X-ray driver RE and applications

John McMaster
JohnDMcMaster@gmail.com

What

- Primary goal: x-ray PCBs @ home
- Derived goal: Camera + Windowz app => Linux app
- How I approached this camera
- Current projects
- Future projects

The problem

- Previous presentation: microscope camera
- No danger factor! (snore)

The solution

- What if data collection risks radiation sickness?

Our star: x-ray camera

- CMOS camera with scintillator
- Several rebrands on the market

Camera teardown

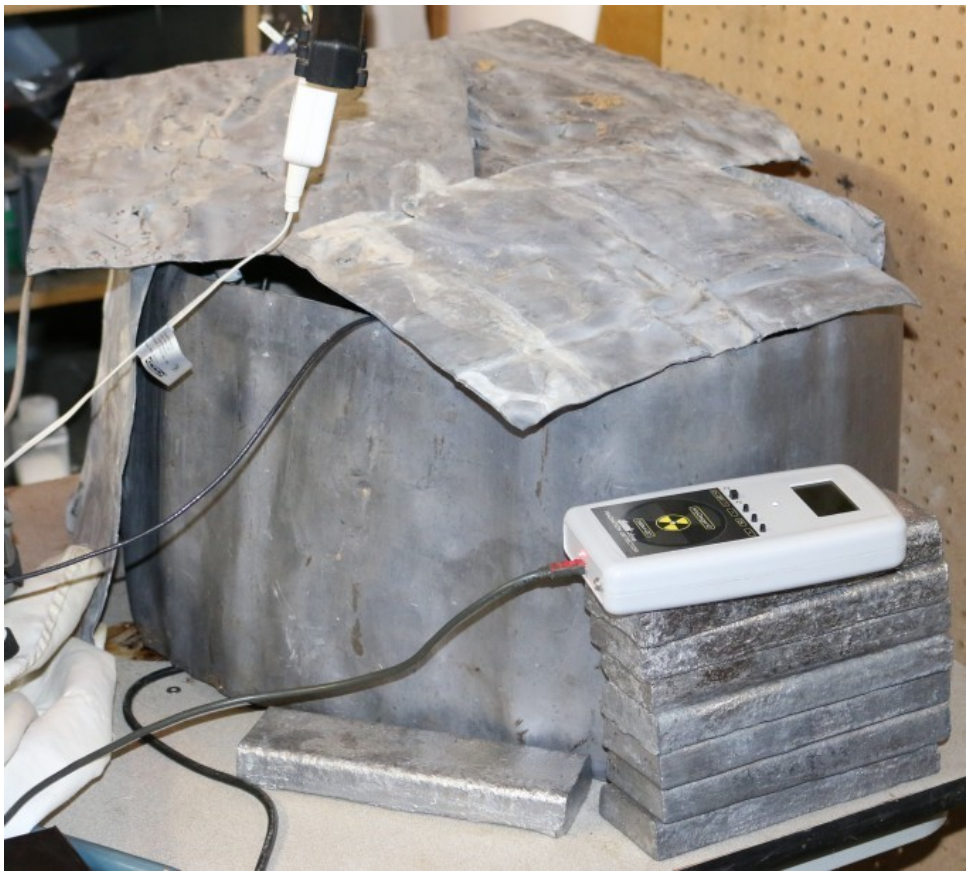
- Acquired broken units to take apart
- Cypress FX2 USB microcontroller
- Something probably an I2C EEPROM
- Actel ProASIC3 FPGA
- Spansion 128 Mb flash
- Something else

Death rays

- High power x-ray heads from the 80's

Safety

- Remote control via WPS-7
- Radiation monitoring, lead bunker
- Hard to use bunker for initial testing



RE setup

- Was the most challenging part of project
- Only snaps picture above certain dose level
- Problem: x-ray head set too low
- Setup overview
  - Laptop w/ VMWare, Wireshark
  - Power control: WPS-7
  - Filament control: variac1
  - Voltage control: variac2
  - 3x mygeiger2 radiation detectors

Wireshark: capture packets

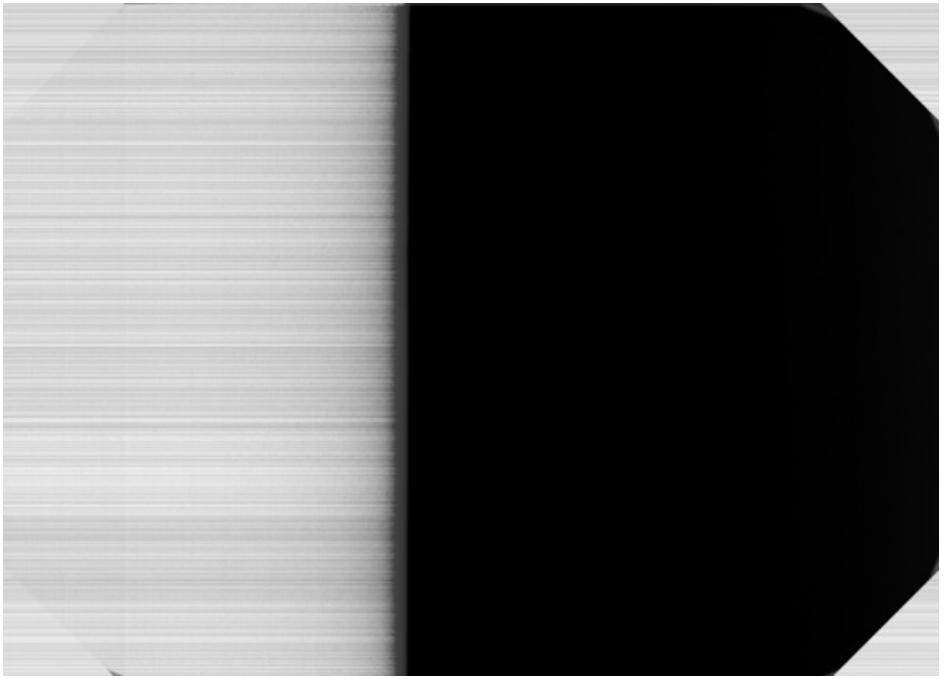
- Run Windows VM through desired actions to copy
- Linux host captures USB packets
- Optional: demo

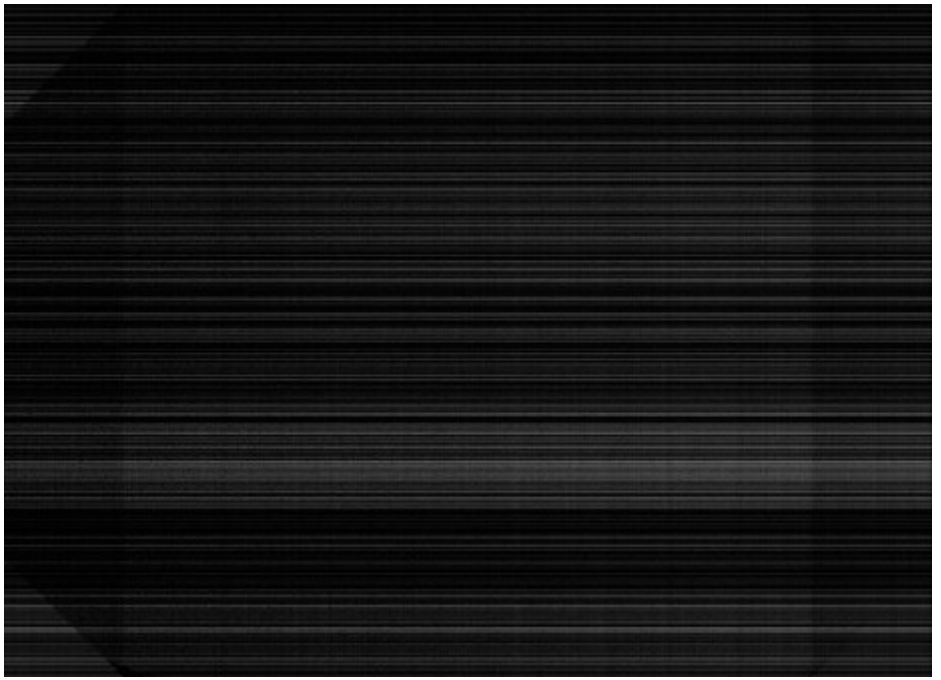
usbrply

- Converts Wireshark packet capture to code
- Rapid application development from .cap
- Add bulk transfers to complete, write to file
- Optional: demo

Decoding the image

- Expect grayscale (no “color”)
- Sensor is odd shape...what will this mean?
- Use lead brick to create easy pattern
- Turns out to be simple 16 bit rectangular data





Fuzzing

- Tried to fuzz to find additional commands
- Accidentally erased EEPROM!

Fixing EEPROM

- Original SW no longer works but mine does
- Example commands to program EEPROM
- First 6 bytes special (VID/PID area)
- Trick: wrap write past alignment

Cleaning up

- Problem: raw capture hard to understand
- What if we want to change exposure, etc?
- Analyze host .dll with .NET Reflector
- Optional: demo

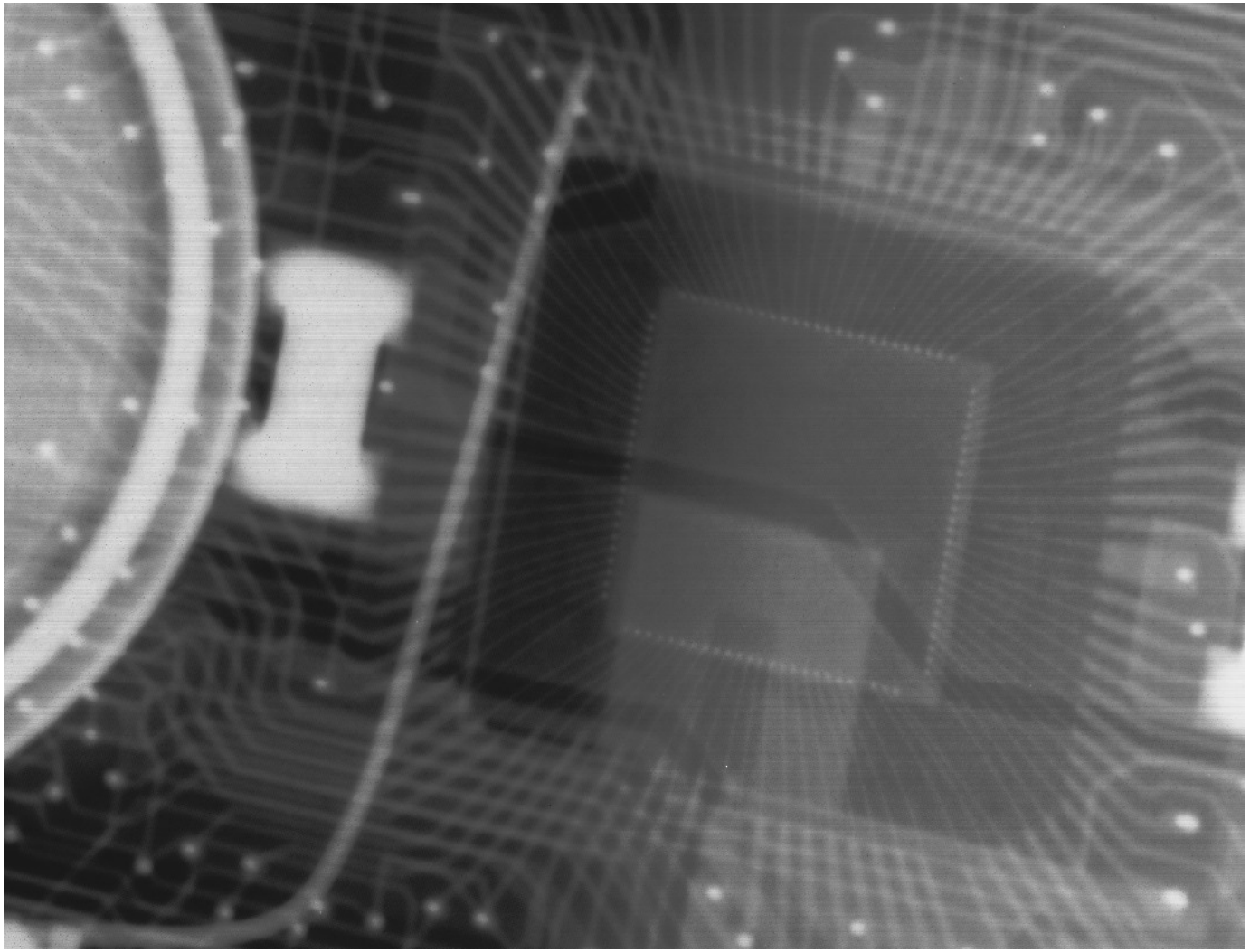
fxload

- Having udev issues getting this to work
- Appreciate help from someone familiar with it
- Workaround: replay firmware load

Next steps

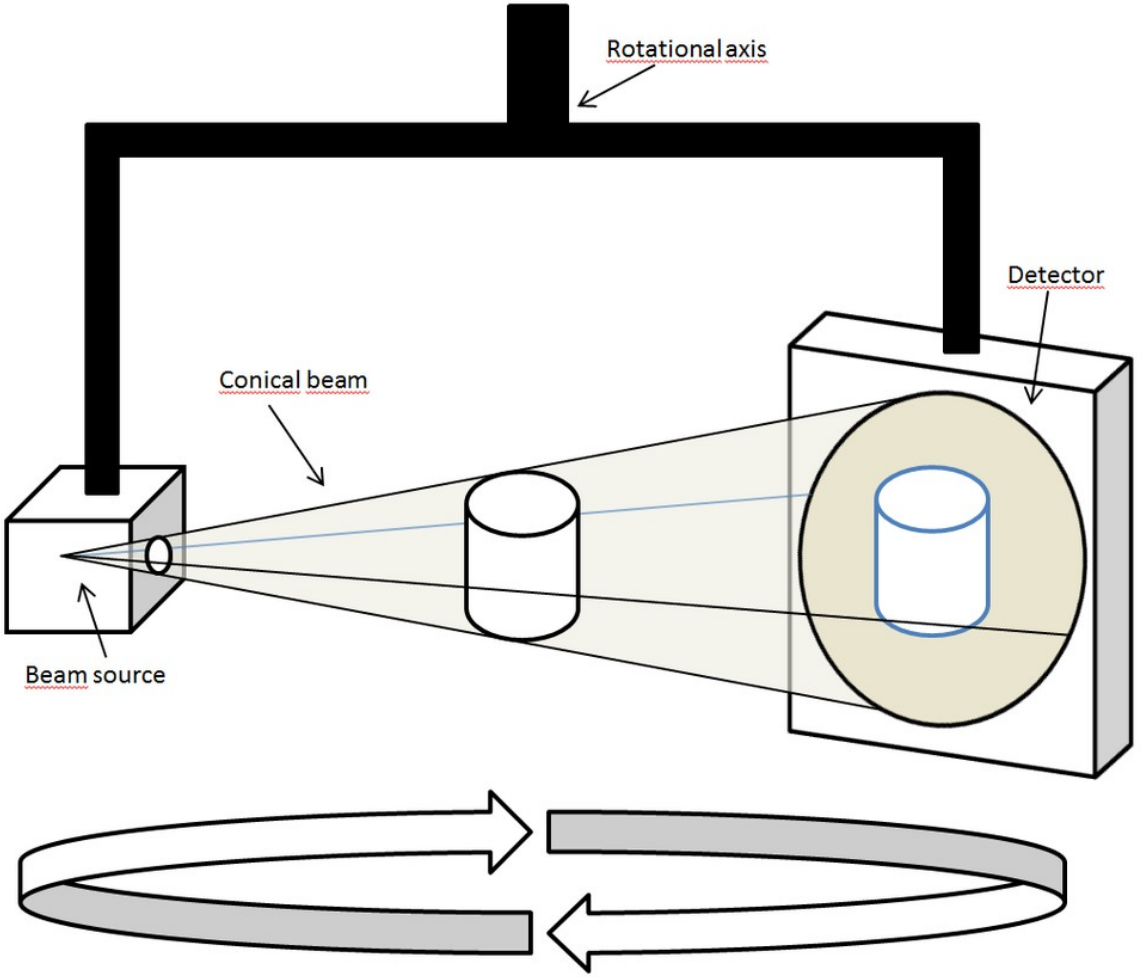
Improving image quality

- Histogram equalization really helps
- Could do more but may be good enough



Cone beam computed tomography (CBT)

- Make 3D model (ex: security module)
- Rotate, take image, repeat

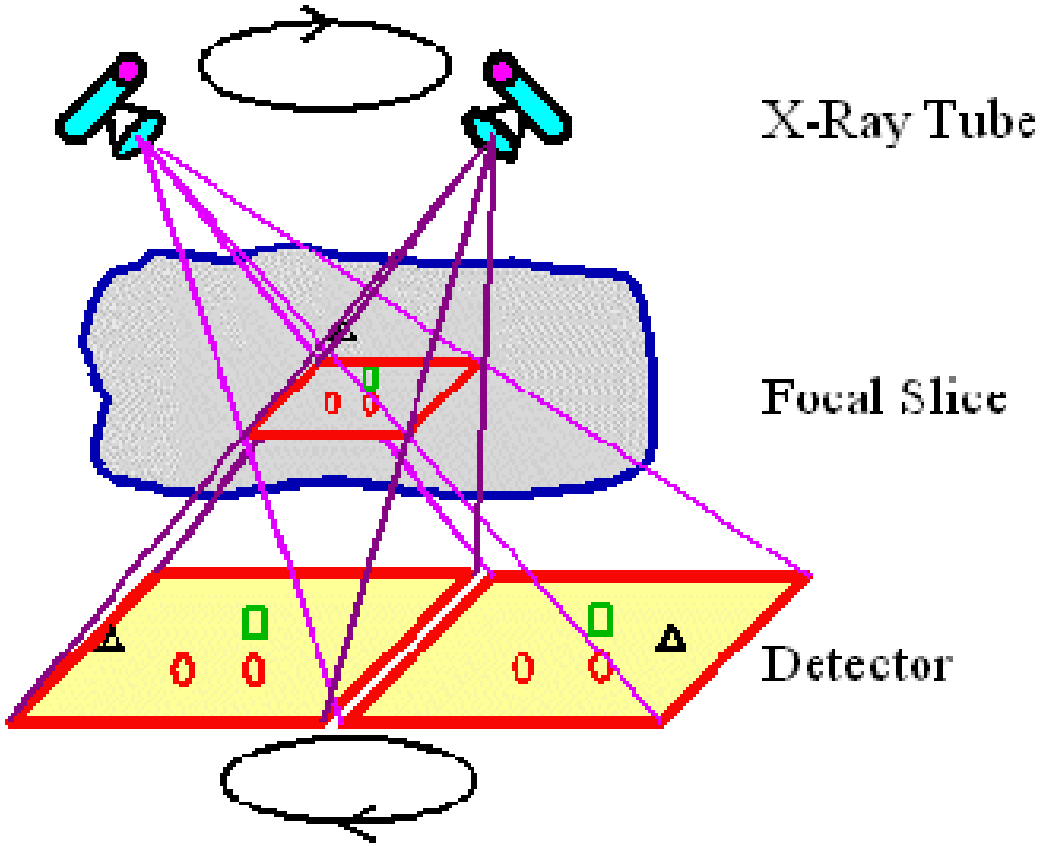


DIY CBCT

- Python script to control x-ray, sensor, indexer
- Video

Future: laminography (“5DX”)

- Well suited for planar objects (ex: PCBs)
- Less published work, more math heavy



Cassette imaging

- Traditional way to do x-rays
- Poor early results, could be improved

Thanks for listening!

- Questions? Interested?
  - JohnDMcMaster@gmail.com
- http://uvicrec.blogspot.com/2015/04/cone-beam-computed-tomography-cbct-test.html
- usbrply: https://github.com/JohnDMcMaster/usbrply/blob/master/main.py