KAOS
Compilers
Hardware
Architectures
Operating
Systems

Aggregate.Org
UNBRIDLED COMPUTING

CPE200
Oct. 27, 2021

Henry (Hank) Dietz
Professor and Hardymon Chair,
Electrical & Computer Engineering

University of Kentucky

What Do We Do?

- KAOS: Compilers, Hardware Architectures, and Operating Systems
- Aggregate.Org (UNBRIDLED COMPUTING):
  - Aggregate Function Networks (AFNs)… in 1st Linux PC Cluster Supercomputer
  - Make the components of a computing system work better together, improving performance & gaining new abilities

University of Kentucky

You are what you eat...

- I am a computer engineer / systems guy
- Background in CS, EE, ME, Math, & Econ
- From 1983, parallel processing researcher
- In 1970s, photo editor of various school pubs and a published professional photographer
- From birth, trained to know how everything in Dad's manufacturing company worked

University of Kentucky

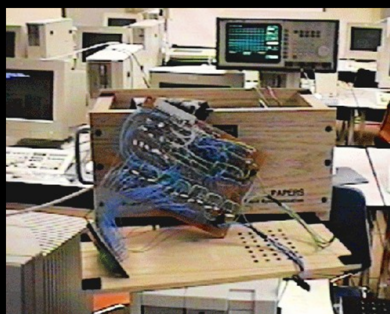
Supercomputers

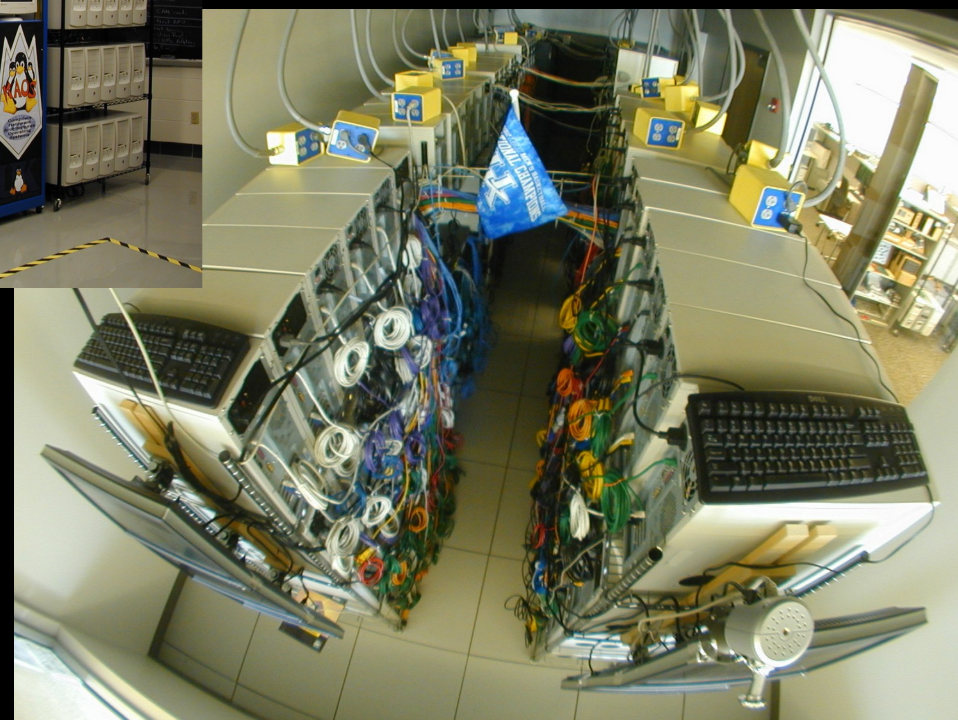
Computers that can solve big problems
and can scale to solve bigger problems.

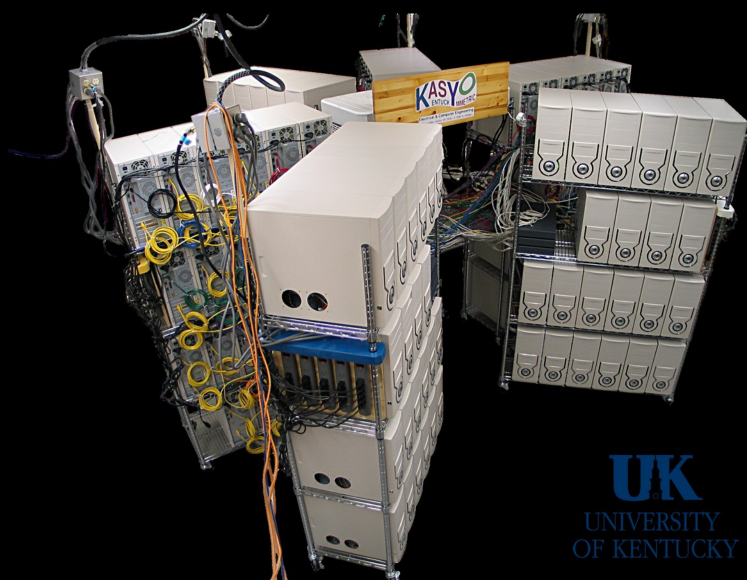
- Mostly about parallel processing
- Need not be huge, expensive, etc.
- We make them cheap
- We also make them able to do new things

University of Kentucky

(Old) Cheap Supercomputers







UK
UNIVERSITY
OF KENTUCKY

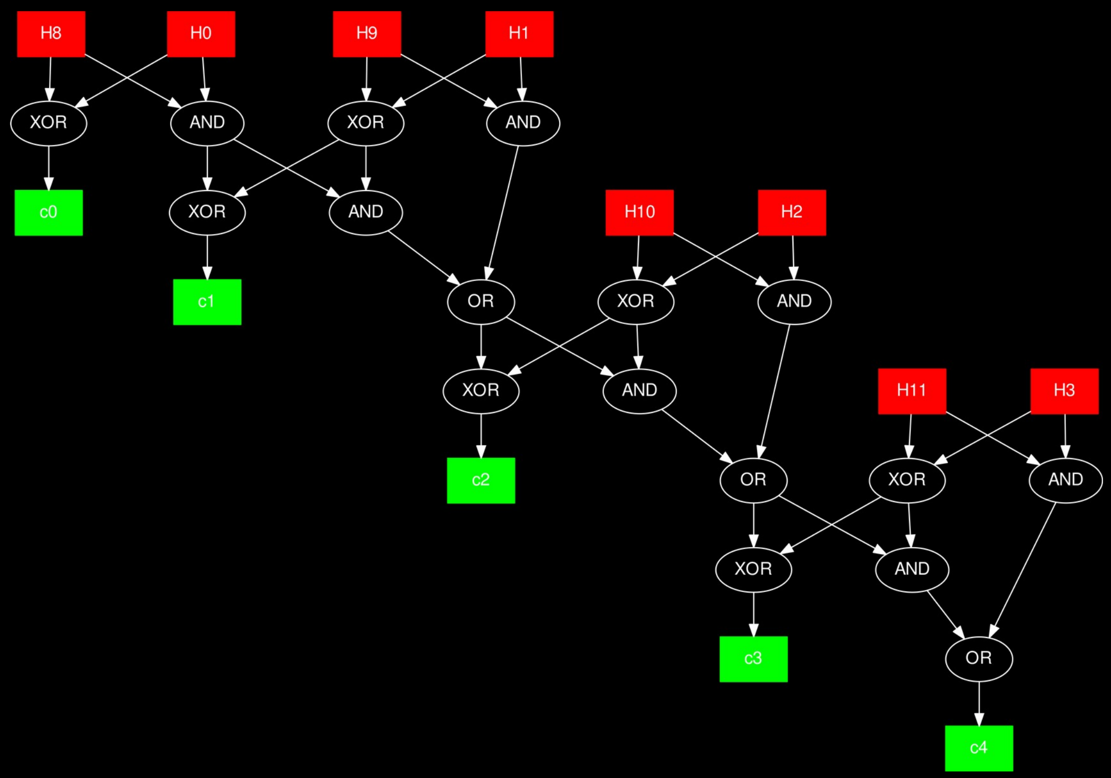
University of
Kentucky

Current Supercomputing Research

- I'm one of the folks who started the cluster supercomputing revolution...
- Several years ago, I realized:
  - My lab has 168kW power, 30 tons cooling
  - My lab heats half the Marksbury building
  - My lab could **not** power **1 high-end rack**!
  - Big systems have **thousands of racks**
- It's really all about power / computation

```
int:4 a, b;
c = a + b;
```

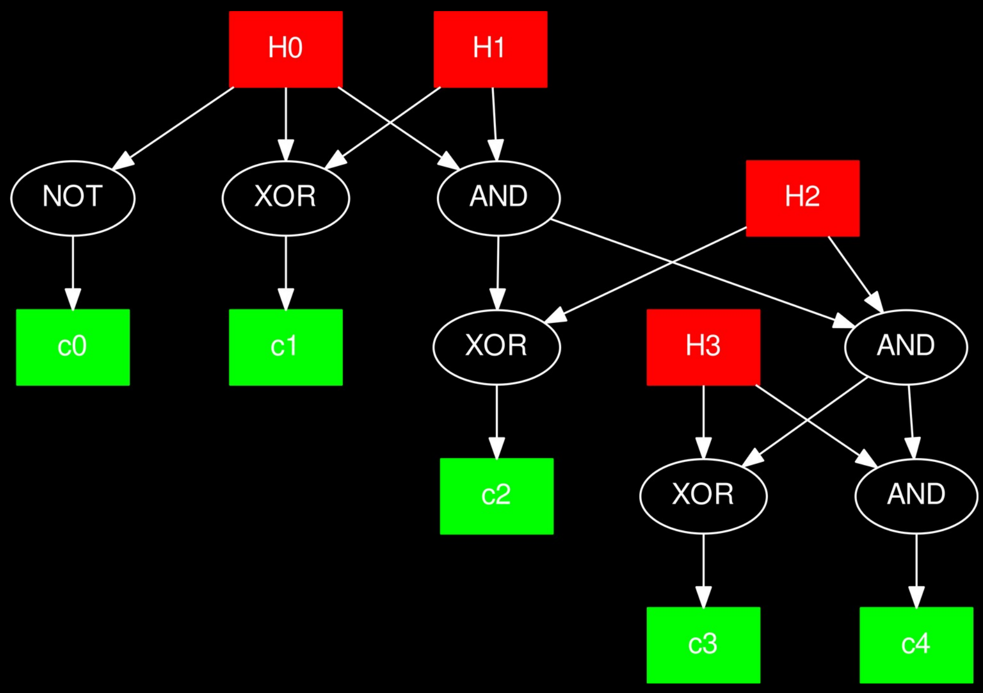
- Optimized, **17** single-gate operations:



University of Kentucky

```
int:4 a, b;
b = 1; c = a + b;
```

- Optimized, **7** single-gate operations:



University of Kentucky

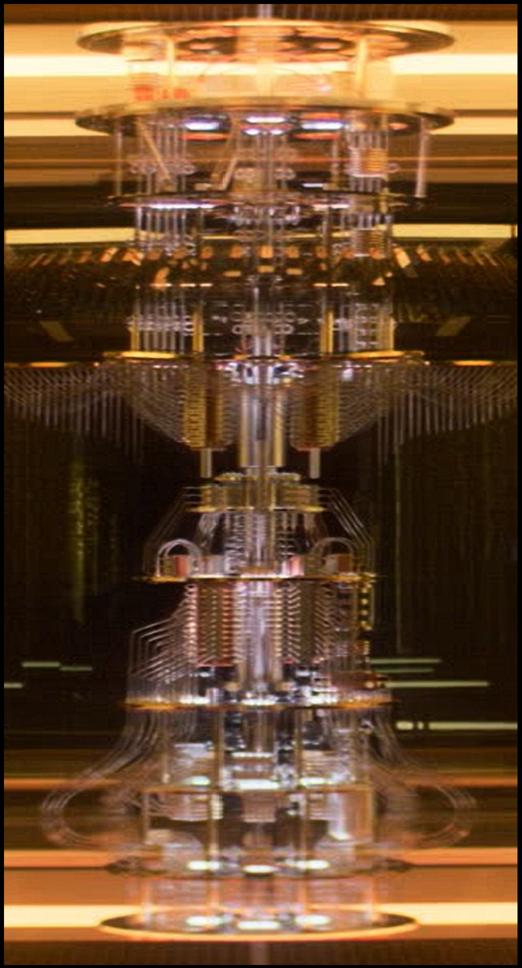
```
int:8 a, b, c;
a = (c * c) ^ 70;
a = ((a >> 1) & 1);
a = b + (c * b) + a;
a = a + ~(b * (c + 1));
```

University of Kentucky

```
int:8 a, b, c;
a = (c * c) ^ 70;
a = ((a >> 1) & 1);
a = b + (c * b) + a;
a = a + ~(b * (c + 1));
```

- About **206,669 gates** unoptimized
- Optimized, it's just `a = 0;`

Quantum Computers?



University of Kentucky

Quantum Computing

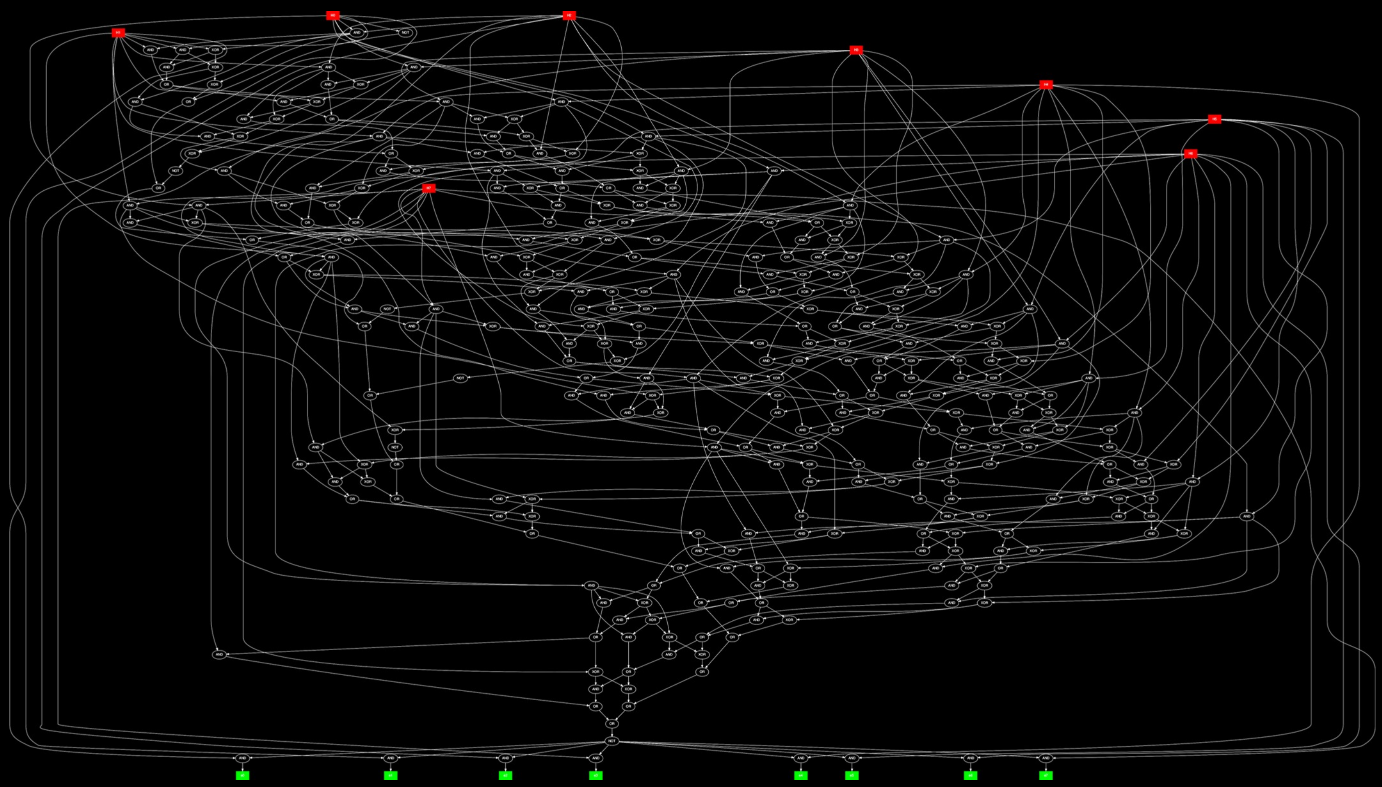
Parallel processing ***without*** parallel hardware.

- **Qubits** instead of bits
  - Each qubit can be 0, 1, or ***superposed***
  - A “gate” operates on superposed values
  - ***Entangled*** qubits maintain values together
  - Measuring a qubit’s value picks 0 or 1
- Quantum computers are **not** the only way to do that: Parallel Bit Pattern computing

University of Kentucky

An Example: Find `sqrt(29929)`

- **310** single-gate operations:



An Example: Find `sqrt(29929)`

- The complete C program, prints `0 173`

```
int main(int argc, char **argv) {
  pbit_init();
  pint a = pint_mk(16, 29929);
  pint b = pint_h(8, 0xff);
  pint c = pint_mul(b, b);
  pint d = pint_eq(c, a);
  pint e = pint_mul(d, b);
  pint_measure(e);
}
```

University of Kentucky

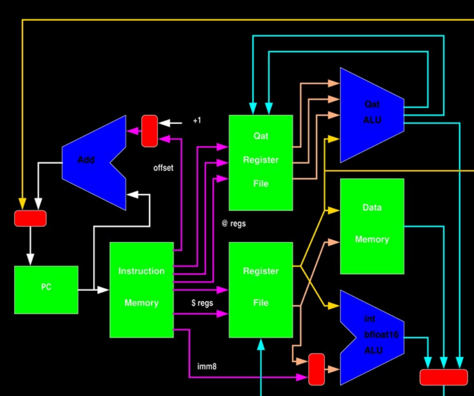
An Example: factor 15

```
had @0,3
had @1,5
and @2,@0,@1
had @3,4
and @4,@0,@3
had @5,2
and @6,@5,@1
and @7,@4,@6
and @8,@5,@3
had @9,1
and @10,@9,@1
and @11,@8,@10
and @12,@9,@3
had @13,0
and @14,@13,@1
and @15,@12,@14
xor @16,@8,@10
and @17,@15,@16
or  @18,@11,@17
xor @19,@4,@6
and @20,@18,@19
or  @21,@7,@20
and @22,@2,@21
had @23,6
and @24,@0,@23
and @25,@22,@24
xor @26,@2,@21
and @27,@5,@23
and @28,@26,@27
xor @29,@18,@19
and @30,@9,@23
and @31,@29,@30
xor @32,@15,@16
and @33,@13,@23
and @34,@32,@33
xor @35,@29,@30
and @36,@34,@35
or  @37,@31,@36
xor @38,@26,@27
and @39,@37,@38
or  @40,@28,@39
xor @41,@22,@24
and @42,@40,@41
or  @43,@25,@42
had @44,7
and @45,@0,@44
and @46,@43,@45
xor @47,@40,@41
and @48,@5,@44
and @49,@47,@48
xor @50,@37,@38
and @51,@9,@44
and @52,@50,@51
xor @53,@34,@35
and @54,@13,@44
and @55,@53,@54
xor @56,@50,@51
and @57,@55,@56
or  @58,@52,@57
xor @59,@47,@48
and  @60,@58,@59
or   @61,@49,@60
xor  @62,@43,@45
and  @63,@61,@62
or   @64,@46,@63
xor  @65,@61,@62
xor  @66,@58,@59
xor  @67,@55,@56
xor  @68,@53,@54
xor  @69,@32,@33
and  @70,@13,@3
xor  @71,@12,@14
and  @72,@70,@71
and  @73,@69,@72
and  @74,@68,@73
or   @75,@74,@74
not  @75
or   @76,@67,@75
or   @77,@66,@76
or   @78,@65,@77
or   @79,@64,@78
or   @80,@79,@79
not  @80
lex  $0,31
next $0,@80
copy $1,$0
next $1,@80
lex  $2,15
and  $0,$2 ;5
and  $1,$2 ;3
```

Figure 10: Code prime factoring 15 (3 columns).

```
pint a = pint_mk(4, 15);  // a=15
pint b = pint_h(4, 0x0f); // b=0..15
pint c = pint_h(4, 0xf0); // c=0..15
pint d = pint_mul(b, c);  // d=b*c
pint e = pint_eq(d, a);   // e=(d==a)
pint f = pint_mul(e, b);  // make non-factors 0
pint_measure(f);          // prints 0, 1, 3, 5, 15
```

Figure 9: Word-level prime factoring of 15.

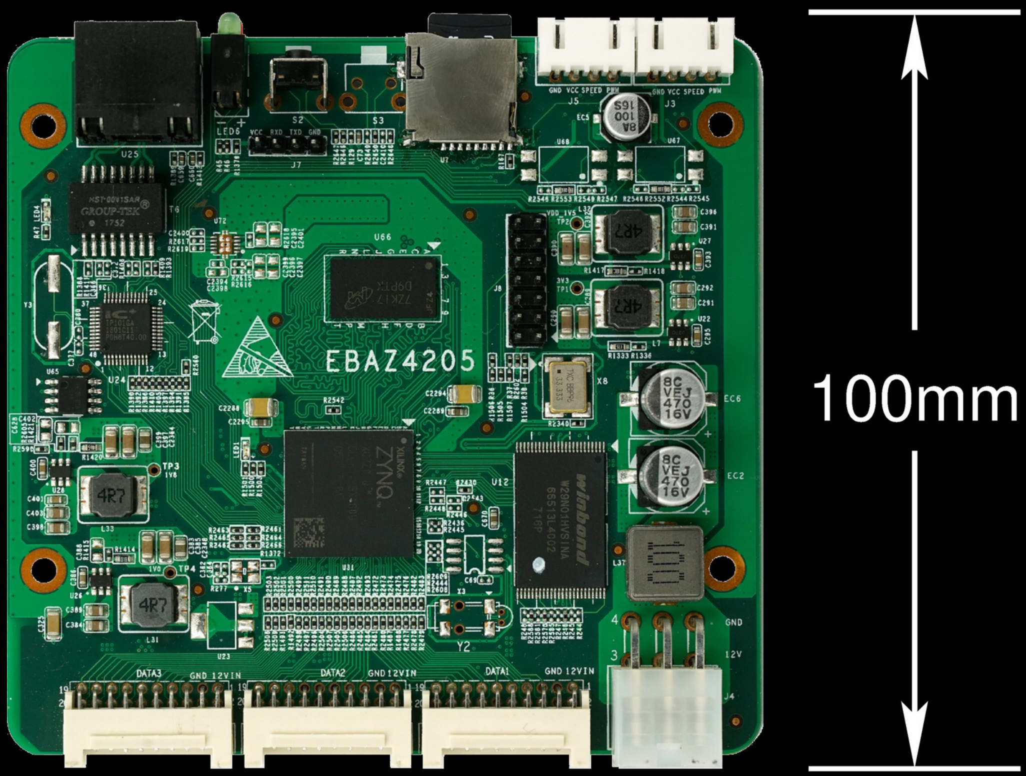


Tangled

(no, not that one)

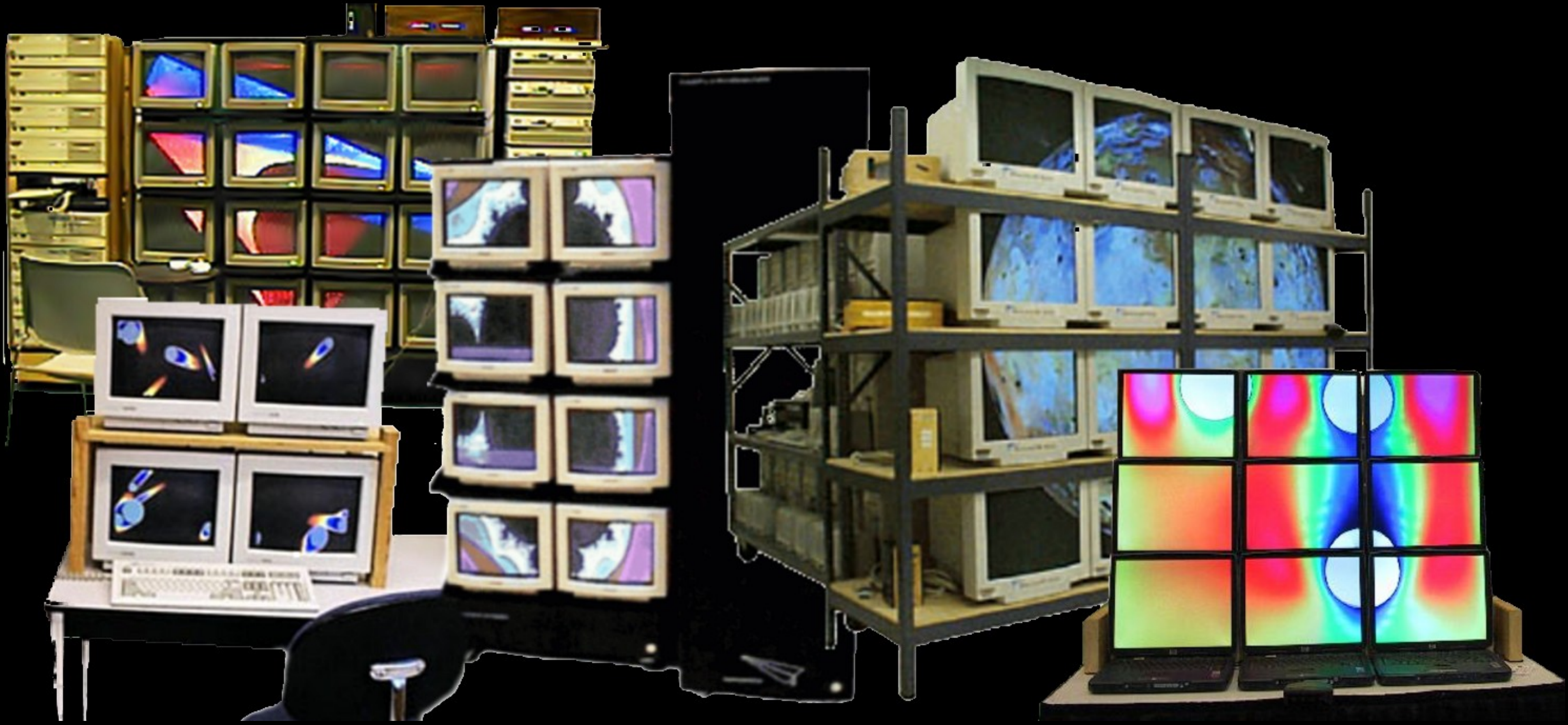
University of Kentucky

Our next superccomputer...



University of Kentucky

Supercomputers Doing New Things



University of Kentucky

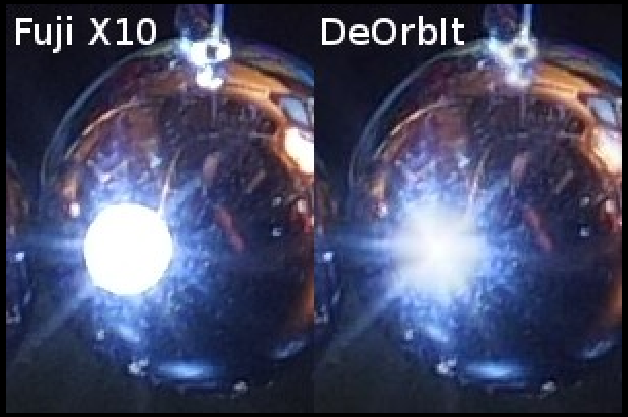
Computational Photography

Cameras as computing systems;
using computation to enhance camera abilities
and / or to process the data captured

- New camera / sensor / processing models
- Intelligent computer control of capture
- Detection / manipulation of image properties

University of Kentucky

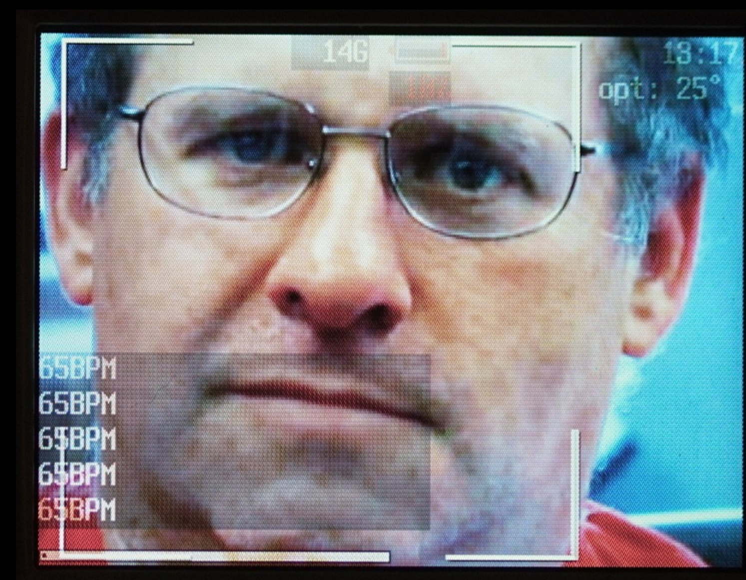
“Raw” Repair



- “Raw” means “uncooked” or “unprocessed”
- Can *credibly* **repair** corrupted data
  - Fuji X10 “white orbs” blooming ⇒ **DeOrbIt**
  - Sony ARW compression artifacts ⇒ **KARWY**
  - Sony ARW PDAF artifacts ⇒ **KARWY-SR**

University of Kentucky

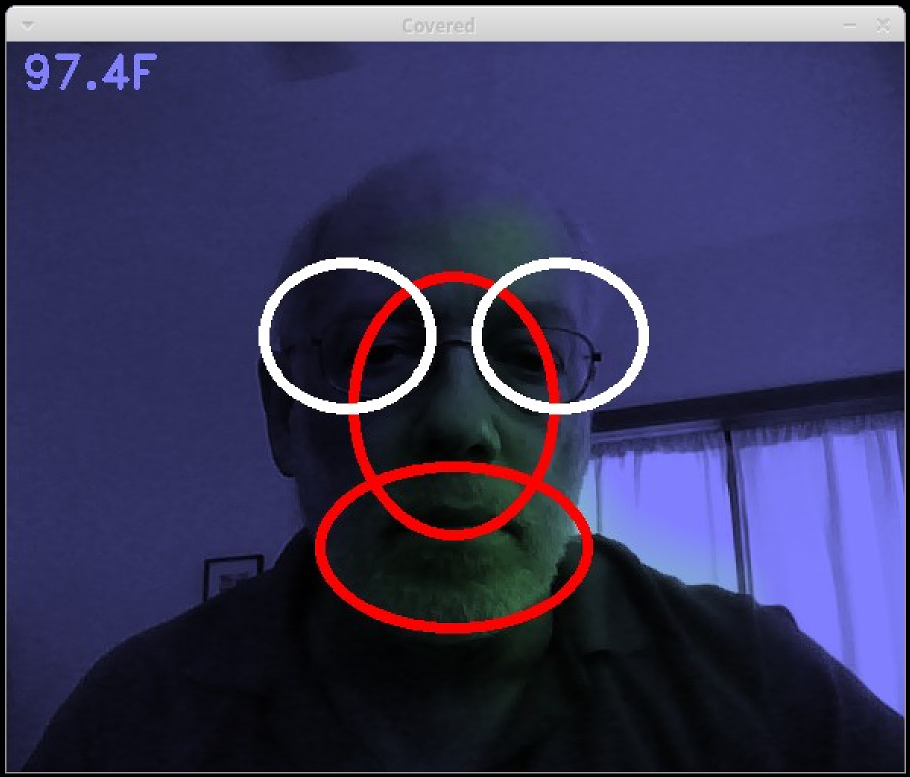
Photoplethysmography



- Reprogrammed a $100 camera to detect heartbeats by detecting color change

University of Kentucky

Covered Safe Entry Scanner



- Detect when a mask is being properly worn
- Also thermal imager & contact tracing

TDCI: Time Domain Continuous Imaging

- TDCI representation: a continuous waveform per pixel, compressed (mostly) in time domain
- TDCI processing enables:
  - High dynamic range (HDR), improved SNR
  - Rendering a virtual exposure for any time interval (start time, shutter speed)
  - Rendering a conventional video at any FPS and shutter angle (temporal weighting)

University of Kentucky

TDCI Example

- Video is converted to TDCI, then new frames synthesized... with significantly better quality

University of Kentucky

FourSee TDCI Camera

- Syncs four reprogrammed PowerShots
- 3D-printed structure for alignment, etc.

University of Kentucky

A Custom 3D-Printed Adapter With M42-Compatible Thread

Lens adapter
M42 x 1mm pitch
to Sony E

On $180 printer,
0.25mm layers!

University of Kentucky

Some of our latest work...

University of Kentucky

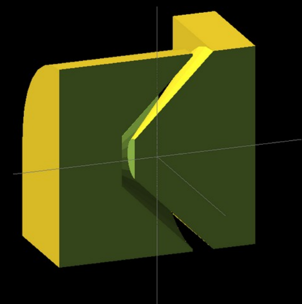
Design For Manufacturability (DFM)

Design product so that it is easy to manufacture.

- E.g., Lego doesn’t easily do curves… and most 3D printers don’t do unsupported spans
- How is this computer engineering?
  - A design is a parametric program (parameterized by machine characteristics)
  - Compiler technology optimizes for DFM

University of Kentucky

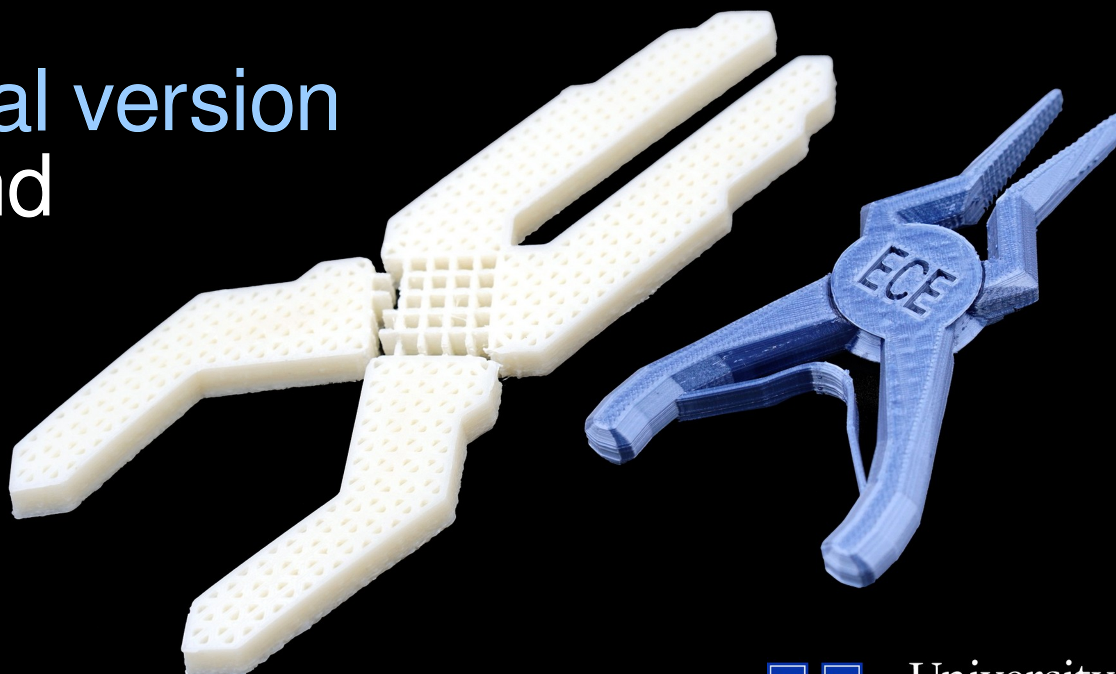
3D-Printed Spanless Hinges



University of Kentucky

A Fancier DFM Example

- In 2016, researchers at the *Hasso Plattner Institute* made "metamaterial pliers": a single part with stiffness, spring, & bending hinge
- Our metamaterial version has a **spring** and a **spanless hinge** and it works…



University of Kentucky

You Can Get Involved

- Talk to me, or Paul Eberhart, etc.
- Most stuff is posted at `AGGREGATE.ORG`
- **Q**uantum computing **E**ducation & **R**esearch In **K**entucky