When Hardware is WRONG...

Or... *They can fix it in software*

Jason L. Wright
jason@openbsd.org
NYCBSDCON 2008

About Me...

- Not Chewbacca
  (he may be related, though)
- Asthetically challenged
- Not *from* Idaho
- But I live/work there...

Why does hardware go wrong?

- Time to market pressure
- Lack of understanding of performance impact
- Or perhaps, hardware folks hate software folks

Bad Implementation

- Sometimes things go wrong despite good intentions...
- Bluesteelnet 5501
  - Has good quality random number generator
  - And a weak output transitor, can't pull line low

sys/dev/pci/ubsec.c

Interrupts

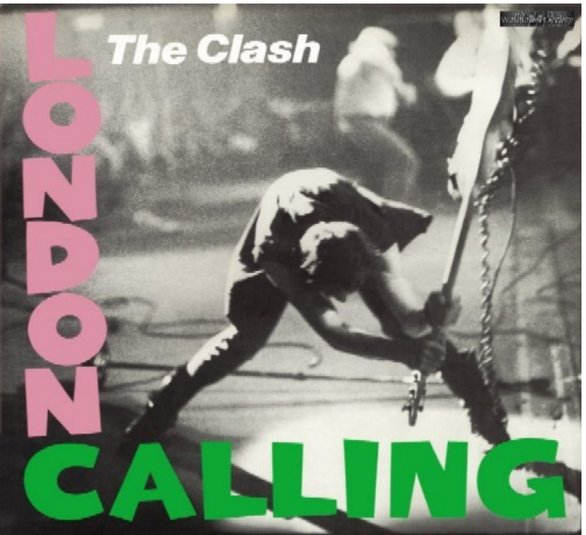
- Interrupt is a signal from hardware that the CPU needs to do something
- Generated externally by network cards, timers, etc.
- Forms the basis of UNIX operation
  - Process scheduling
  - Packet processing

So, why implement interrupts?

- The alternative is polling (boo, hiss!)
  - Faster (ok, yeah, I'll buy that)
  - At the expense of WASTED TIME!
  - This will not do for event driven OS's
- SafeNet SafeXcel 1141 – no interrupt implemented for public key crypto completion!
- “Well, you can estimate how long it will take and then poll.” Um, yeah.
- sys/dev/pci/safe.c

Edges and Levels

- Two types of interrupts
  - Edge (ISA, boo! Hiss!)
  - Level (everything else)
- Sharing only possible with level triggered ints
- But, you have to know who is calling!



AMD7930 audio

- No way to know if it interrupted
- So, “maybe” it did!

```
foreach interrupt_handler x {
    if (interrupted(x))
        return;
}
```

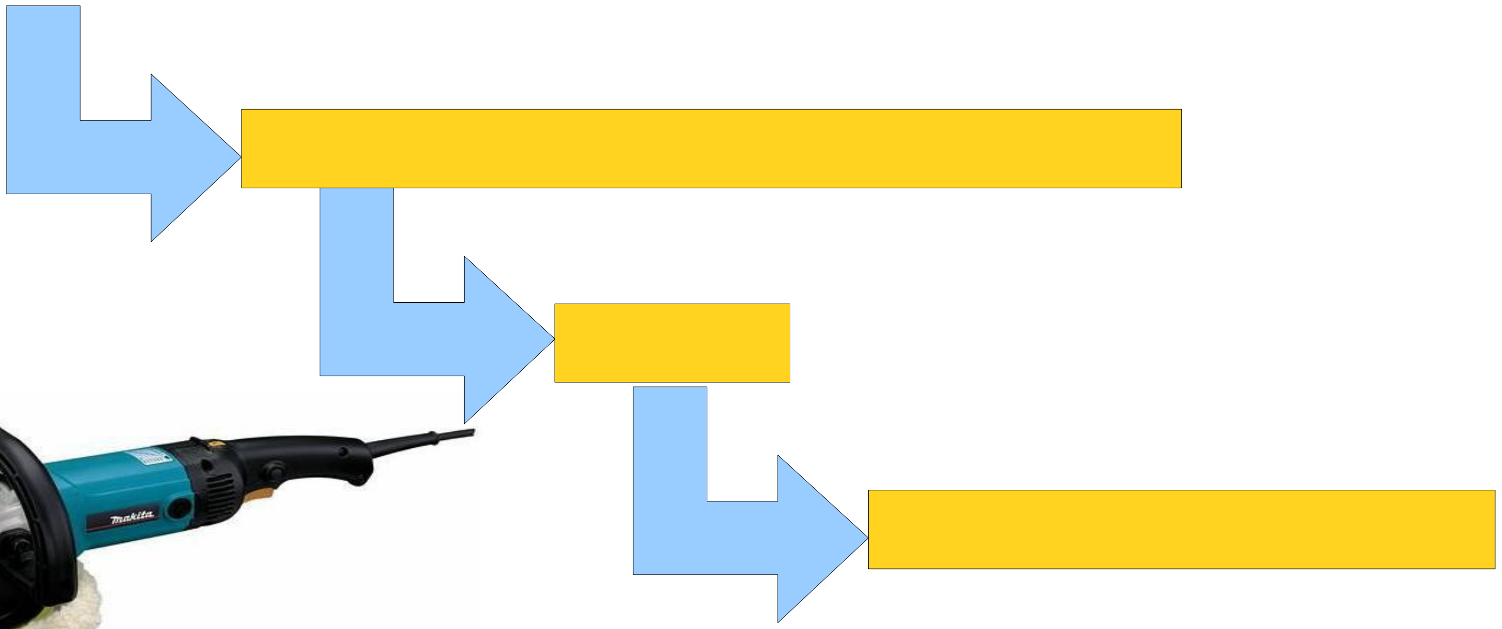
sys/arch/sparc/dev/amd7930.c

Non-Contiguous Buffers

- Form the basis for disk I/O (struct iovec)
- Also basis for network I/O (struct mbuf)

  (mbufs are called skbuf's under Linux, if anyone cares)

- Aka scatter-gather I/O



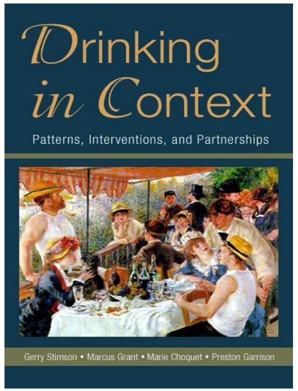
Contiguous Buffers

- Sure, you can make non-contig buffers contig
- Allocate a big buffer, copy
- Do operation
- Copy big buffer BACK into non-contig buffers
- Ah, but there's a wrinkle (or two...)

Dealing with non-contiguousness

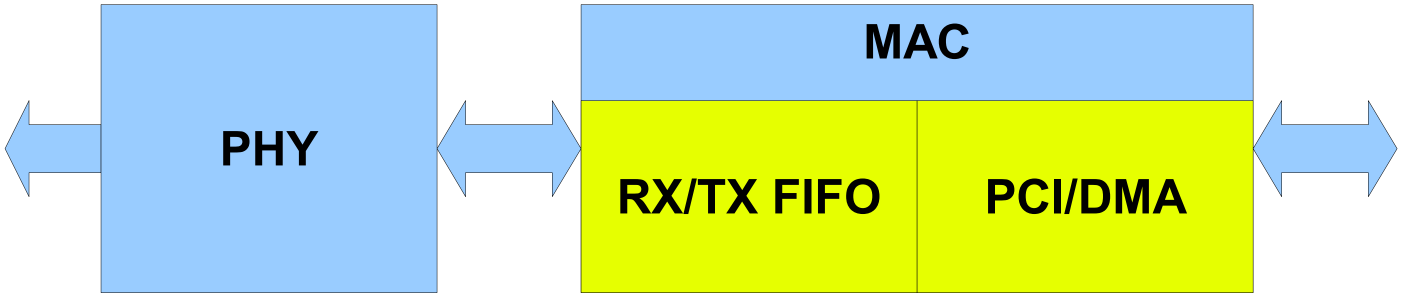
- Can't allocate DMA-able memory in an interrupt context (the DMA map isn't interrupt safe)
- So, either pre-allocate the biggest buffer(s) you'll ever need (ew!)
- Or, provide a process context (double ew!)
- NetOctave NSP2000 doesn't support scatter-gather... oops... and performance is horrible

sys/dev/pci/noct.c

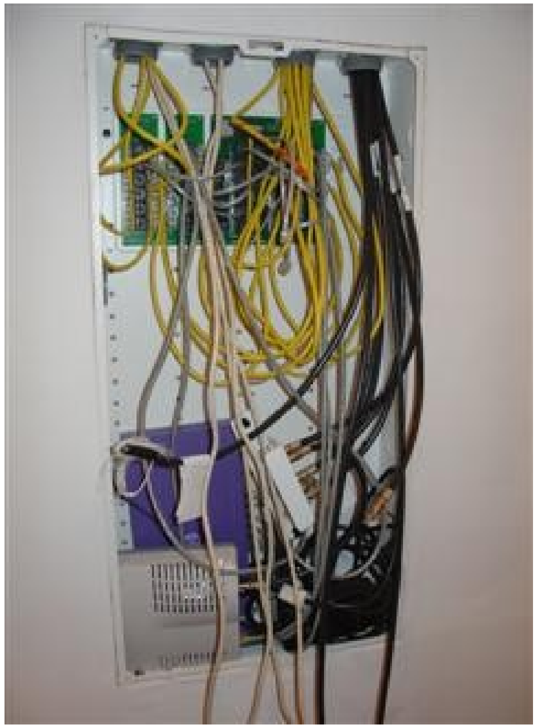

Drinking in Context
Patterns, Interventions, and Partnerships
Gerry Stimson • Marcus Grant • Marie Choquet • Preston Garrison

Network Devices

Basic structure of a PCI (or whatever) net device



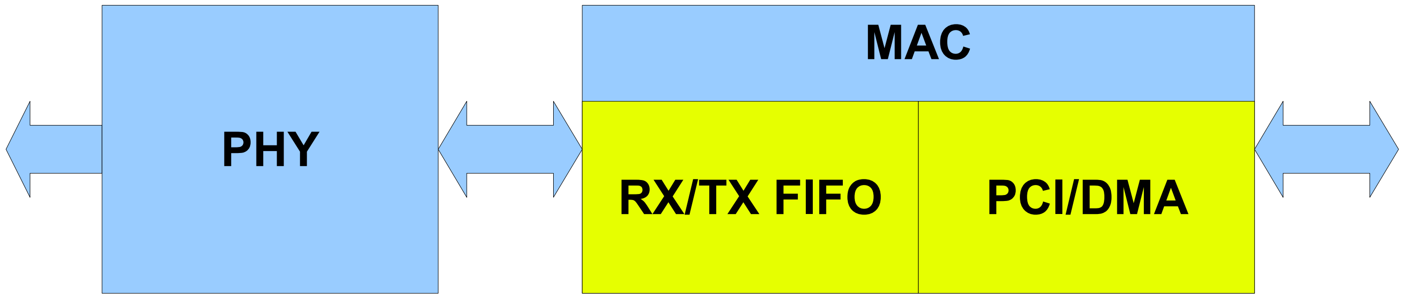
- PHY is the physical layer interface to MAC
- MAC is the guts: interface to DMA, bus, and PHY

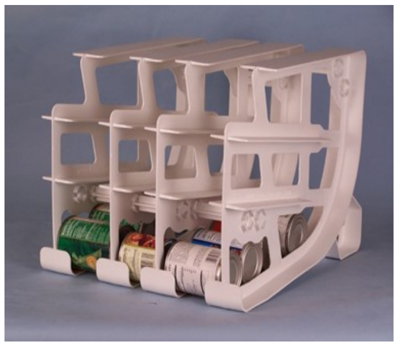


When FIFO is too small

- RX/TX checksum offloading
  - Cool idea! Let the MAC do the math
- Jumbo grams (MTU ~9000 bytes)
  - Cool idea! More bandwidth, less interrupts
- But if your TX FIFO is 8192 bytes...
  - You get Nat Semi 83820/83821

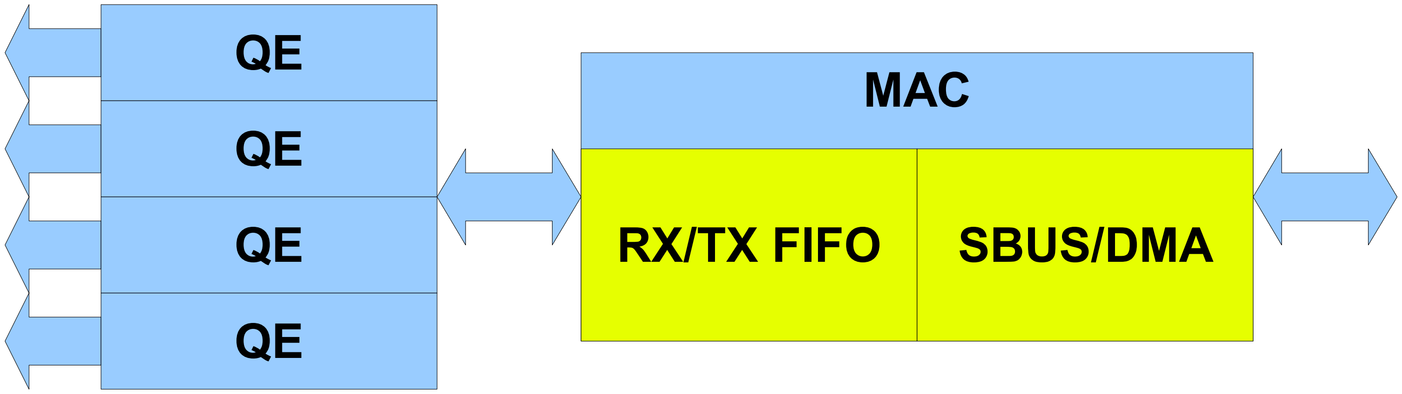
sys/dev/pci/if_nge.c





Bus bugs...

Locking up the bus (SBus) is bad...



sys/dev/sbus/qe.c

Bus Bandwidth

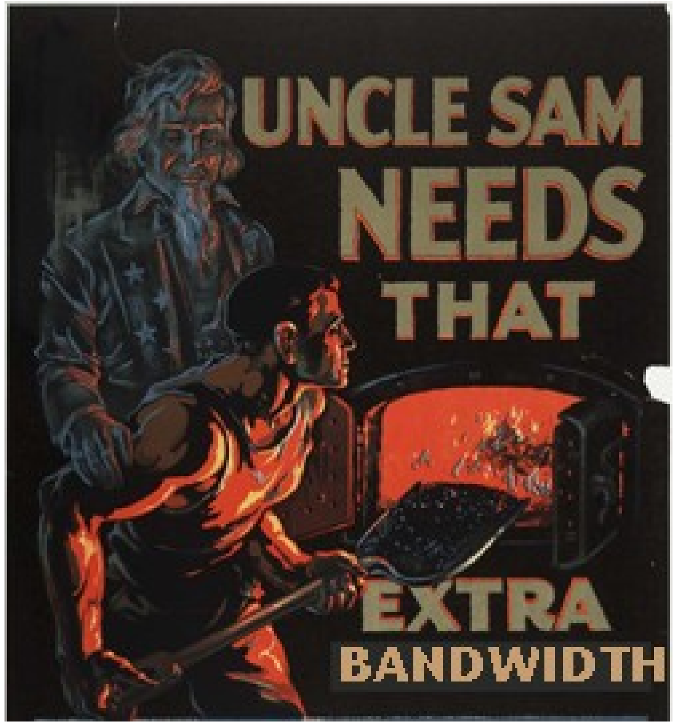
- Simple equation:

  Bit speed * bits = bandwidth

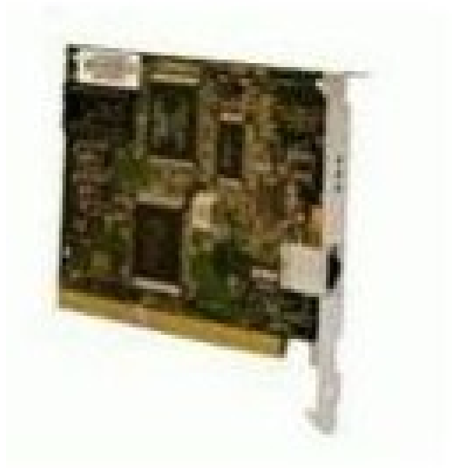
$8\,Mhz \cdot 16\,bit = 128\,Mbit/sec$ ISA

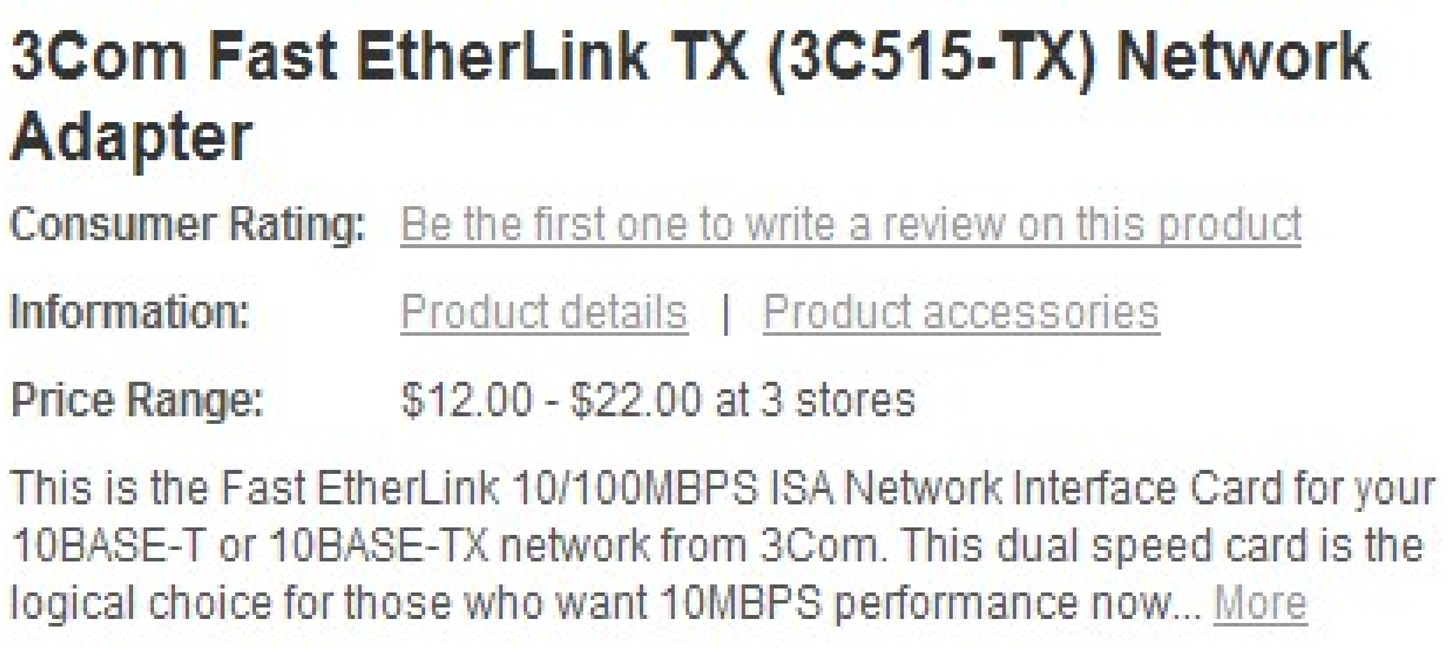
$33\,Mhz \cdot 32\,bit \approx 1.0\,Gbit/sec$ PCI

- But wait, that's maximum theoretical BW



100Mbit/sec on a 128Mbit/sec bus





3Com Fast EtherLink TX (3C515-TX) Network Adapter

Consumer Rating:	Be the first one to write a review on this product
Information:	Product details | Product accessories
Price Range:	$12.00 - $22.00 at 3 stores

This is the Fast EtherLink 10/100MBPS ISA Network Interface Card for your 10BASE-T or 10BASE-TX network from 3Com. This dual speed card is the logical choice for those who want 10MBPS performance now... More

sys/dev/isa/if_ef_isapnp.c

Mmm, pizza...

DMA, or how many bits is enough?

- Suppose you had a 32bit address space
  (ie. Modern CPU)
- But, you need to support a 24bit card
  (address space is 24bits, that is)

The answer to the question: all of them

Bounce(y) Buffers

- Allocate chunks below the boundary (16MB)
- Copy data from high memory into the chunks
- Do the operation
- Copy it back into high memory
- Return chunk to the pool

- ISA/EISA

Virtual DMA

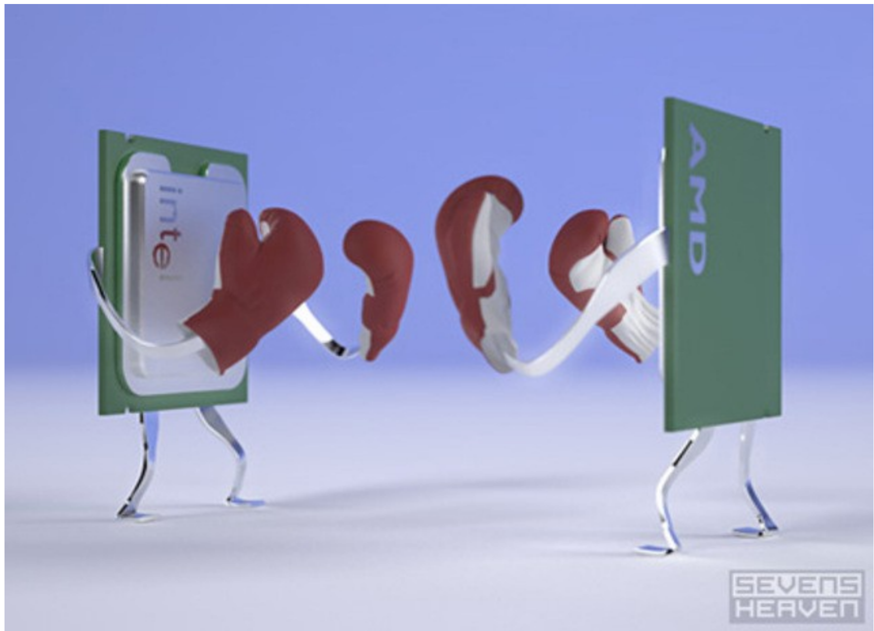
- Requires extra hardware (and code to use it)
- Cards are given DMA virtual addresses (DVAs)
- Hardware remaps DVA to the *real* address
- But, there's a limited amount of address space

- SPARC/SPARC64: *iommu*

sys/arch/sparc{64}/dev/iommu.c

Intel vs. AMD (EM64T vs AMD64)

- AMD has virtual DMA hardware (GART)
  - Not enforced usage, like sparc
- Intel, welcome to the 1990s! Bounce buffers are back in style
  - 32bit PCI devices in a 64bit machine



Write after Free...

- Suppose you free'd an address, but the device wasn't done...
- CMD Tech (pciide.c)

Questions? Comments?