Aggregate.Org

UNBRIDLED COMPUTING

CPE200, 253FPAT
Noon, March 8, 2023

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

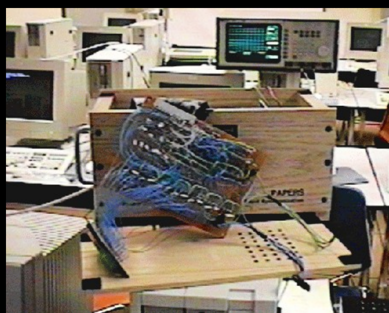
University of Kentucky

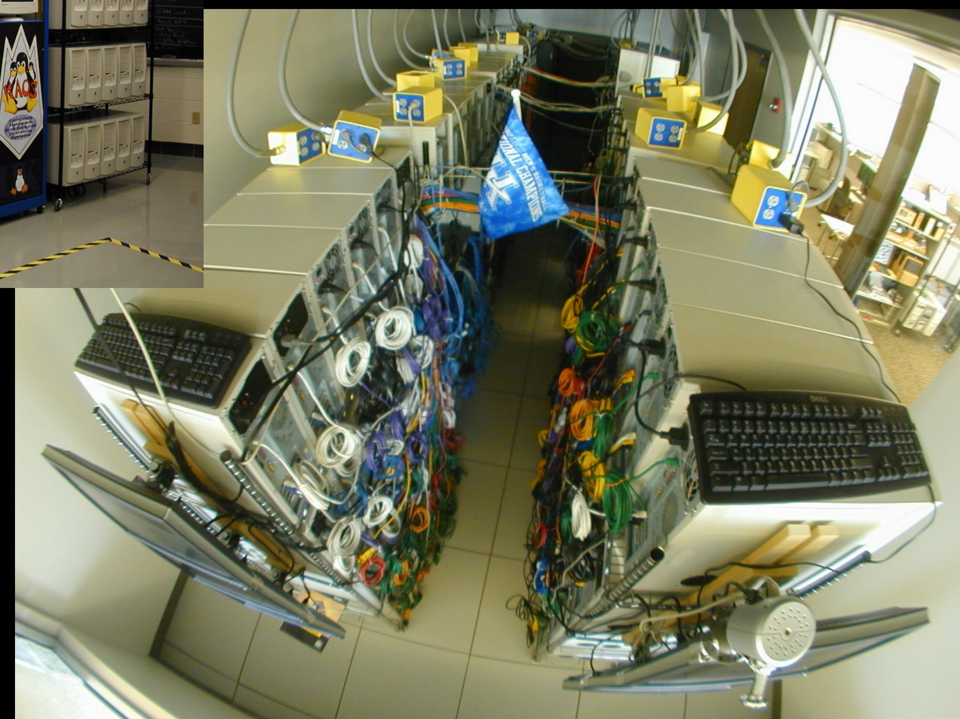
Supercomputers

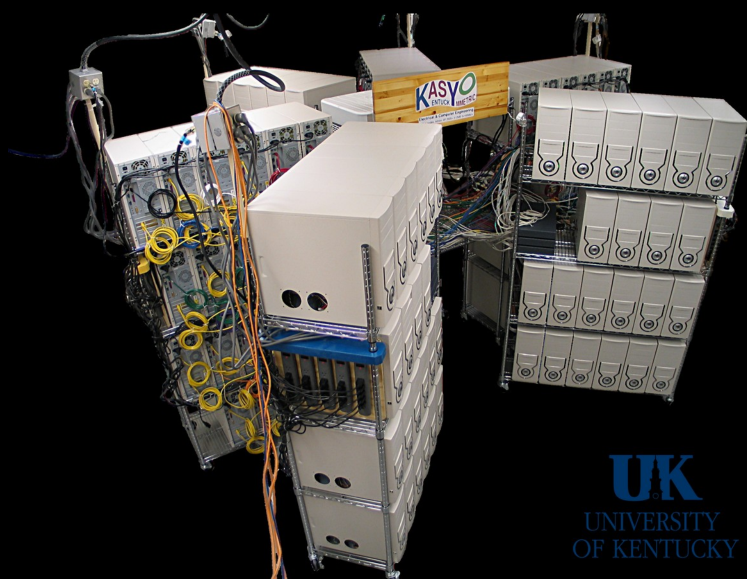
Computers that can solve big problems
and can scale to solve bigger problems.

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

(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

University of Kentucky

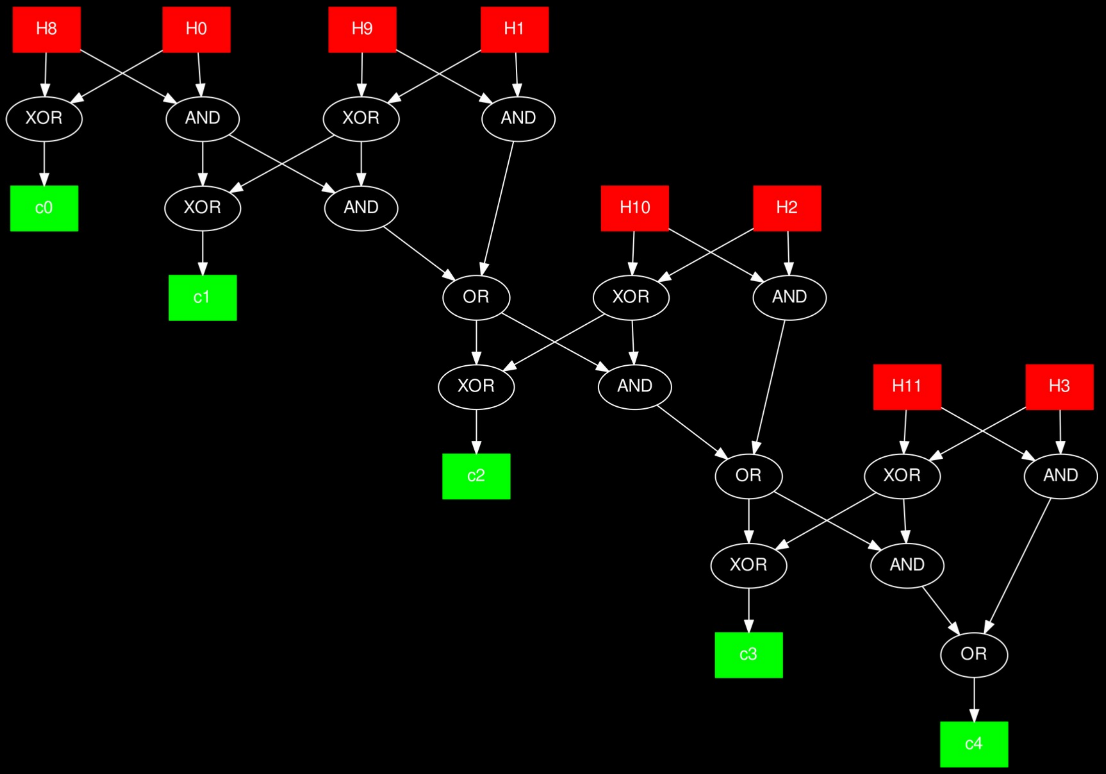
```
int a, b, c;
c = a + b;
```

- 32-bit carry lookahead adder…

  roughly **650** single-gate operations!

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

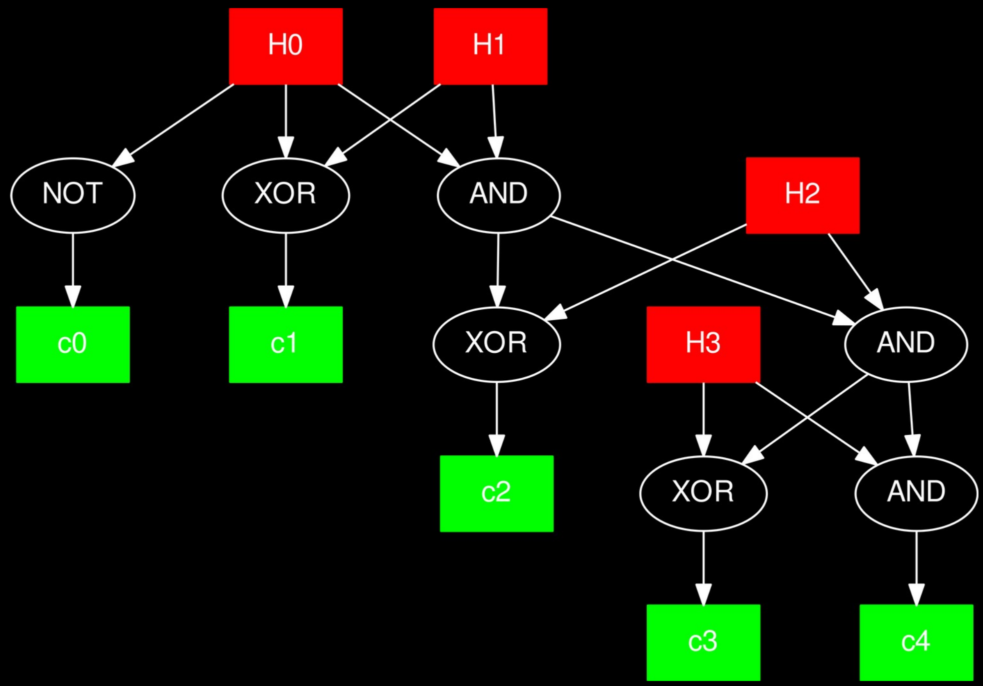
- Optimized, **17** single-gate operations:



University of Kentucky

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

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



University of Kentucky

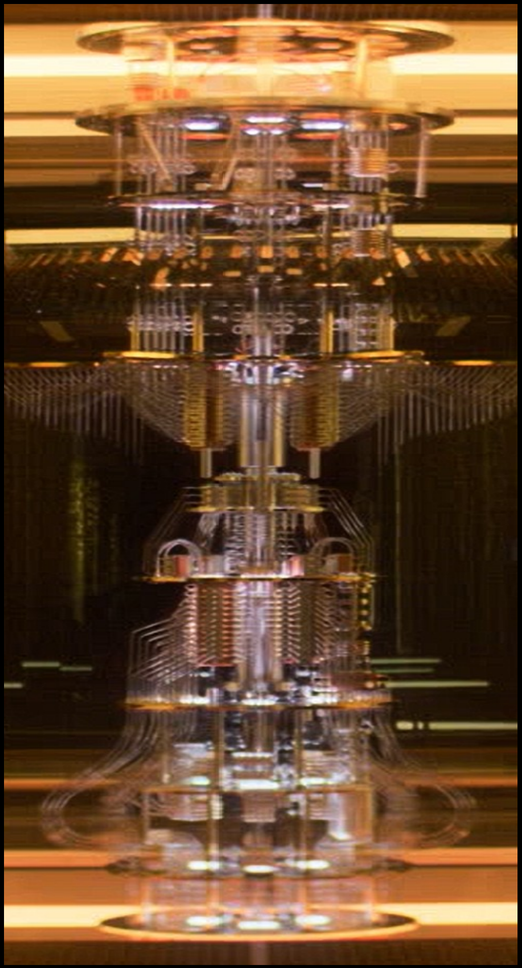
```
int:8 a, b, c;
a = (c * c) ^ 70;
a = ((a >> 1) & 1);
a = b + (c * b) + a;
a = a + ~(b * (c + 1));
```

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

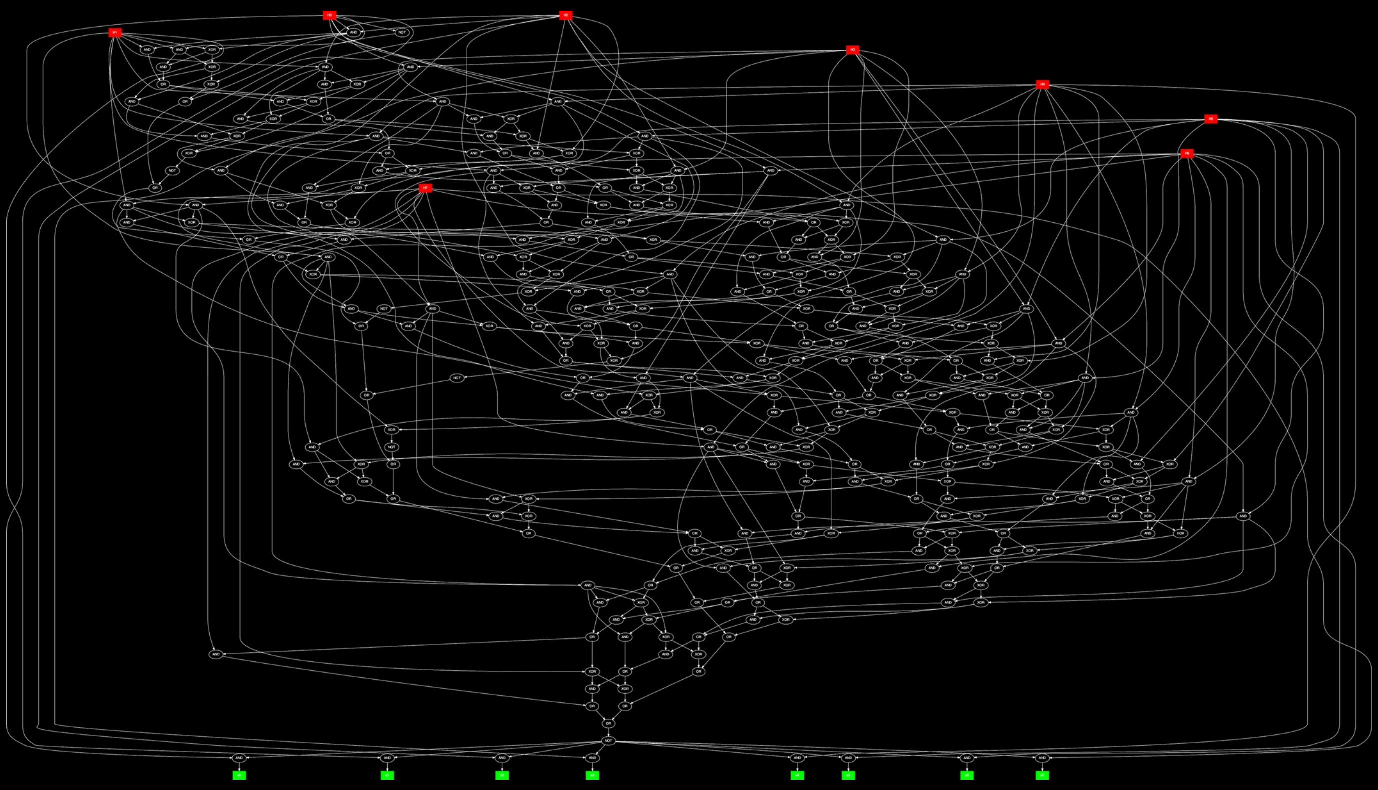
Quantum Computing

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

An Example: Find `sqrt(29929)`

- **310** single-gate operations:



University of Kentucky

An Example: Find `sqrt(29929)`

- C++ `isqrt(29929)` returns `173`

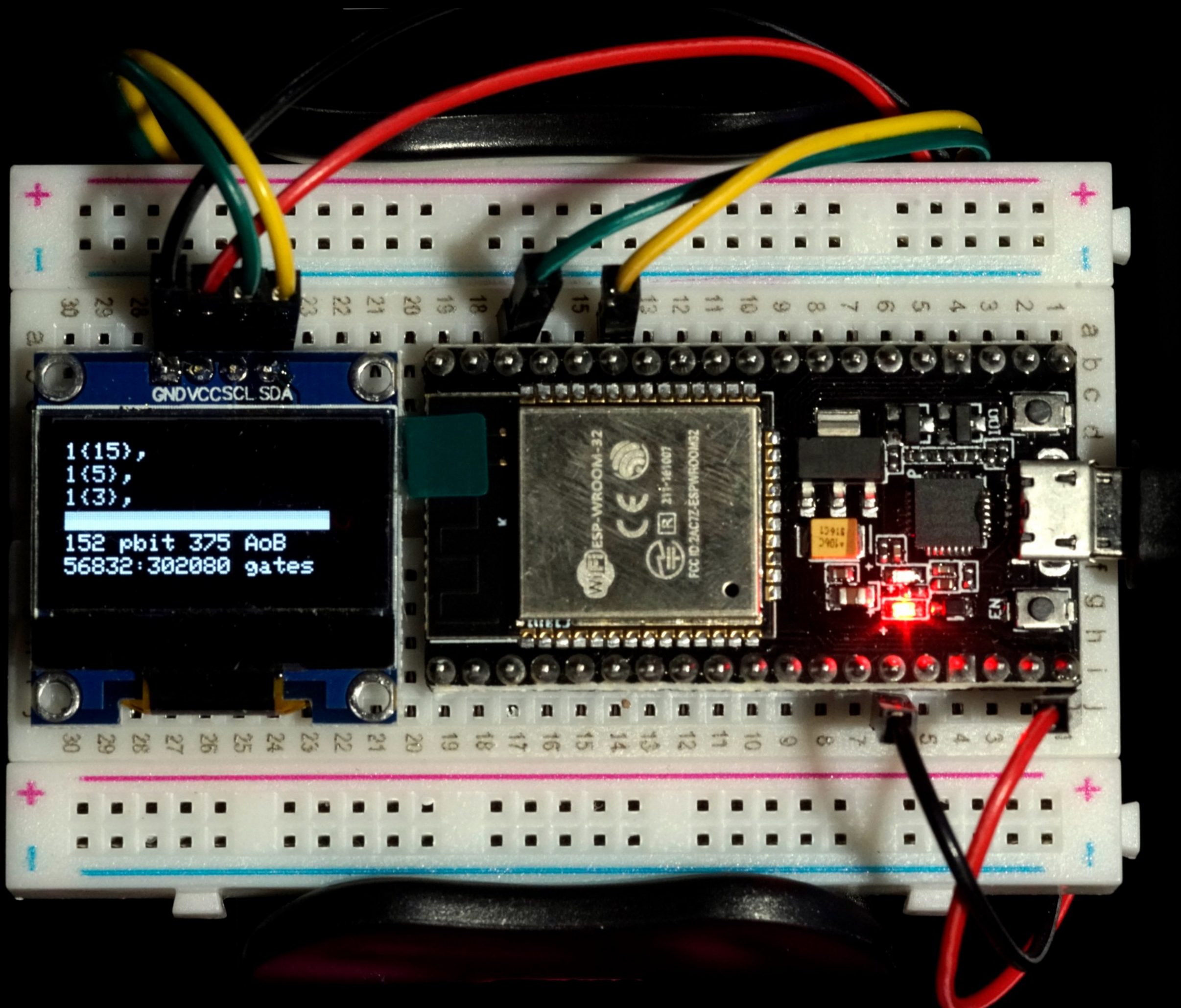
```
int isqrt(int val) {
  pint a(val);
  pint b = pint(0).Had(8);
  pint c = (b * b);
  pint d = (c == a);
  pint pos = d.First();
  return(pos);
}
```

An Example: Prime Factoring

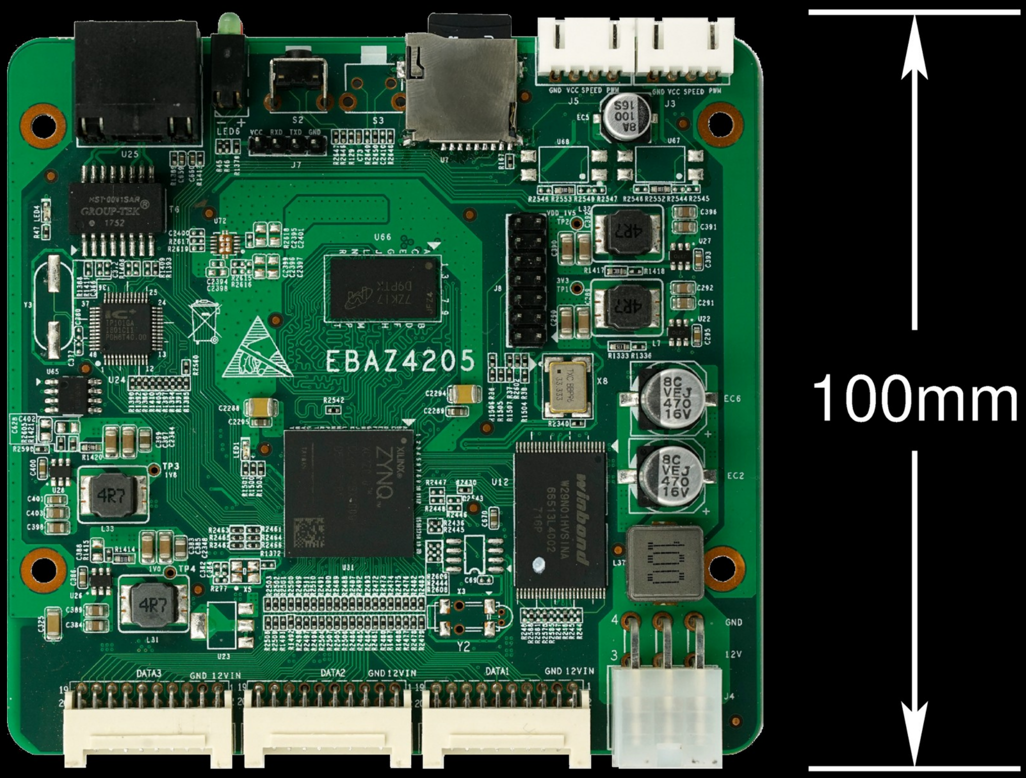
```
// Factor value a
// Primes that fit in 5 bits:
// 2,3,5,7,11,13,17,19,23,29,31
int a = 11*29;
pint b = 0;
b = b.Had(5,0); // 1st factor
pint c = 0;
c = c.Had(5,5); // 2nd factor
pint d = b * c; // multiply 'em
pint e = (d == a); // which gave a?
pint f = e * b;
Print(f);
```

University of Kentucky

An Example: factor 319

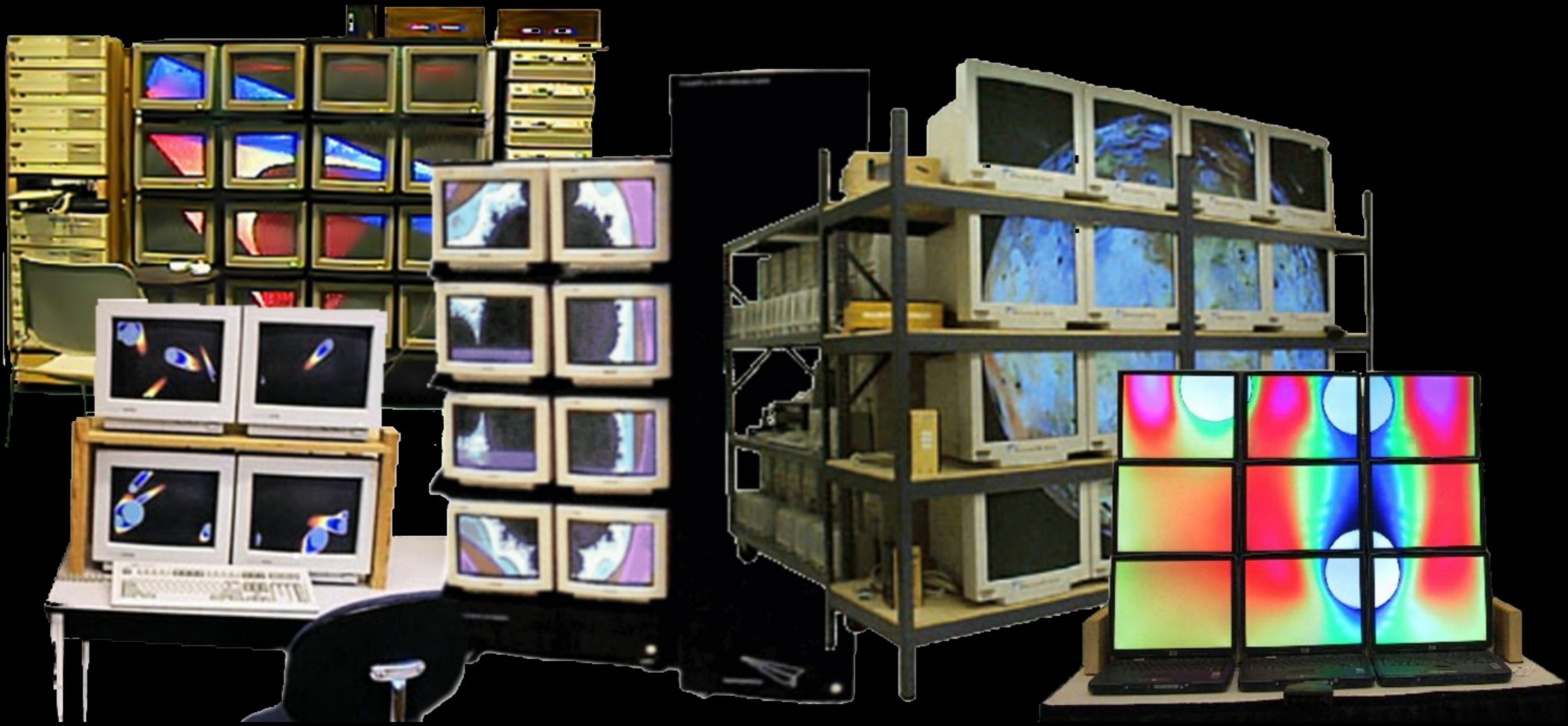


Our next superccomputer...



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

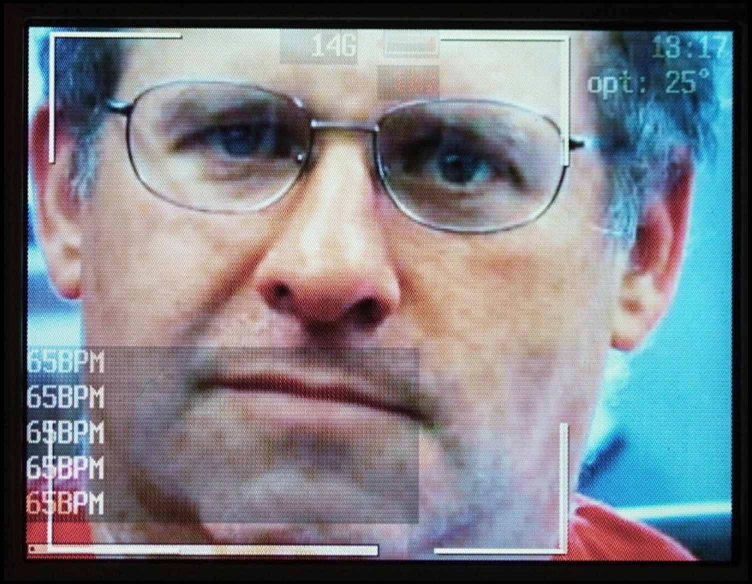
University of Kentucky

"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

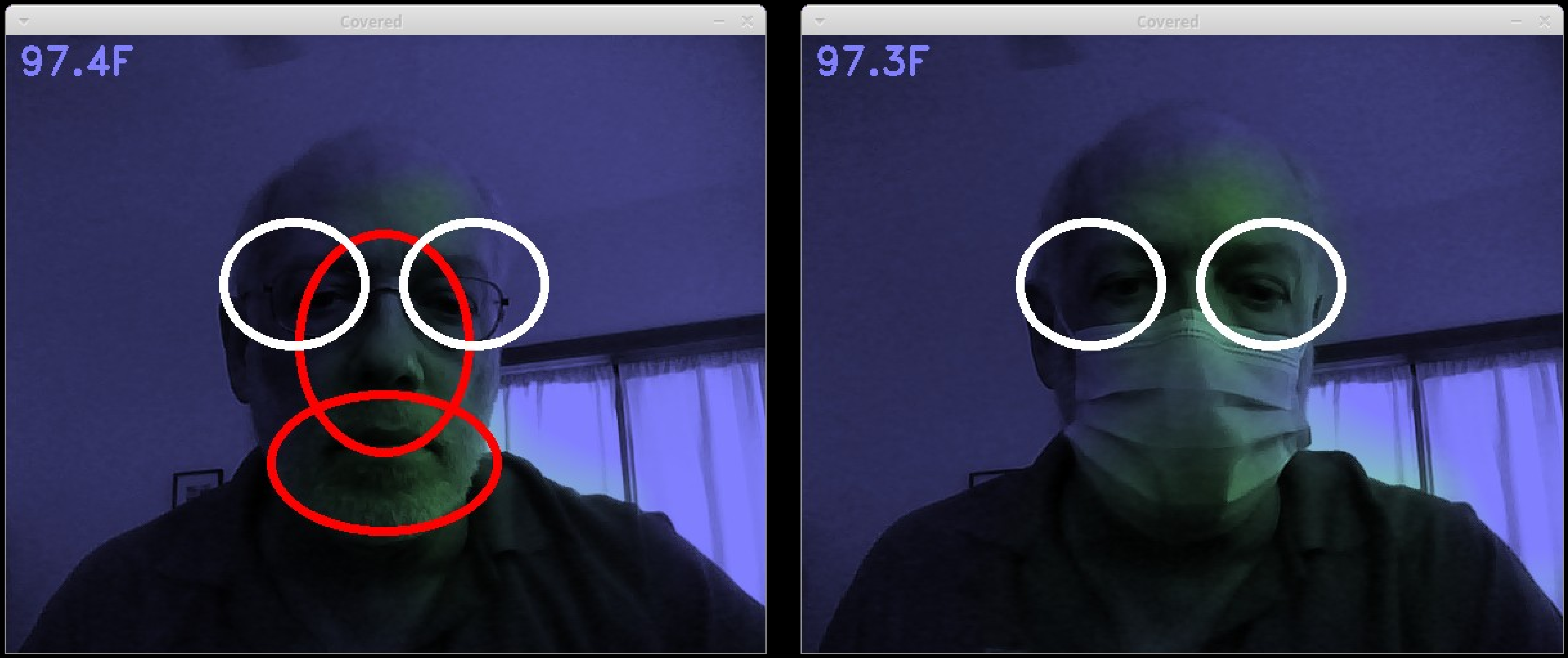
Photoplethysmography



- 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

University of Kentucky

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

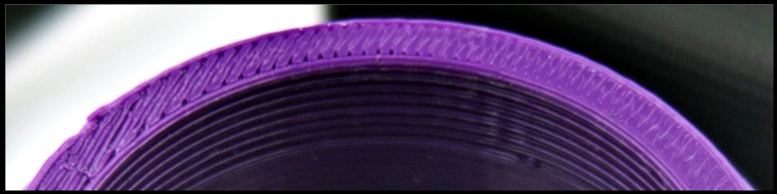
- Video is converted to TDCI, then new frames synthesized... with significantly better quality

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

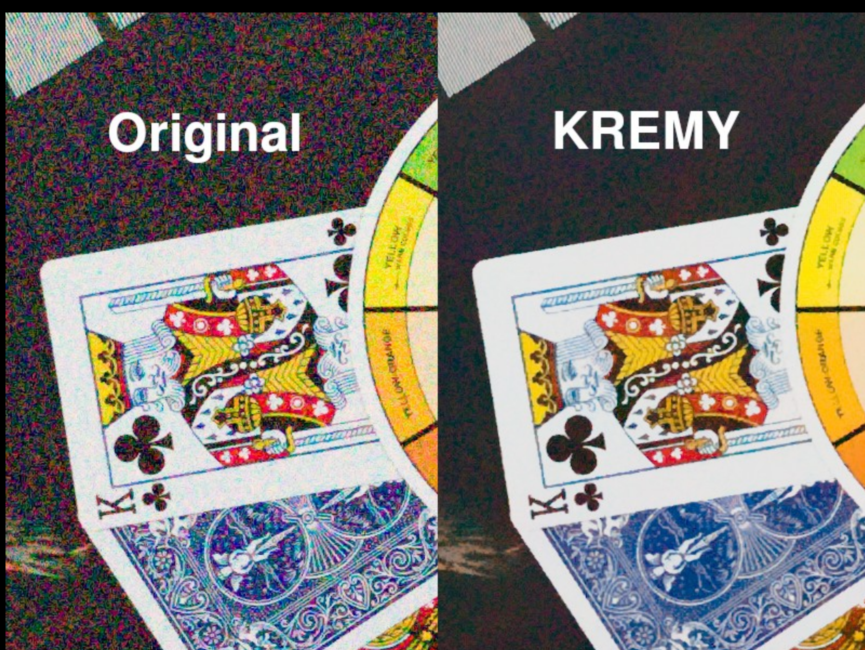


Lens adapter
M42 x 1mm pitch
to Sony E

On $180 printer,
0.25mm layers!

University of Kentucky

Some of our latest work...



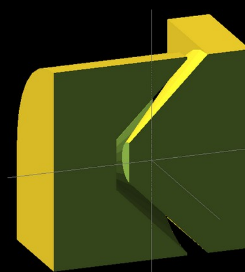
University of Kentucky

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

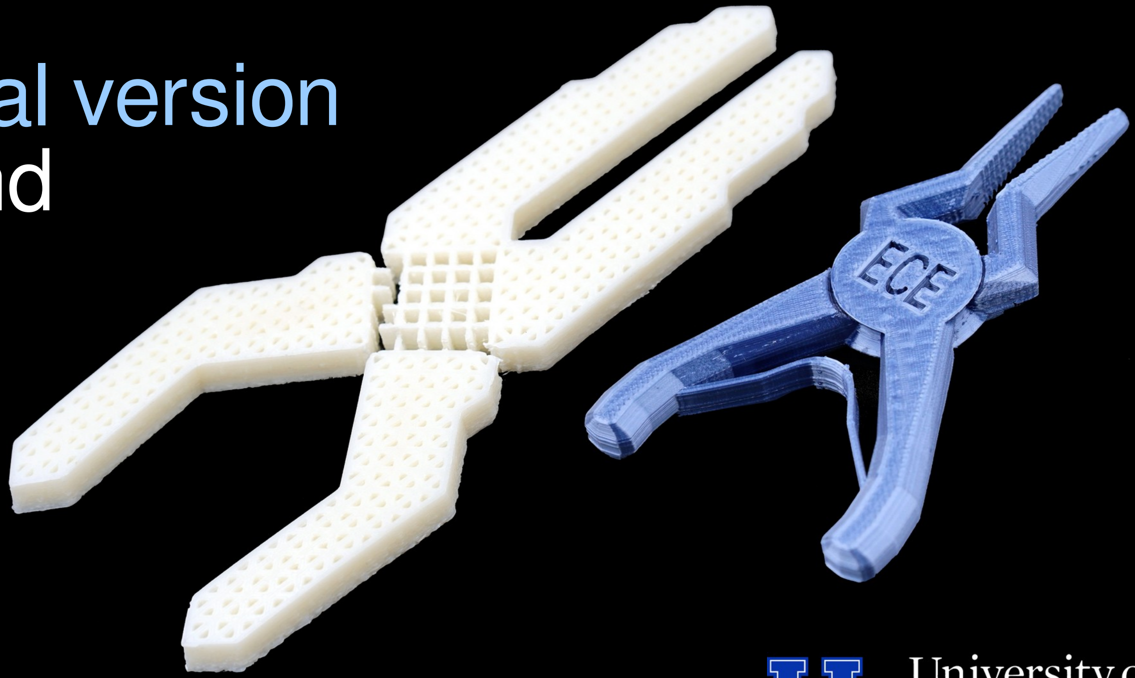
3D-Printed Spanless Hinges



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

Spring 2023: TR 12:30-1:45 EE599-001/EE699-001

Programmable Cameras and IoT

This course will start by introducing the basic principles of photography and the details of how digital cameras work. However, cameras are no longer just about photography; they are *sensors in embedded computing systems* that can serve a wide range of applications. For example, using **CHDK**, it is trivial to program a **Canon PowerShot** camera to serve as a non-contact tape measure. The course will use CHDK cameras to explain how camera internals work and students will get hands-on experience using and programming these cameras. Cameras are also now cheap sensors for use within Internet of Things (IoT) devices. An **ESP32-CAM** IoT module that costs under $10 includes a 2MP camera and can be programmed for tasks as diverse as wirelessly serving live video via an HTML browser interface to unlocking a door when a person's face is recognized. We will discuss IoT devices in general and use of the ESP32-CAM and its OV2640 camera in particular. Students will implement simple IoT projects using the ESP32-CAM via the Arduino programming environment.

Prerequisites: Familiarity with C/C++ programming. No background with photography is required.

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