Software Radio Retrospective

Past, Present, and the [Near] Future

Matt Ettus
Ettus Research
matt@ettus.com

Introduction and [My] History

Introduction

Who Am I?

- Core GNU Radio contributor since 2001
- Designed USRP in 2003
- Founded Ettus Research in 2004
  - Acquired by National Instruments in 2010
- Attended and exhibited at SDRForum/WinnComm since 2006

Please Ask Questions!

A Brief History of SDR

- SPEAKEASY project starts in 1991
- Joe Mitola coined the name “Software Radio” in 1991
- SDR Forum founded in 1996 (as MMITS)
- Matt writes his first software radio app in 1996
- Matt hears the term “Software Radio” for the first time in 1997
- Vanu Inc in founded in 1998
- GNU Radio project founded by Eric Blossom in 2001
- Vanu GSM basestation in 2003
- USRP developed in 2003 and 2004
- OpenBTS GSM basestation first demo in 2008 (on USRP)
- Full SDR LTE implementation in 2012 (on USRP)

Why Software Radio [for me]

- Hardware prototyping is slow and expensive
- Parts get obsoleted
- Great for experimentation
- True believer in open spectrum
  - Now known as dynamic spectrum, white spaces, underlays, overlays, unlicensed bands, etc.
- Future proofing, ride Moore's Law, etc.

The GNU Radio Story

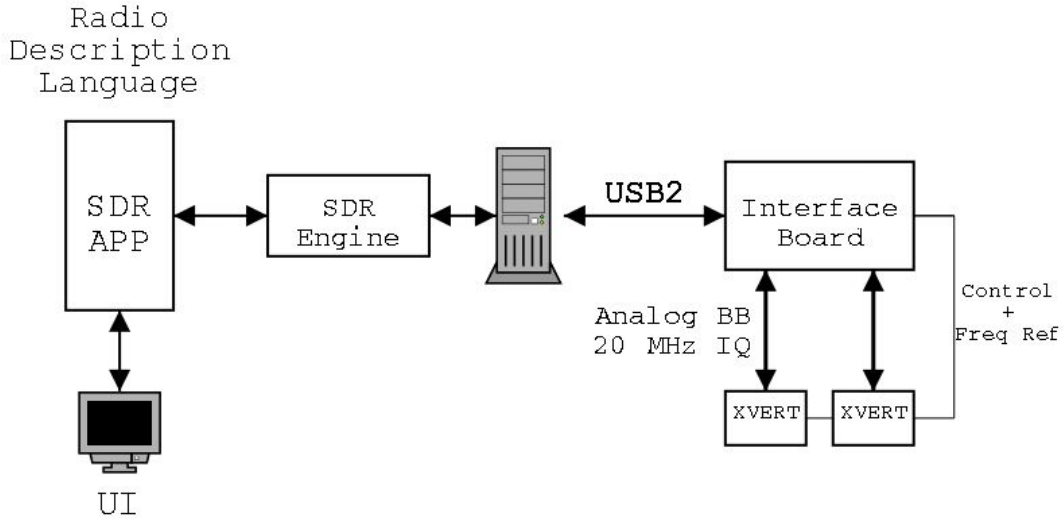
GNU Radio is

- A FREE framework for Software Radio
- A platform for real-time signal processing on commodity hardware
- A graphical design environment for radio
- A community

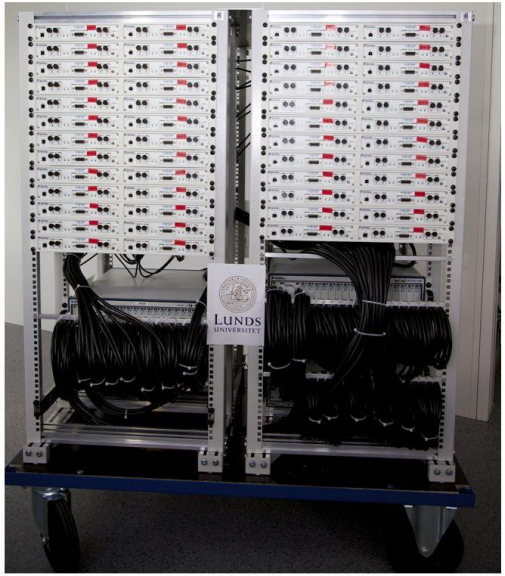
GNU Radio started out in 2001 as a fork of Spectrumware/pSpectra from MIT (The original Spectrumware people started Vanu Inc)

The USRP Story

- I wanted to experiment with SDR, MIMO, and Open Spectrum
- No reasonable devices existed
  - Tried sound cards, data acquisition boards, eval boards, etc.



USRPs Today



(a)

(b)

Computation and Software

Computation for Signal Processing

- Several major paradigms – CPU, GPU, DSP, Massive Multi-core, FPGA, ASIC
- Open Problems
  - Compiling for complex architectures
  - Transparently mapping computation across different architectures
- Trade offs
  - General purpose vs. specialized functional units
  - Developer performance vs. application performance
  - Intelligent compilers vs. armies of cheap programming labor
  - Custom systems vs. standards and high volumes
  - Single-purpose vs. multifunctional devices
  - Power vs. Performance

General Purpose Processors

- SIMD instructions in 90's
  - Sun's VIS 1994
  - Intel's MMX in 1996
- Single-threaded wall hit in 2006 at 4GHz (power)
- Today Parallelism is key
  - 16+ Cores 32+ Threads, ~4 GHz, 512-bit wide AVX in Core i7
- Open Issues
  - Resource sharing with the rest of the system
  - Real-time issues in the OS
  - Harder to program than they used to be

FPGA

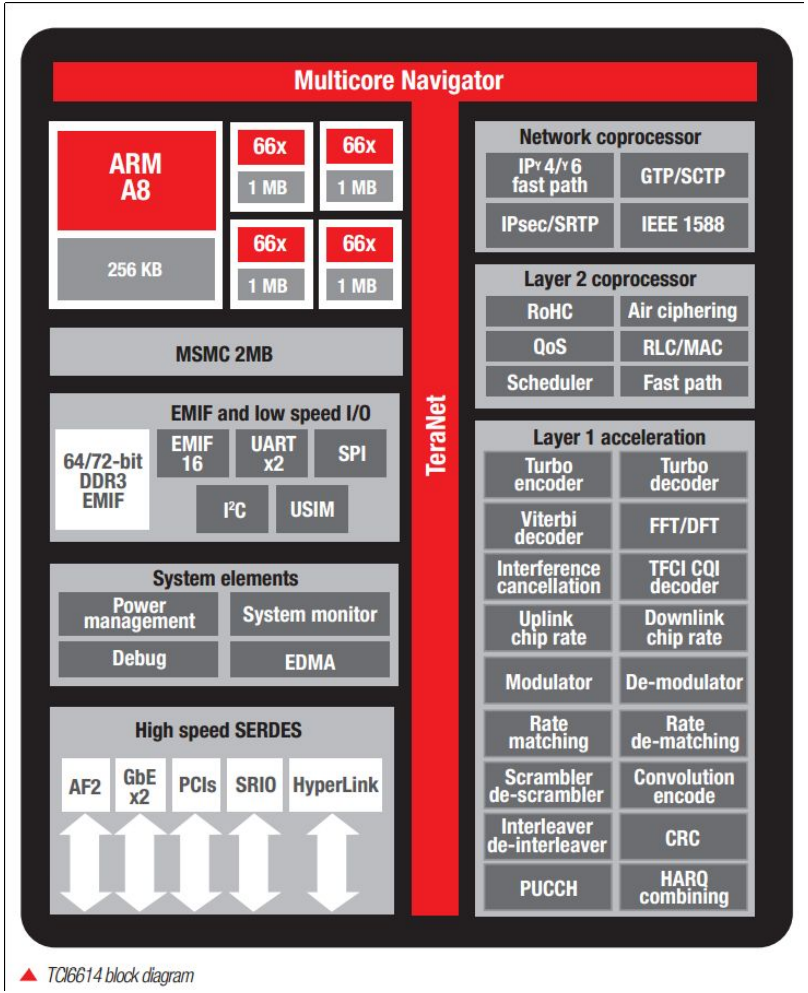
- 1998 -- basically fancy expensive CPLDs
  - Tiny, slow, glue logic
- 2004 -- No hard multipliers for a reasonable price
- Today -- 1000+ GMACs for a couple hundred dollars
- Hard to program
  - Low level gateware (Verilog and VHDL) still most common
  - Graphical design still in its infancy
  - High-level languages like OpenCL and VivadoHLS (C/C++)
  - Component interoperability a problem
  - RFNoC to the rescue!

GPU

- 1998 -- 3D only (Quake on 3DFX Voodoo)
- First GNU Radio GPU work in 2007
- CUDA released in 2007
- Today -- general computation common
  - Thousands of threads
  - Extreme bandwidth
- Open Issues
  - Poor latency
  - Difficult to program
  - Programmer must understand lowest levels of design
  - CUDA is vendor specific
  - OpenCL to the rescue?
    - Still won't run on this laptop...

Programmable DSP

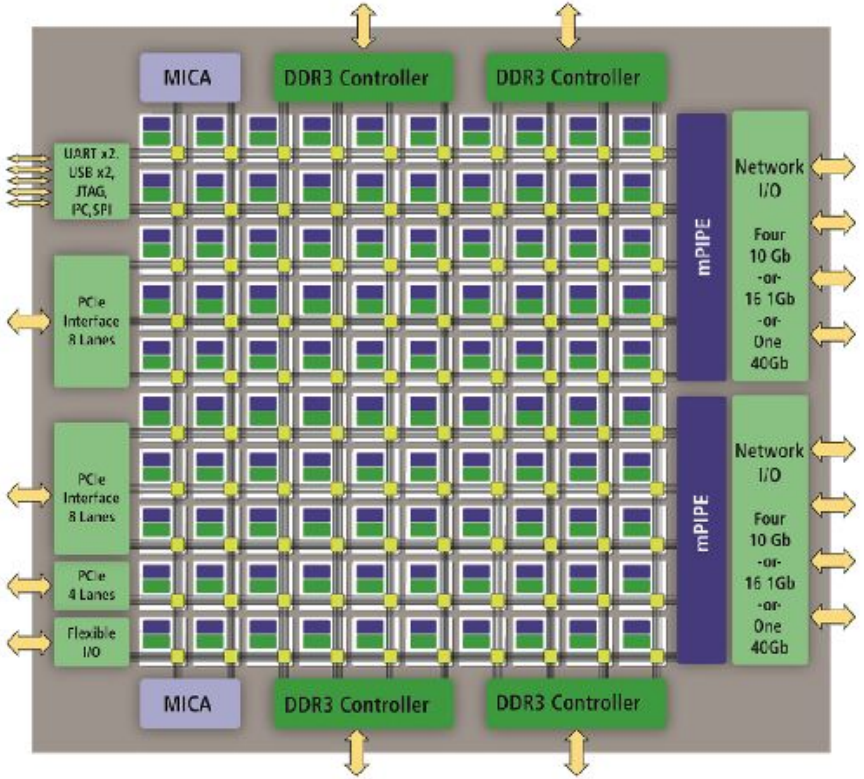
- 1996
  - Good at FIR and IIR filters, FFTs
  - ~20 MIPS
- Today
  - Small programmable section Good at FIR and IIR filters, FFTs
  - Lots of special function units
  - ~80 GFLOPs
  - Harder to program than they used to be
  - Far fewer vendors than there used to be



TCI6614 block diagram

Massive Multicore Processors

- Sandbridge, Coherent Logix, Picochip, Tilera, Adapteva, etc.
  - Hard to program, proprietary
- Intel Xeon Phi (aka Larrabee)
  - Starts to look a lot like GPUs with x86 instruction set...
- 1000-10000 core chips coming soon
  - Limited periphery, limitations of 2D-routing will come into play



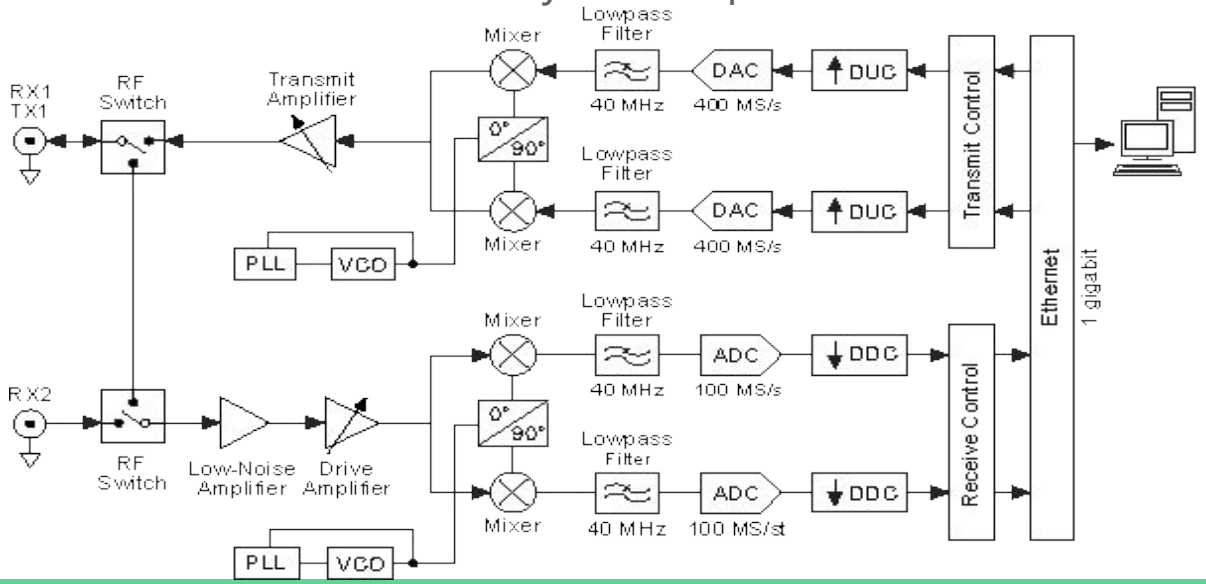
RF Hardware

Challenges in RF Hardware

- Dynamic range of wideband low-power radios
- Flexible filters
  - More than 35 TDD and FDD LTE bands worldwide
  - Numerous attempts at electrically or mechanically tunable filters
    - Poor linearity, high loss, low reliability, high cost
- Flexibility costs power

Direct Conversion Radio

- Lowest cost, most common SDR architecture
- Very flexible, multiple-octave frequency coverage
- Bandwidths being scaled to >2GHz
- IQ Balance and DC offset are mostly solved problems



Direct-to-ADC Receivers

- Holy grail of SDR design
- Not really practical for general coverage devices
  - Except with extensive front-end filters, as in Carrier Aggregation base stations
- Current state of the art -- TI ADC12J4000
  - 12-bit 4 GS/s
  - SFDR ~-60dBc (5 dB better than 2 years ago)
  - 2 Watts (half as much as 2 years ago)
  - 20 dB Noise Figure
  - With 20dB gain -->
    - -16dBm full scale (strong signals clip)
    - -70dBm spurs!

“Everything” Chips

- Circa 2007
  - Motorola RFIC
    - 100 MHz to 4 GHz
    - Never produced commercially
  - Bitwave Semiconductor
    - Out of business
- Today
  - Analog Devices AD9361
    - 70 MHz to 6 GHz, 2x2 MIMO, 56 MHz BW
  - Lime Microsystems
  - Dynamic range still limited in comparison with discrete designs

Antennas and Filters

- Not unusual to need to receive -110dBm signals in the presence of 0dBm signals
- Need good filters, but tuneable filters the holy grail
  - “Even with these new technologies, the RF content in a high-end smartphone will increase from $6 to $10 over five years.”
- Current hot research areas
  - Tuneable antennas
  - MEMS filters
  - RF-Optical filters
  - Full Duplex radios

Conclusions and Predictions

What have we accomplished

- Deployed SDR Cellphone Base Stations (GSM+LTE), including USRP-based ones
- Some white spaces radios finally coming to market
- Software Radio is the dominant paradigm for new defense applications
- Heavy presence in specialized radios, prototyping, research, etc.
- Huge corpus of code available
  - Open Source means you don't have to reinvent the wheel
  - Open Source leads to reproducible research
  - GNU Radio is the dominant platform for research in SDR, Cognitive Radio and DSA
- Numerous testbeds

What hasn’t happened [yet]

- Solving Public Safety interop problems
- SDR Cellphones
- SDR WiFi cards
- Dynamic Spectrum Access in the non-defense world
- Direct-to-ADC radios
- The everything radio

Predictions

- High-volume, low-cost, low-power radios will slowly become more software-based over the next 5 years (first mobile phones, then IoT much later)
- Software Radio will become increasingly important for design, test, and medium-scale deployment
- Mobile phone/data licensed spectrum will grow to occupy ~50% of Sub-5GHz spectrum
- TV White spaces will be dead in the developed world
- Government action on PCAST (3.5 GHz band) will reinvigorate commercial interest in Dynamic Spectrum
    - Won't be nearly as dynamic as promised...
- Defense radios will continue to be the biggest users of Dynamic Spectrum Access

More Predictions

- Nearly all commercial wireless communications systems will coalesce around a very small set of standards
  - The latest 3GPP/LTE
    - Public Safety radios will eventually just be ruggedized LTE devices with custom apps
  - The latest 802.11
  - One standard for IoT, keyfobs, remote controls, smart home, etc.
  - RFID/NFC
- 5G Will coalesce around 2 major technologies
  - Massive MIMO for sub-6GHz access
    - 80+ bits per second per Hz on USRPs today
  - mmWave for multiple Gbps per user, only viable in dense urban areas
    - Integrated phased arrays with many antennas
    - Only 2x2 to 4x4 MIMO (separate RF chains)

Thanks!

Matt Ettus

Ettus Research

matt@ettus.com

@EttusResearch