ECEN 5653/4653
Real-Time Digital Media
(A Linux-Based Systems Approach)

Lecture 1 - Introduction
Prof. Sam Siewert - My Background

- Co-Founder of Embedded Certificate Program with Dr. McClure
- First Taught Linux Kernel Lab in 1996 at CU
- First Taught ECEN 5623/4623 in Spring 2000

Education
- Physics/Philosophy Student … U. of California, Berkeley
- BS Aerospace Engineering U. of Notre Dame
- BS Work in Electrical and Computer Engineering U. of Houston
- MS, PhD Computer Science U. of Colorado

Experience (~20 Years in Embedded)
- Presently with CU and Trellis-Logic LLC (Founder)
- Intel Architecture Group (Atom, Scalable Cloud Solutions)
- CTO at Atrato Inc., a Digital Media Storage Start-up in Broomfield
- Consulting with Numerous Digital Media Firms
- 12 Years NASA JSC, NASA JPL / CU, NASA JPL / Ball Aerospace
- 12 Years Commercial Telecomm, Storage/Networks, Embedded, Digital Video

Experienced in Delivery of Hard and Soft Real-time Systems, Embedded Small Scale and Large Scale
- Numerous Hard Real-Time RTOS/FPGA Systems (Shuttle, Satellites)
- One Small Scale Linux 2.4.x Optical Switch
- Several Large Scale Linux 2.6.x Digital Video On Demand Systems
Scalable Linux Systems…

- Cell Processor: from Play-Station-3 to HPC
- GPU: from PC Gaming to Top-500

- From ARM Wristwatch to GIS and Video Services

- Huge Value in Open Source Drivers, Tools, and Applications – Speeds Up Time to Market

- But, How can we accommodate Real-Time (Embedded) services and Digital Video/Audio on Linux?


© Sam Siewert
Many Real-Time Systems

- Real Time – Must Respond to Requests for Service by a Deadline relative to request
  - Failure to Respond Prior to Deadline Results in a System Failure
  - Request Rate for Service Driven by Real-World Events
  - Controls Processes and Delivers Deadline Driven Services

- Many Hard RT (Embedded) Apps: Anti-Lock Braking, Aircraft Flight Control, Robotic Systems

- Many Soft RT Apps
  - Streaming Media (On-Demand), Interactive Systems (Games)
  - Post Production
  - Digital Broadcast

Hard Real-Time

Soft Real-Time
Real-Time Digital Media Systems

**Embedded Media Devices**
- Set-Top Boxes (Linux, VxWorks)
- Mobile Media Systems: Smart Phones, Tablet Computing, (Kindle) Readers, Notebooks, DVD Players, iPODs, etc.
- Digital Camera Systems (SD, HD, HD-SDI, 2K, 4K, 6K)
  - Resolutions/Formats -
  - Consumer/Pro-sumer Digital Media
- Game Consoles: X-box, PS3, Wii, Nintendo
- Gesture Recognition, Augmented Reality
- SD, HD Cameras and Interfaces: Composite, S-Video, Component, DVI, HDMI
- More to Come?, No Doubt

**Scalable Digital Media Server Systems (Head End)**
- Post Production for Digital Cinema, TV, Web
  - 2K, 4K, 6K Streams from Digital Cameras
  - Frame/Color Editing, CGI (Computer Generated Imagery), Soundtrack, Write to Distribution Media
- Digital Cable Head-Ends: Server 10K+ Customers, Broadcast, On-Demand, Guide Data, DOCSIS Internet, VoIP
- IPTV Head-Ends: Internet, Switched-Digital Video, On-Demand
- CDN: Content Distribution/Delivery Networks with Streaming
- Viral Video and Social Networking Video/Audio Streaming
- Digital Cinema: HD Digital Projectors, 3D Digital Projectors
- Closed Circuit Security Systems: Multi-Camera NTSC/HD
Digital VoD System End-to-End

- VoD Servers
- DSL or FTTH
- Video Vault
- Billing System
- IPTV
- Edge QAMs
- HFC Cable Network
- STBs

Pitcher
Catcher

© Sam Siewert
National On-Demand System with DRM (Digital Rights Management)
Digital Media Focus

Scalable Post Production Pipelines
- Multi-core Threading with POSIX Threads (CPU scaling)
- I/O Scheduling and Tuning (Disk bottlenecks and Capacity Scale)
  - RAID10, 50, 60, Beyond
  - Deadline Driven
- NVM Architectures for Cache (PCIe Flash, NVM Express, SSD)

Real-time HD Frame Transformations
- Color Transformations, Edge Enhancement
- Integrated Graphics & Video (Interactive TV)
- Green Screen Replacement, Video Morphing
- Segmentation, Recognition, Tagging
- Post-Capture Focusing (Demonstrated at CES)
From Linux … RT Embedded Linux

- Inspired by Linus Torvalds and Minix Pedagogical Unix
  - Linus has now written 2% or less of Linux (But started it all!)
  - Highly leveraged GNU open source tools (Richard Stallman)
  - Point was to create open source Unix (hobby, full source OS)
  - Interest in embedded real-time applications arose in mid 1990s
  - Principally used in Best Effort, Near RT, or Soft RT Apps
  - Linux and Solaris have Dominated Digital Media/Web Services

- Linux as a Best Effort Service in an RT Executive
  - RT Services preempt the Linux kernel itself
  - Linux runs in slack time providing convenient user interaction

- Linux as a Mixed Soft Real-Time / Best Effort Service Environment Using NPTL
  - Popular approach supported by Red Hat, WRS, etc.
  - Concept of Carrier Grade Linux
  - Many recent kernel extensions in 2.6.18 and Later to Support
What’s Controversial, What Isn’t

Linux Hard Real-Time remains to Be Seen…
- HRT requires deterministic response from services (SW or HW)
- HRT SW services are tricky on modern microprocessors
  - Topic of ECEN 5623/4623
  - Consider Linux with FPGA/ASIC state-machines for HRT
  - Consider Linux with Offload (Subsumption Architectures)
  - Consider an RTOS with a few simple SW services
  - Linux can be a BE service in an HRT executive
- HRT Linux SW Services Risky
  - Many kernel features are not deterministic
  - Kernel is still not fully pre-emptable

Linux Soft Real-Time is Well Proven
- Android Clients, Scalable Linux Head-Ends
- Continues tradition of Soft RT-Unix (e.g. LynxOS, Solaris)
- Very Few systems really are HRT
- HRT systems have few HRT and many BE and Soft services
What We Will Cover

- Linux Multi-Core Scheduling and Digital Media Processing, including Vector Processing (SSE, CUDA, Many-Core)
- Digital Video and Audio Encoding and Decoding – MPEG2/4
- Digital Media Server File systems and storage
- Digital Media Interfaces: Composite, S-Video, Component, DVI, HDMI
- Digital Media Transport
  - Broadband (modulated): OTA NTSC (Historical), ATSC, Digital Cable QAM
  - Baseband (Networked): IPTV, Viral Video
- Digital Cinema
- Emerging Topics: Infotainment Systems Design, Smart Phones/Tablet/Readers, Augmented Reality, Digital Cinema, 3D Cinema/TV, IPTV, Storage and CPU Virtualization
What We’ll Do

- Learn and appreciate HRT, SRT, Best Effort Service definitions
  - Emphasis on Soft Real-time
  - Dynamic Scheduling and Load Balancing, Scaling
- Learn Effective Methods to Code Soft RT Services using NPTL
- Use Latest Linux Kernel – Fedora Core 14 (at least FC12, 2.6.31)
- Leverage Digital Video Open Source
  - MythTV
  - Tools for Encode/Decode: ffmpeg, VLC
  - Analysis Tools: TSReader

- Complete Linux-based Digital Video Labs
  - Intro to NPTL and POSIX mechanisms in Linux
  - Intro to MPEG Transport and Program Streams
  - Digital video encode/decode in SW and with HW acceleration
  - Evaluate Linux for Soft RT and Understand HRT/SRT Partitions

- Complete an Extended Lab (Project)
  - Digital Video (RT streaming and stream analysis)
  - Trick-Play
  - On-Demand Server
Using MythTV for OTA Content

- Optionally Purchase WinTV-HVR-1600 and ATSC antenna
- Optionally Purchase CUDA supported GeForce GPU
- Install Fedora Core 14 (12 or better) and Cards
- Install, Test, and Use MythTV to Capture Off-the-Air HD Content (ATSC)
- Explore PVR Features and CX18 Driver
- Ingest Camera Data (Possible)
- Alternatively Use Open Content
How We’ll Do It

1/3 Theory – Lectures/Reading (On-going)
- Lectures
  - What is HRT? (HRT details covered in ECEN 5623), Why is Digital Media not HRT
  - SRT, QoS, Analysis Methods, Scheduling, Resources
  - Digital Video encoding/decoding, transcoding, bit-rates, timing
  - Digital Media Scalable RAID Storage/IO Systems
- Read lots of papers on topic!!

1/3 Practice – 4 Universal Labs (8+ Weeks)
- Linux Coding (C and POSIX)
- MPEG2/4 DV Stream Analysis (C and Open Source Tools)
- MythTV, ffmpeg, custom MPEG parsers, trick-play, etc.
- Digital video with SW encode/decode or trick play operations

1/3 In Depth Study of Digital Media Applications (6 Weeks)
- Digital DSP/Image Processing Design
- Optimization Using Vector Processing and/or Threading
- Produce Original Video/Audio DVD
Linux Platform Options

Roll Your Own – Ubuntu or Fedora Distributions

Using Amazon “Free” Micro-Instances
  – Requires a Credit Card Account with Amazon
  – Can Do Spot-Instance Bids for Large Nodes and Clusters
  – Can Do Spot-Instance Bids for GPGPU Nodes

Using CU ECES Provided GPGPU Nodes

Using Virtual-Box Ubuntu or Fedora Installation
Why?

Most RT (Embedded) Systems are BE, SRT or Hybrid HRT+SRT+BE
- Digital Media has Hybrid HRT+SRT requirements and RT interaction with Users
- You’ll understand when Linux makes sense in RT embedded systems and how to integrate it
- You’ll learn current Linux advantages/disadvantages to RTOS

Digital video rapidly growing industry
- Post Production, Animation, Broadcast
- Interactive Systems (Games, Augmented Reality)
- On-Demand Media (Web, IPTV, Cable VoD)

Fundamentals of Video/Audio Encode and Decode
- How to Deliver (Transport) Digital Media
- How to Scale Systems to 100,000’s of Subscribers
- It will be fun!
- You Too Could Work at DreamWorks, ILM
- Local Industry in Denver: Cable-Labs, TWC, Liberty Media, Comcast, …
Administrivia

Introductions
- TA Introduction (Office Hours)
- Students (Introductions)
- ECE Dept. Front Desk - ecefd@schof.colorado.edu

CULearn
- http://culearn.colorado.edu

Course Information
- http://eceee.colorado.edu/~ecen5653/
- E-mail list (please sign up on sheet being passed around)
- Lecture Notes and Labs on the Web Page (please do not read more than one week ahead)
- RT Linux Terminology – QUIZ IN 2 WEEKS

Home Lab – You MUST Supply Home Linux System and Buy Cards
- Need eces.colorado.edu System Account - ecehelp@schof.colorado.edu
- http://plus.colorado.edu – Account creation/activation
- Fedora Core 12, 2.6.31
- MythTV, ffmpeg, Development Tools
- Laptop or Desktop Linux System OK for Code Development
More Administrivia

Text Books (None Required)
- No Single Text (yet 😊)
- Readings noted on Syllabus
- Can Re-use RTECS from 5623/4623 or When you take it
  - Covers NTSC and SD Video, Ucompressed
  - Covers VoIP
  - Basic Audio PCM and Soundcards
- Will Provide DV standards to read like 13818-1, 13818-2
- DV Texts are optional, but recommended
Extended Lab!

- You will Produce your Own Short DVD (Movie)
- Extended Lab Resources Web Page
  - Specification Given to You, Custom DV Algorithm at Frame Level
  - [http://ecee.colorado.edu/~ecen5653/ecen5653/labs/](http://ecee.colorado.edu/~ecen5653/ecen5653/labs/)
  - Groups of 2 Students or Individual
  - Collaboration Encouraged, But Credit MUST be Given to Sources of Help
  - Creativity and Variations on Approaches to Meeting Requirements Highly Encouraged

- Home Labs
  - Must have FC-14 (12 or better) Linux Installation
  - All MUST Use Open Content From our Web pages (SD, HD)
  - Optional OTA ATSC Equipment
    - DV projects require Hauppauge WinTV-HVR-1600 (Low Cost)
    - CX18 Driver for Connexant MPEG encoder with NTSC, ATSC, and Clear QAM input encoding
    - Can Use “Fair-Use” SD/HD Content from ATSC (Off-the-Air)
  - Optionally Acquire From Digital Cable Public TV Clear QAM (Cable)
    - Requires HD Tuner/Demod, e.g. Macro Imaging Technology myHD MDP-130 ([http://www.mitinc.co.kr/mitinc/e_site/](http://www.mitinc.co.kr/mitinc/e_site/), [http://www.DTVsolution.com](http://www.DTVsolution.com))
  - Optionally From NTSC or HD Cameras
    - We have Loaner SD NTSC Cameras
    - Pro-sumer HD Cameras that encode to MPEG4 ($250+)
SW and Optional HW Tools Used in Labs

- **Open Content or Optional OTA Content**
  - Frame Rate
  - Frame Compression
  - RT Transport (MPTS), Buffering
  - Sequencing and Dropouts
  - RT Stream analysis
  - HD Content analysis (Open Source from Orange Blender)
  - Record, Playback, Trick-play (FF, REW, Jump, Pause)
  - MPEG tools, transport stream analysis, your own code!

- **SW Tools - VLC, ffmpeg, Linux, GIMP**

- **Optional HW Content Capture/Creation**
  - Loaner Equipment on Campus Only or You Purchase
  - NTSC Cameras – CCTV
  - Capture Cards for SD DV Uncompressed PPM or MPEG Encoded
  - HW Encode/Decode with WinTV-HVR-1600 or WinTV-HVR-950Q and VLC on Laptop

© Sam Siewert
Extended Lab Grading

Choose Lab Partner Before Final Extended Lab
- On E-mail Sign up, Submit group preferences
- Due with last regular lab
- Can’t guarantee partners, but will attempt to honor preferences

Requirements for Extended Lab
- Your MUST Develop and Implement Original DSP/Image Processing
- Apply your Algorithm to Each Frame
- Accelerate with Threading and Vector Processing
- Demonstrations can be given to TA/Prof anytime during extended lab for check-off as you go (last 3 weeks)
- Final DVD Must be Submitted to Professor, You can Also Demo (Recommended)
Linux and Fedora Core Skills

Introduction Session
Linux Installations


- CUDA 3.1
  - FC-12
  - Ubuntu 9.10

- CUDA 4.0 / 3.2
  - FC-13
  - Ubuntu 10.10

- CUDA 4.1 RC2
  - FC-14
  - Ubuntu 11.04
What are NPTL POSIX Threads?

- [http://eceee.colorado.edu/~ecen5653/ecen5653/code/](http://eceee.colorado.edu/~ecen5653/ecen5653/code/)
  - User=ecen5033, pass=cubuffs
  - Start looking at this code and playing with it

- Thread Creation and Scheduling Policy
- Thread Parameters
- Thread Entry Points and Functions
- Thread Completion

- Intro to First Lab – Check CULearn for Labs!
Let’s Look at Some Code

- Threading and Image Segmentation for Many-Core and GPGPU
  - Example to be Published in IBM Paper on 1/24
  - Example Code Used in First Lab – Threading!!
  - Grids for Threading, Step One toward GPU Acceleration

- Examples of Video/Still-Frame Enhancement