools

- **http://www.blackfin.uclinux.org/projects/gcc3**
    - binutils
    - gcc
    - gdb
    - elf2flt

- **http://www.blackfin.uclinux.org/projects/uclinux533**
    - kernel

uClinux Development (cont')

Libraries

◆ **uClibc**

- **The latest version: uClibc 0.9.15**
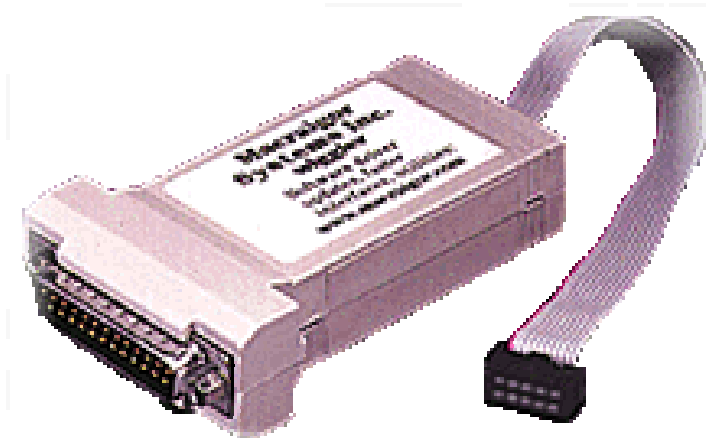- **download: http://blackfin.uclinux.org/projects/uclinux533/**

uClinux Development (cont')

JTAG FlashProgrammers (e.g. IGLOO)

(Or else - Make your own wriggler – Schematics freely available)

JTAG tools for debugging

Wigglers