GPUDirect + SDR:
How to Move **One Billion** Samples per Second over PCIe

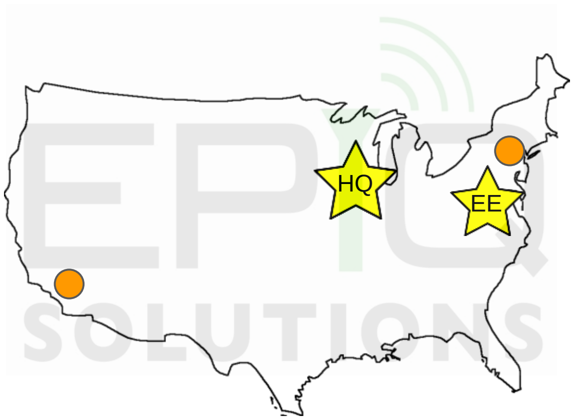
John Orlando - Epiq Solutions
(on behalf of our Sidekiq Development Team)

GRCon 2019

EPIQ SOLUTIONS

Company Background

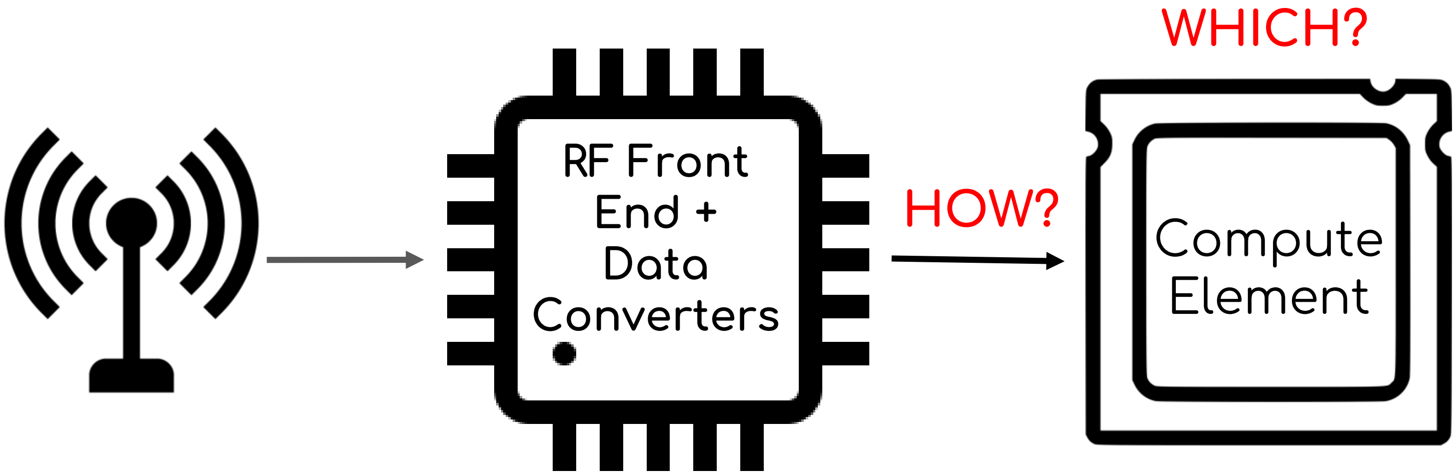
- Celebrating 10 years of delivering:
  - The smallest commercially available SDR modules in the world
  - Delivering complete RF sensing solutions to a broad range of commercial, defense, and security markets
- Team of ~40 (and growing), predominantly engineers
  - RF/hardware, software/DSP, FPGA, mechanical, test/production, sales
  - **HQ** in Schaumburg, IL (engineering + production/test)
  - “Epiq East” (**EE**) in Alexandria, VA est 2016 (sales/BD)
  - WE ARE HIRING! STOP BY OUR BOOTH



Regional Account Management (San Diego, CA + Nashua, NH)

GRCon 2019

Problem Statement



How Much Throughput is Needed These Days?

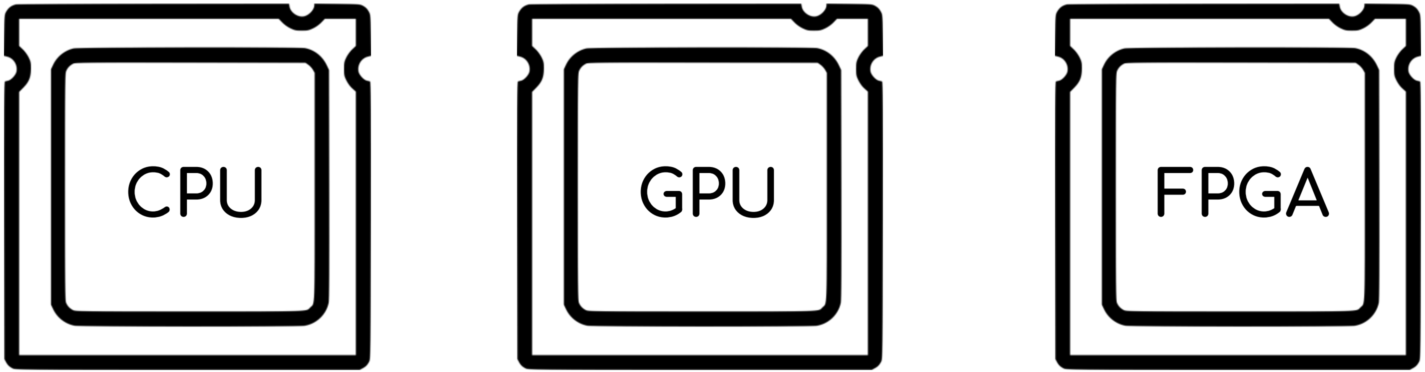
Wireless Standard	Max Bandwidth per Antenna	Typical Quadrature Sample Rate	# of Antennas	Baseband I/O Bandwidth Required (sans overhead)*
GSM / 2G	200 KHz	541.667 Ksps	1	2.1 MBps
UMTS / 3G	5 MHz	7.68 Msps	1	30.72 MBps
LTE / 4G	20 MHz	30.72 Msps	1 to 4	122.88 MBps to 491.5 MBps
802.11n	40 MHz	40 Msps	1 to 4	160 MBps to 640 MBps
802.11ac	160 MHz	160 Msps	1 to 4	640 MBps to 2560 MBps
5G-NR	400 MHz	491.52 Msps	1 to 4+	1964 MBps to 7856 MBps+

So...Many...Samples...

*Assumes 16-bits 'I' and 16-bits 'Q' for each sample

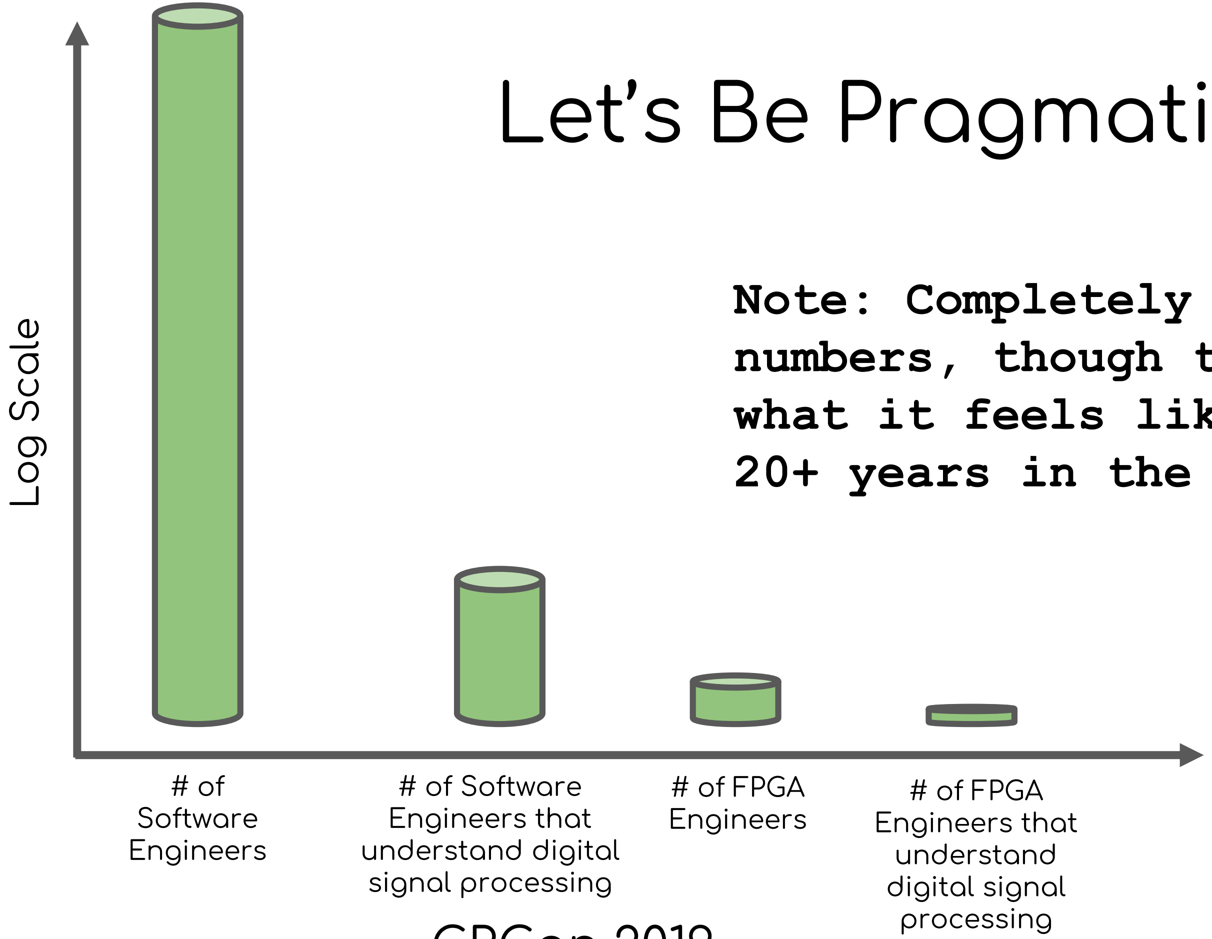
EPIQ SOLUTIONS

Options for Compute Elements



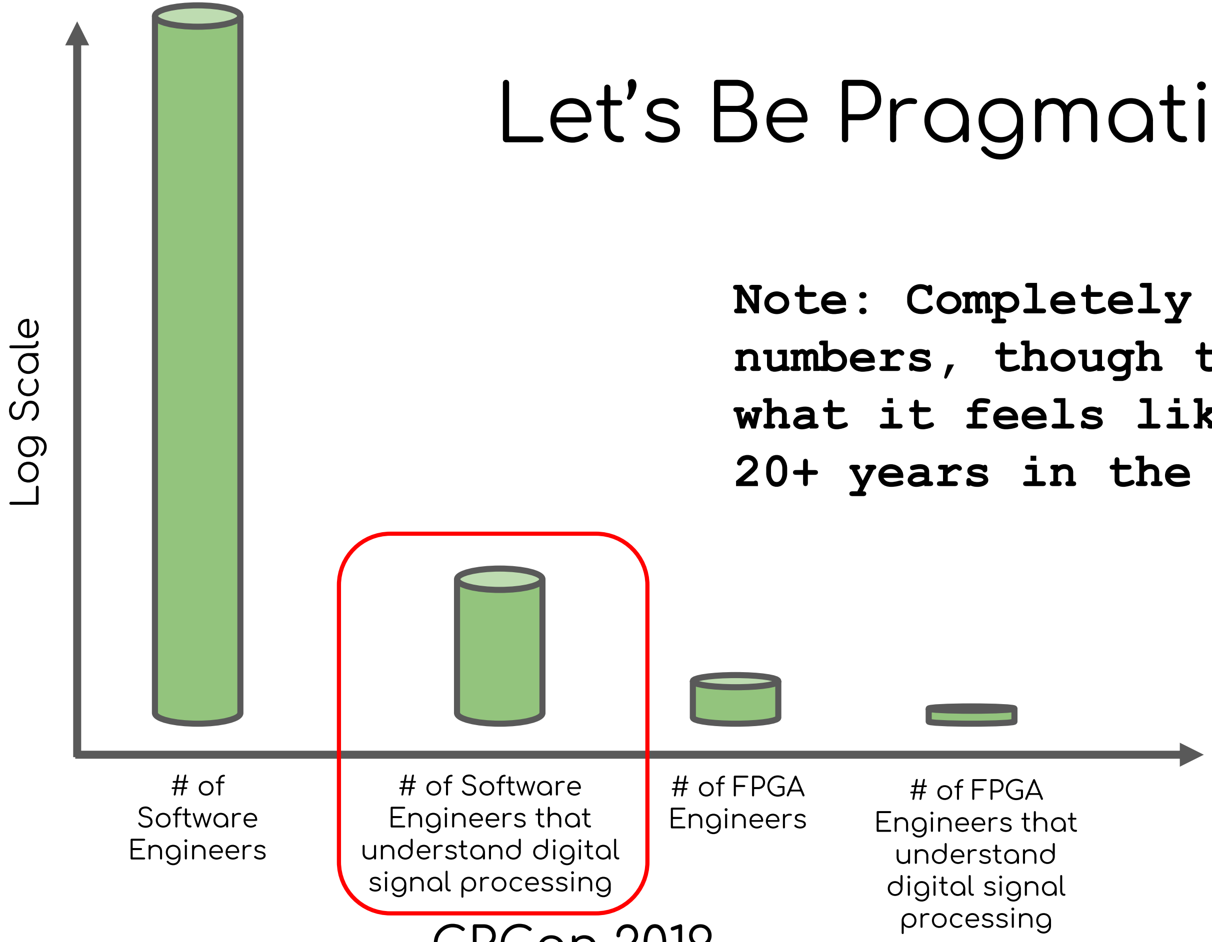
(also DSPs, but much less common in SDR these days)

Let's Be Pragmatic



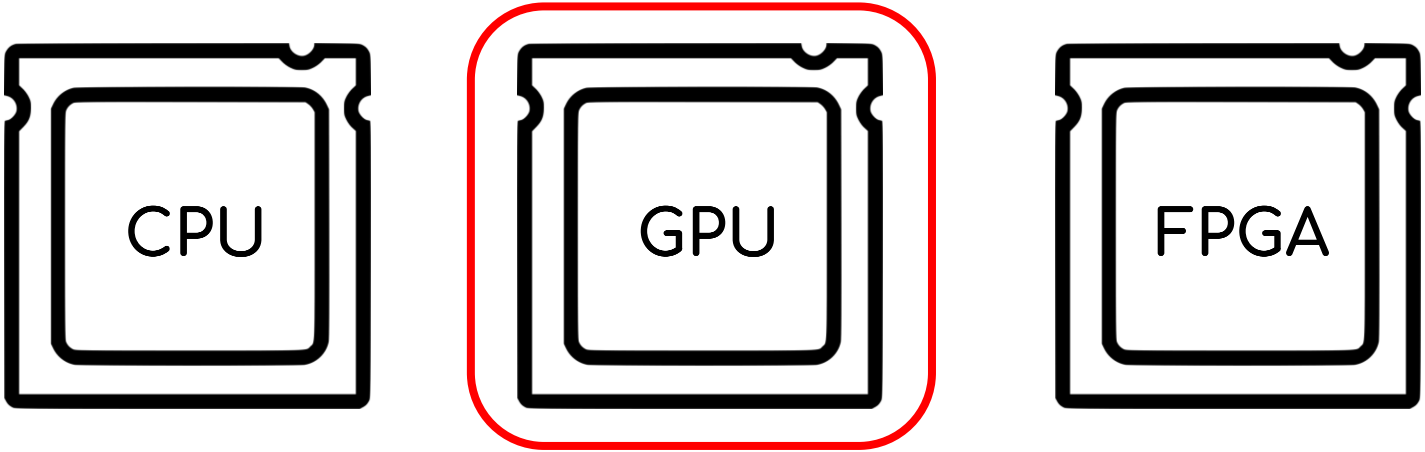
Note: Completely made up numbers, though this is what it feels like after 20+ years in the industry

Let's Be Pragmatic



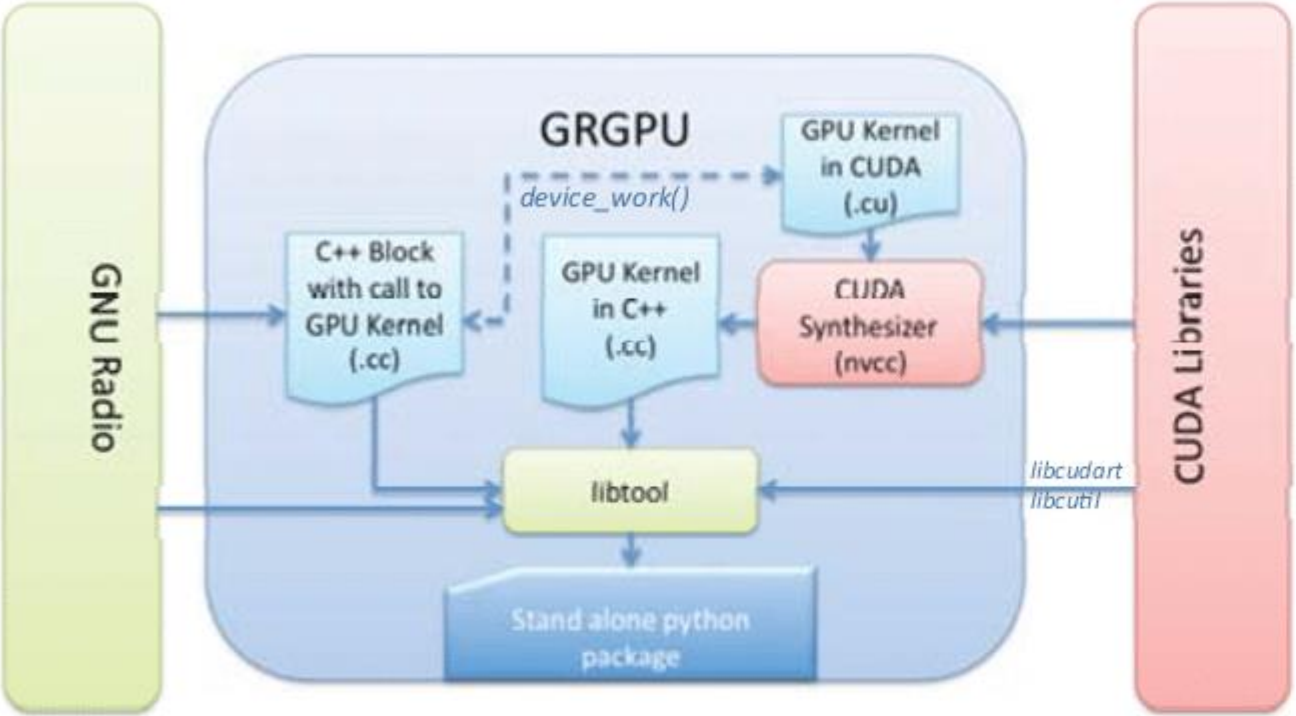
Note: Completely made up numbers, though this is what it feels like after 20+ years in the industry

Options for Compute Elements



Eureka! Write software for a GPU to do SDR!

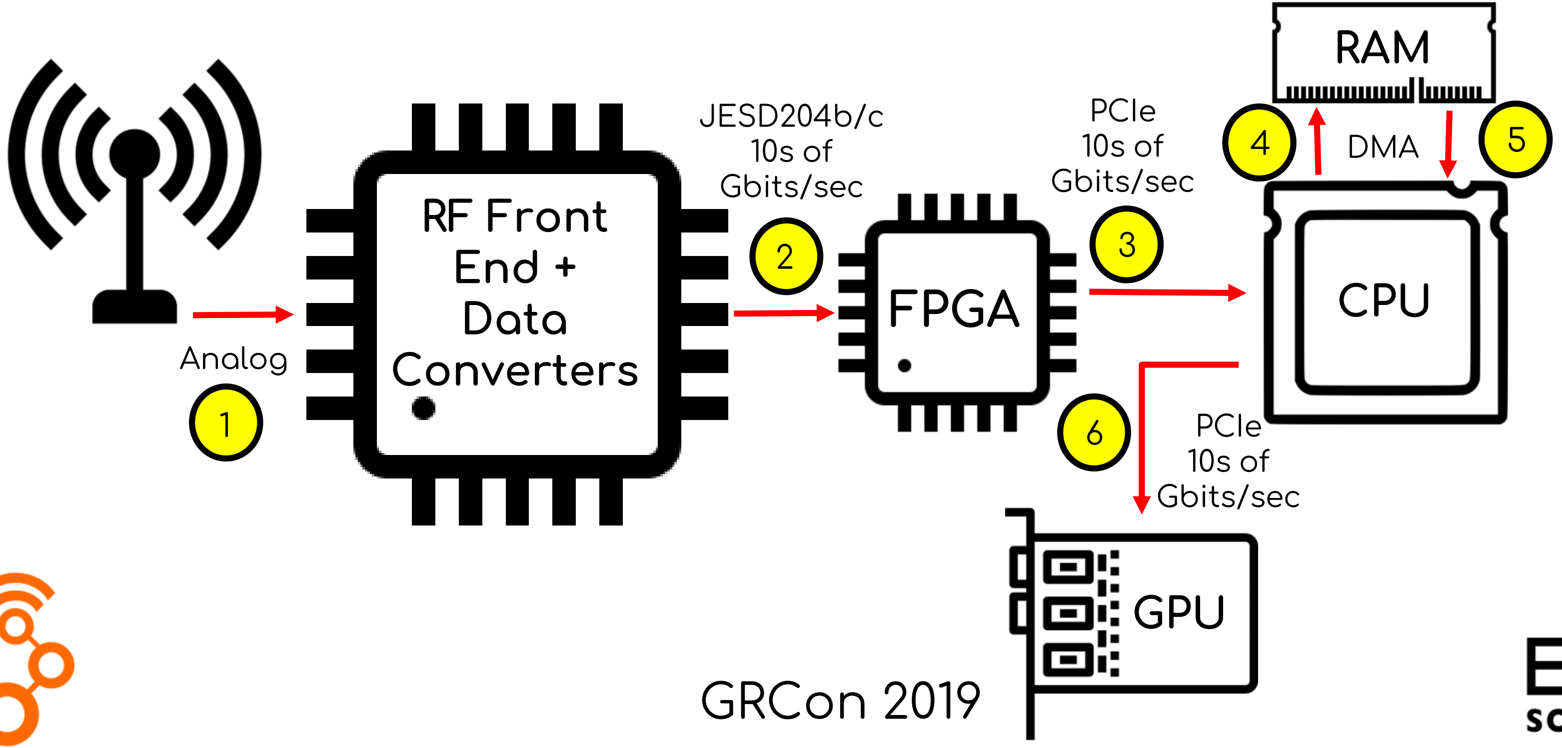
Previous Published Work



GR-GPU work from 2011

(work and image courtesy of Plishker, Zaki, Bhattacharyya, Clancy, and Kuykendall)

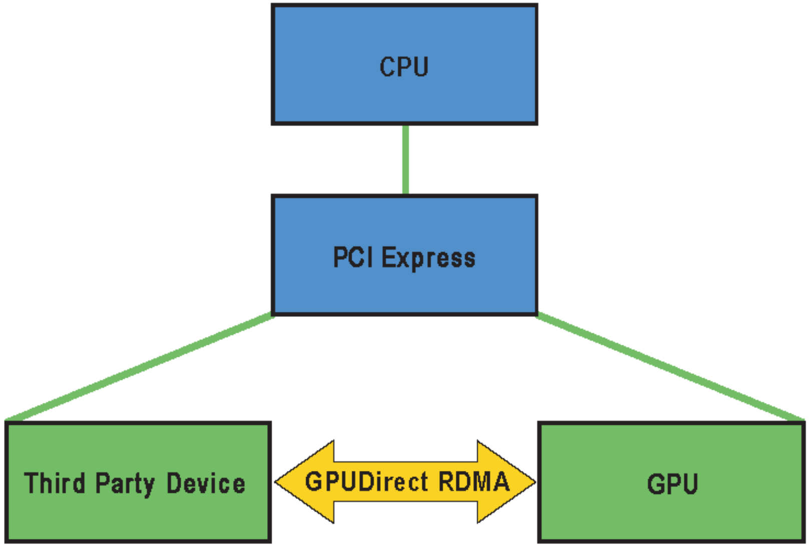
Using GPUs with SDR: Classic Architecture



Can We Do Better?
Can We Increase Throughput?
If So, By How Much?

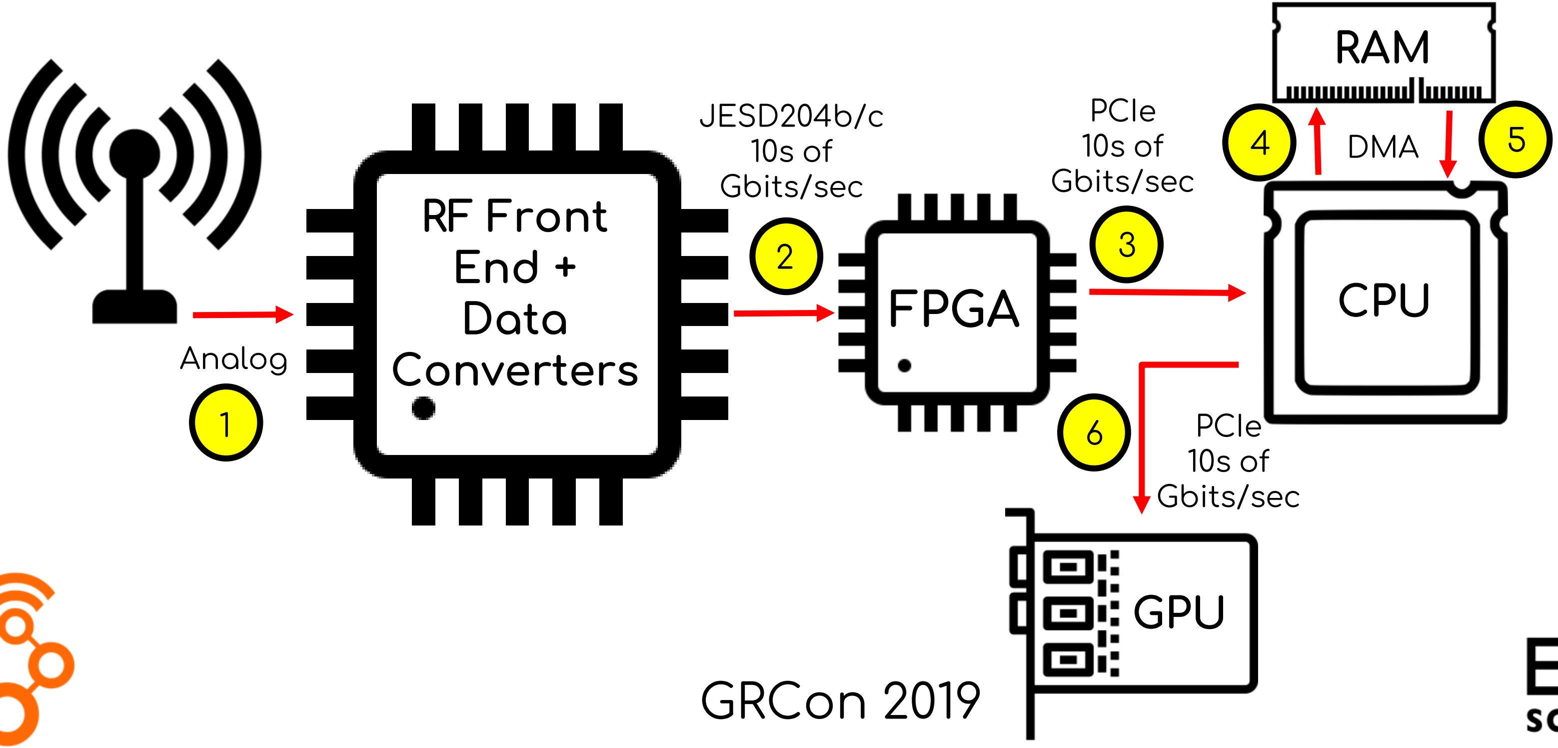
GPUDirect RDMA

- What is it?
  - Developed by Nvidia, first introduced in 2013
  - Software driver to allow PCIe devices to target DMA transactions for GPU memory instead of CPU memory
  - CUDA-based computation in GPU for high-end parallel signal processing
  - CPU sets up transactions and then gets out of the way
  - In theory:
    - More efficient
    - Higher throughput
    - Fewer CPU cycles
    - Lower system power consumption
    - Winning!



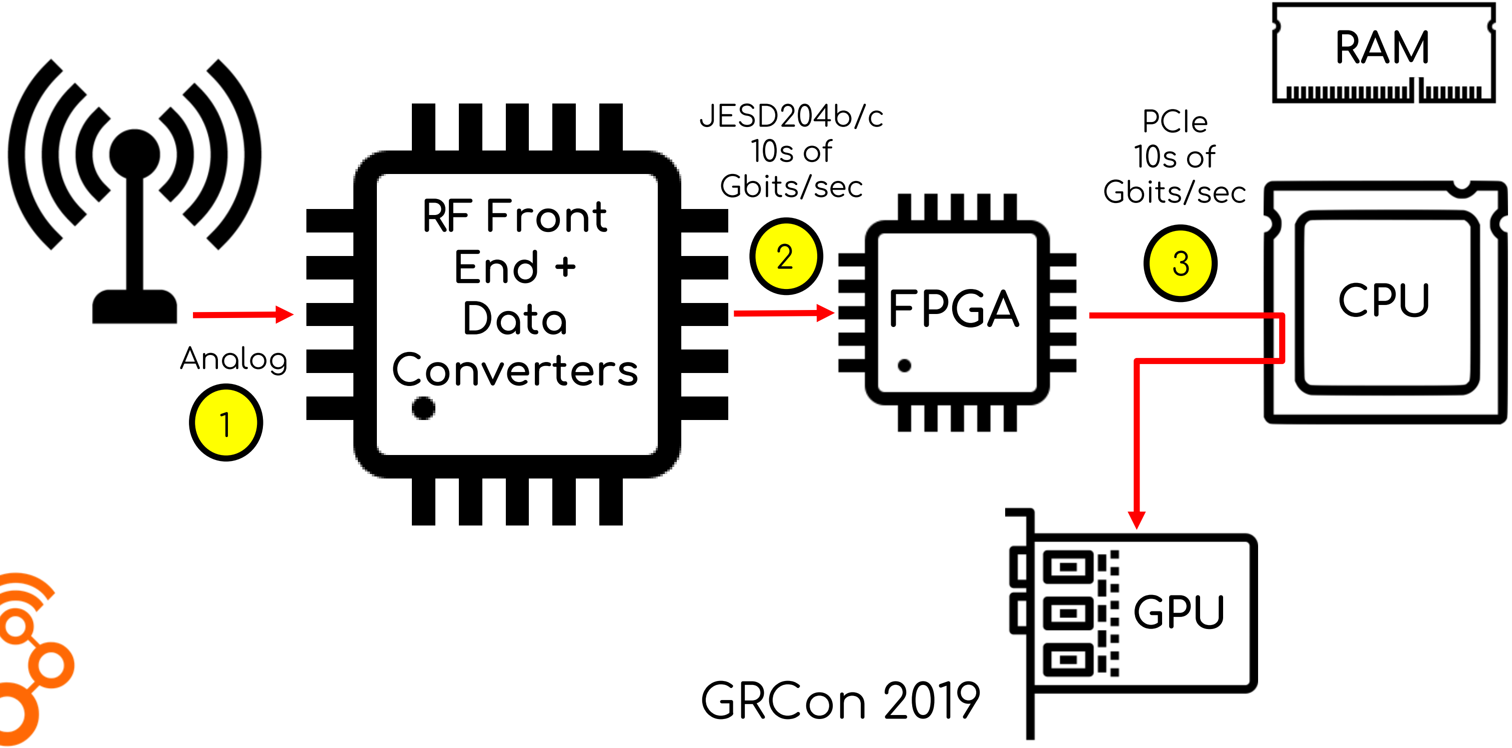
(Image courtesy of Nvidia)

From This...

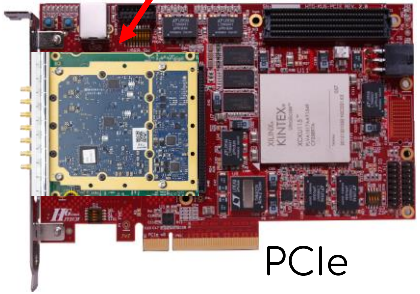


Analog
1
RF Front End + Data Converters
JESD204b/c
10s of
Gbits/sec
2
FPGA
PCIe
10s of
Gbits/sec
3
RAM
4
DMA
5
CPU
6
PCIe
10s of
Gbits/sec
GPU

...To This


RAM
JESD204b/c
10s of
Gbits/sec
PCIe
10s of
Gbits/sec
RF Front
End +
Data
Converters
2
FPGA
3
CPU
Analog
1
GPU

What You Need - PCIe-based Radio Module

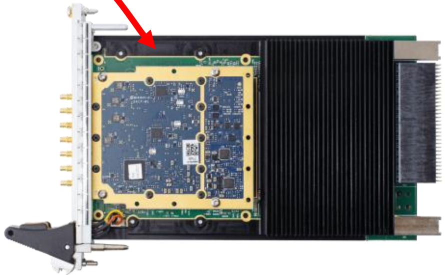


PCIe

- For installation in rack mounted servers or standard motherboards
- PCIe full height, half length carrier
- Xilinx Kintex Ultrascale FPGA
- Gen3 PCIe x8

Sidekiq X4

- VITA 57.1 FMC Card
- 4x4 MIMO (dual ADRV9009 RFICs from ADI)
- Up to 800 MHz IBW
- Integrated Rx pre-select filters
- Supported by our standard libsidekiq API



PCIe

- For installation in 3U VPX systems
- Xilinx Kintex Ultrascale FPGA + Zynq Ultrascale+ SoC
- Gen3 PCIe x8

	Sidekiq MiniPCIe	Sidekiq M2	Sidekiq Stretch (M2 2280)	Sidekiq X2	Sidekiq X4	Sidekiq VPX4
Max # RF Rx / Tx	2 / 1	2 / 2	1 / 1	3 / 2	4 / 4	4 / 4
RF Tuning Range	70 MHz - 6 GHz	70 MHz - 6 GHz	70 MHz - 6 GHz	1 MHz - 6 GHz	1 MHz - 6 GHz	1 MHz - 6 GHz
Integrated Rx Pre-Select Filters	No	No	Yes	Yes	Yes	Yes
Max total IBW	50 MHz	50 MHz	50 MHz	200 MHz	800 MHz	800 MHz
ADC/DAC sample width	12/12-bits	12/12-bits	12/12-bits	16/14-bits	16/14-bits	16/14-bits
FPGA on-board	Xilinx Spartan 6 LX45T	Xilinx Artix 7 XC7A50T	Xilinx Artix 7 XC7A50T	None (FPGA on host)	None (FPGA on host)	Xilinx Kintex Ultrascale KU115
Linux computer on board	None	None	None	None	None	Quad-core ARM A53
Host Interface	Gen1 PCIe x1 + USB2	Gen2 PCIe x1 + USB2	Gen2 PCIe x1	FMC HPC*	FMC HPC*	Gen3 PCIe x4 + Ethernet
Typical power consumption	2-2.5 W	2.5-3.5 W	1-2 W	4-10 W	6-18 W	25-50W
Size	30 x 51 mm	30 x 42 mm	30 x 51 mm	69 x 77 mm	69 x 77 mm	100 x 160 mm

All Have PCIe

*FPGA card would typically provide PCIe interface to host

GRCon 2019

EPIQ SOLUTIONS

What You Need - CPU + GPU Compute Element

+

Quadro P2000

- Motherboard with x86 CPU + Nvidia GPU card
- GPUDirect requires the GPU card and radio device be on same PCIe root complex

- Nvidia Xavier platform
- Integrated octa-core ARM + GPU on embedded module

(Linux Only)

How Does GPUDirect Work, Roughly:

- **Step 1:** CPU pins physical memory buffers in GPU
- **Step 2:** CPU informs FPGA of physical memory buffers via DMA descriptor list
- **Step 3:** CPU gets the hell out of the way
- **Step 4:** FPGA can perform DMA of I/Q directly into GPU pinned memory
- **Step 5:** GPU crunches I/Q like a champ
- **Step 6:** Goto Step 4

Does it Work? Let's Find Out...

- Updated the PCIe DMA driver used with our Sidekiq SDR cards to support GPUDirect
- Built two hardware test platforms for experimentation
  - Platform 1: Intel Core i7 CPU in Gigabyte motherboard with Nvidia P2000 GPU card
  - Platform 2: Nvidia Xavier platform with octa-ARM + GPU on module
- Both platforms run Linux (GPUDirect isn't supported in Windows)

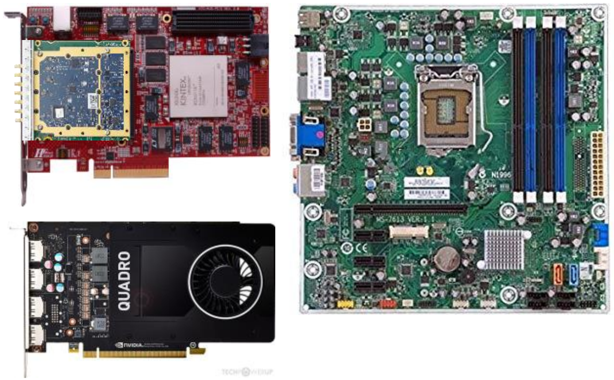



Test Scenario

- Test 1 (aka “the old way to use a GPU”)
  - FPGA -> CPU -> RAM -> CPU -> GPU
- Test 2 (aka “the GPUDirect way”)
  - FPGA -> GPU
- Perform simple computational kernel in GPU
  - FPGA generated known payload in place of I/Q data
  - Calculated average of transported data and returned it to CPU for validation
- Measure transport throughput into GPU

Test Results...Drumroll Please...

Platform 1



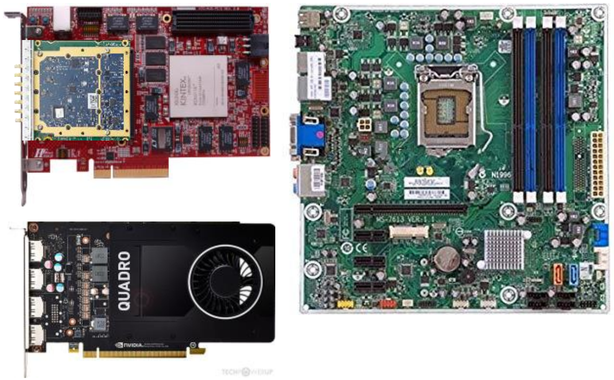
- Test 1 (the old way): 1718 MB/sec
- Test 2 (the GPUDirect way): 1733 MB/sec

- 1% performance improvement!

- Yay? WTH
- Headscratching ensues...
- Check the hardware

Test Results...Drumroll Please...

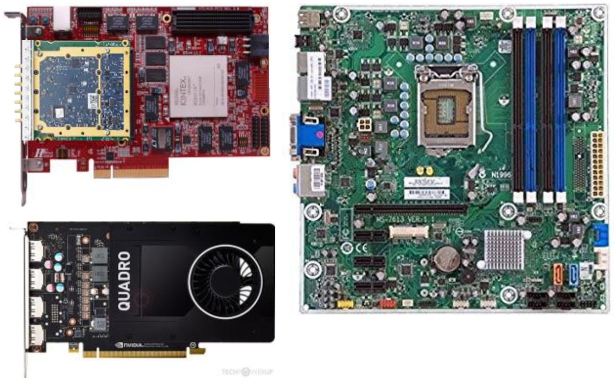
Platform 1



- FPGA loaded with a PCIe Gen2 x4 DMA engine
- Let's Math
    - PCIe Gen2 = 5 Gbits/sec per lane
    - x4 lanes = 20 Gbits/sec
    - Reduce for 8b/10b coding = 16 Gbits/sec
    - Real-world PCIe bus utilization of ~85% = 13.6 Gbits/sec
    - In bytes: 1700 MB/sec
    - In other words:
        - The PCIe bus is the limiting factor here
        - The FPGA->CPU->RAM->CPU->GPU path <u>is just as fast</u> as the FPGA->GPU path
    - So what now?

Test Results...Drumroll Please...

Platform 1, Take 2

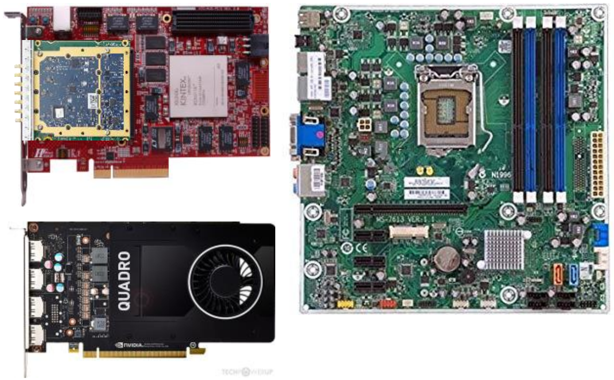


- FPGA now loaded with a PCIe Gen3 x8 DMA engine
- Motherboard supports up to Gen3 x16
  - PCIe Gen3 = 8 Gbits/sec per lane
  - x8 lanes = 64 Gbits/sec
  - Uses 128b/130b coding = 63 Gbits/sec
  - Should be PLENTY of throughput

- Re-test and hope sanity is restored

Test Results...Drumroll Please...

Platform 1, Take 2



- Test 1 (the old way): 3219 MB/sec
- Test 2 (the GPUDirect way): 4677 MB/sec

- 45% throughput improvement!

- Sanity is restored

Test Results...Drumroll Please...

Platform 2

- Test 1 (the old way): 2079 MB/sec
- Test 2 (the GPUDirect way): 3035 MB/sec

- 46% throughput improvement!

- Nearly identical % improvement compared to x86

Yes Virginia...

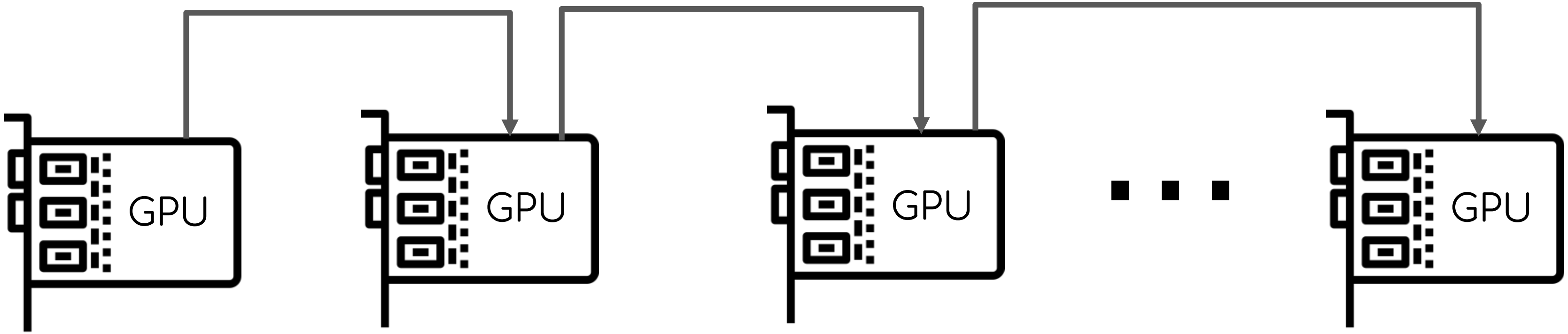
- ...you can move > 1 billion samples/sec over PCIe into a GPU using GPUDirect RDMA
  - 4677 MB/sec equates to ~1.17 Gsamples/sec for 16-bit ‘I’ and 16-bit ‘Q’
  - Enough to digitize 1 GHz of RF spectrum
- ...SDRs based on PCIe provide the most efficient, scalable, low-power transport out there today
- ...GPUDirect + SDR is a natural fit...and using a PCIe-based SDR is a near optimal solution

Does it Matter?

- If you care about single channel bandwidths in the range of 1 GHz...
- If you care about MIMO (2x2, 4x4, etc) where channel bandwidths are hundreds of MHz...
- If you want to keep your CPU cycles for something else...
- If you want to implement signal processing applications in GPUs...

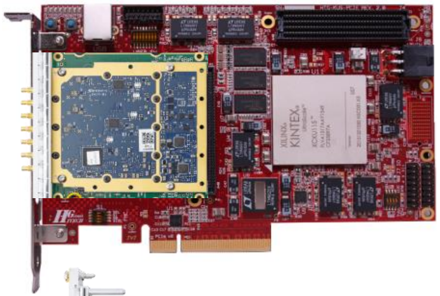
- ...then YES...GPUDirect matters...BIG TIME
- (It may be the only way to achieve the desired results)

Expanded Processing Options



GPUs can be chained together via NVLink interconnect

GPUDirect Support in Libsidekiq



- Initial support coming in 4Q2019
- Sidekiq X4 capability
    - 4-channel Rx @ 245 Msamples/sec (phase coherent)
    - 2-channel Rx @ 491 Msamples/sec (independently tunable)
    - Support for PCIe carrier (server deployment)
    - Support for 3U VPX carrier (ruggedized deployment)
- Sidekiq Stretch capability
    - 1-channel Rx a 61.44 Msamples/sec

Areas for Exploration

- Improved DMA descriptor management
  - CPU is still involved...for now
- More optimal data transport size
  - 4K DMA blocks currently, but GPUs like BIG chunks of data
- Handling of meta-data in I/Q stream
  - GPU checks timestamps in GPU?
  - GPU plucks out gain value for normalization?
  - Raw I/Q sans meta-data?
- Integration with GPU frameworks
  - Photon (http://photon.mitre.org), others
- OpenCL support?
  - DirectGMA / bus-addressable memory

Quick Shout Out

- …to MITRE Corp for helping fund a portion of this GPUDirect development work
- …to our Sidekiq team doing the heavy lifting
- …to our customers that continue to use and deploy Sidekiq in all kinds of interesting applications all over the world

If you are at the top of your game…

- …and you like building hardware or software
- …and you like working on hard problems
- …and you like being surrounded by a team of capable makers that know how to get things done

WE ARE HIRING
COME TALK TO US AT OUR BOOTH

(or send an email to jobs@epiqsolutions.com)

THANK YOU GRCON 2019!