PBS&T

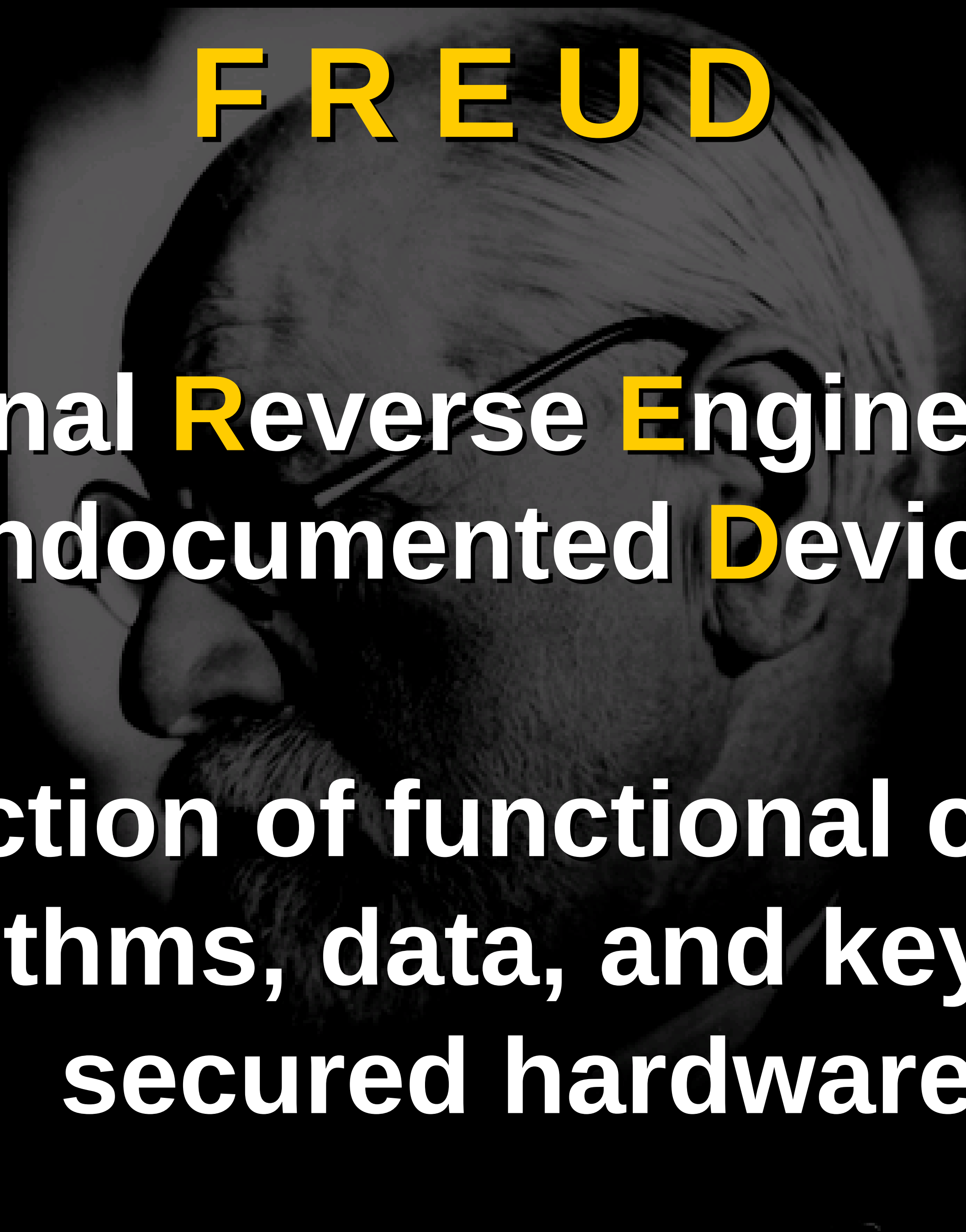
FREUD Applications of FIB

Invasive FIB Attacks and Countermeasures in Hardware Security Devices

Valery Ray

vray@partbeamsystech.com

FIB User Group, Washington DC, USA



FREUD

Functional Reverse Engineering of Undocumented Devices©

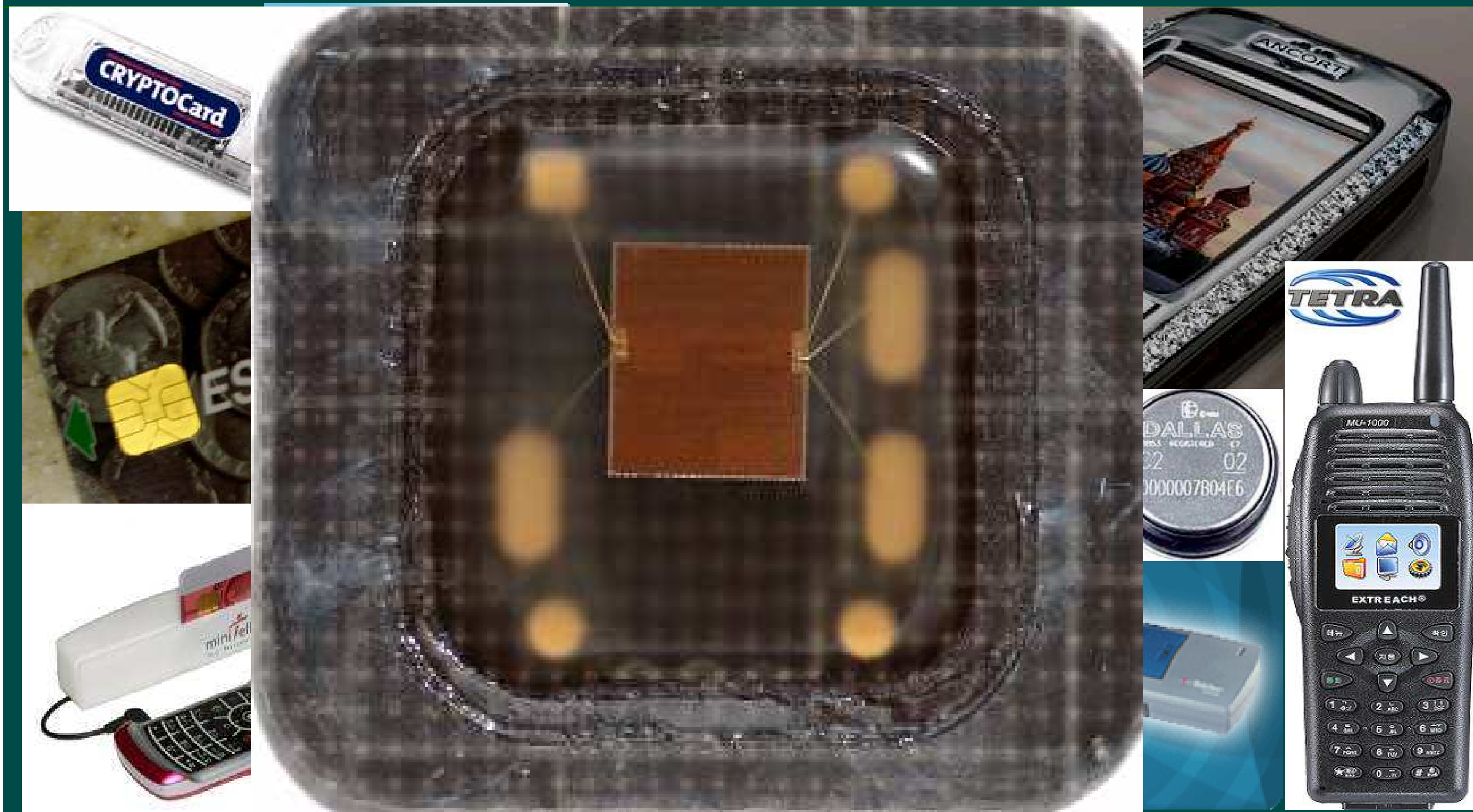
Extraction of functional codes, algorithms, data, and keys from secured hardware

Outline

- Targeted Devices and Applications
- Workflow of FIB “invasion”, challenges and tricks
- Signal extraction and injection
- Limitations of FIB instrumentation
- Countermeasures against FIB methods

Typical Targeted Devices:
µC in Distributed Security and Encryption Applications

Typical Targeted Devices:
µC in Distributed Security and Encryption Applications



Workflow of FIB "Invasion"

- **Navigation on undocumented secure devices**
- **Capturing layout and localizing nodes**
- **Bypassing protective shields**
- **Making contacts, extracting data, injecting signals**

Navigation on Secure Devices

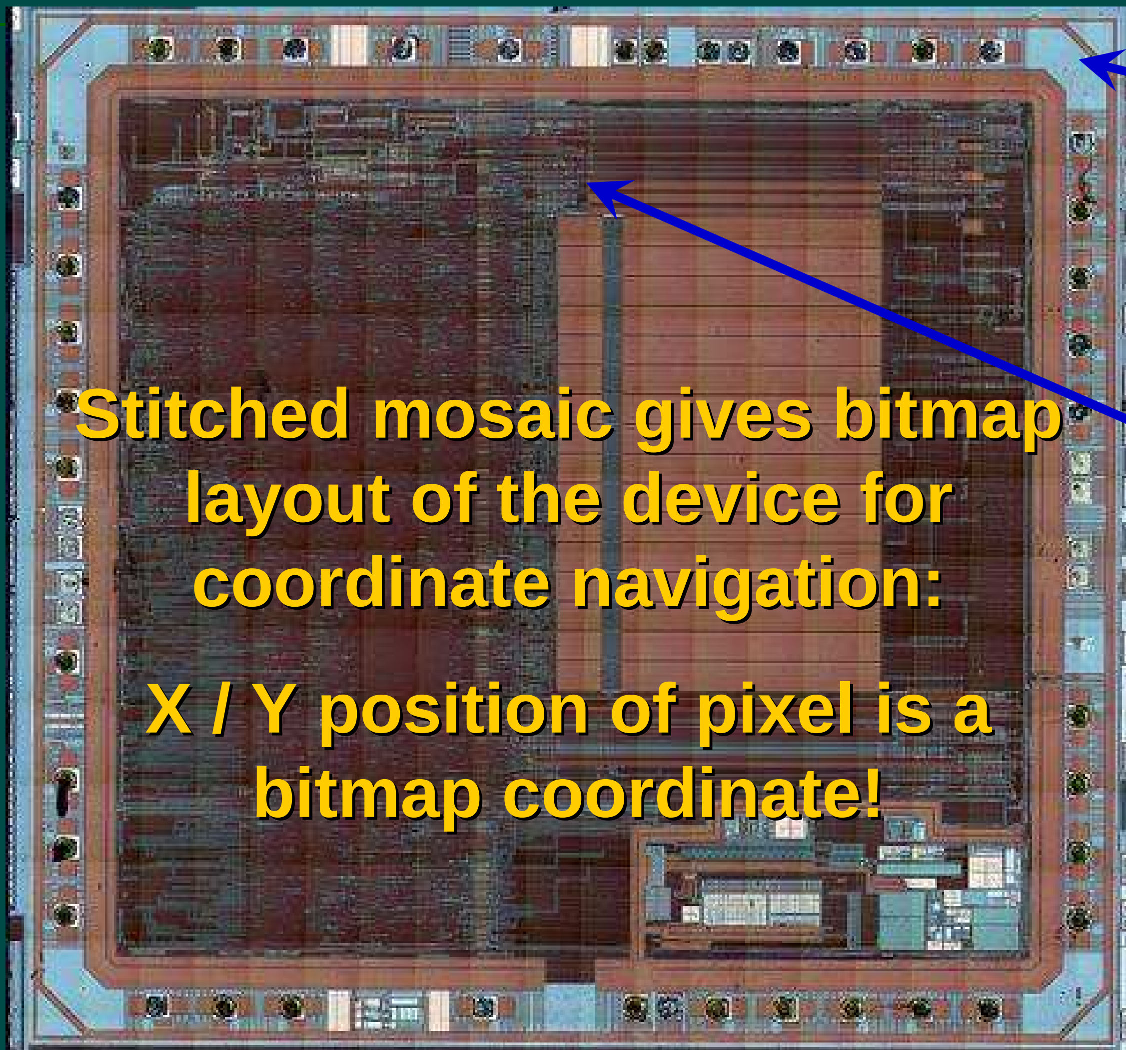
- Shields prevent direct navigation with optics
- Have to use sacrificial device to localize nodes
- Two steps of localization – coarse and precise
- Dual positioning: coordinates and local reference

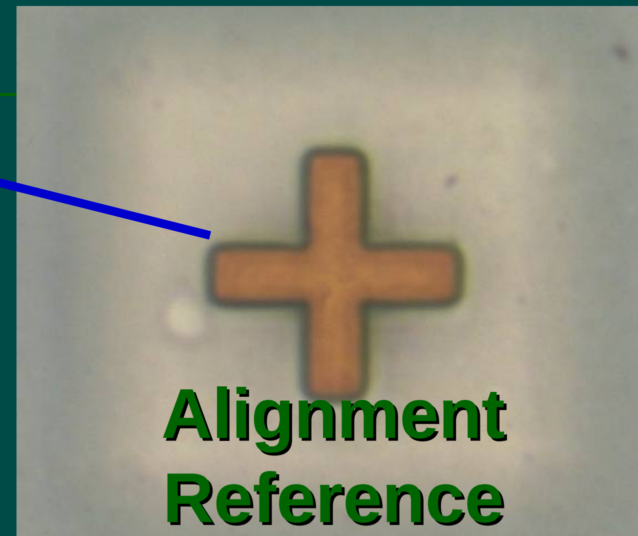
Shield images © Christopher Tarnovsky www.flylogic.net

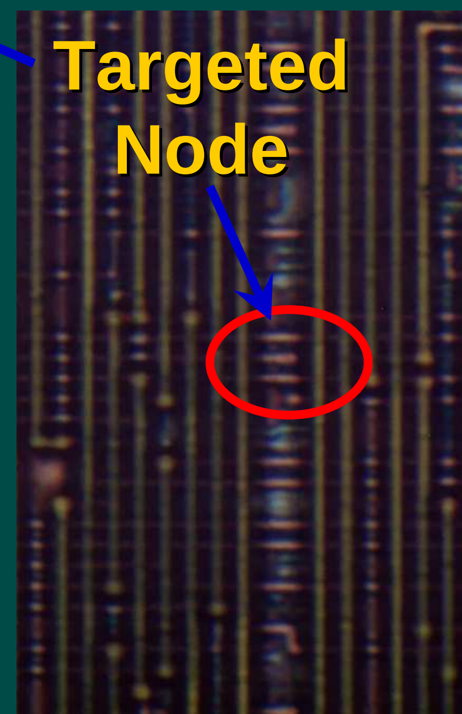
Coarse Localization on Sacrificial Device (s)

- Remove shield (wet chemistry or RIE)
- Scan device under Optical Microscope, stitch mosaic bitmap, locate nodes, define coordinates
- Establish references for coordinate conversion
- Convert bitmap coordinates to FIB stage position

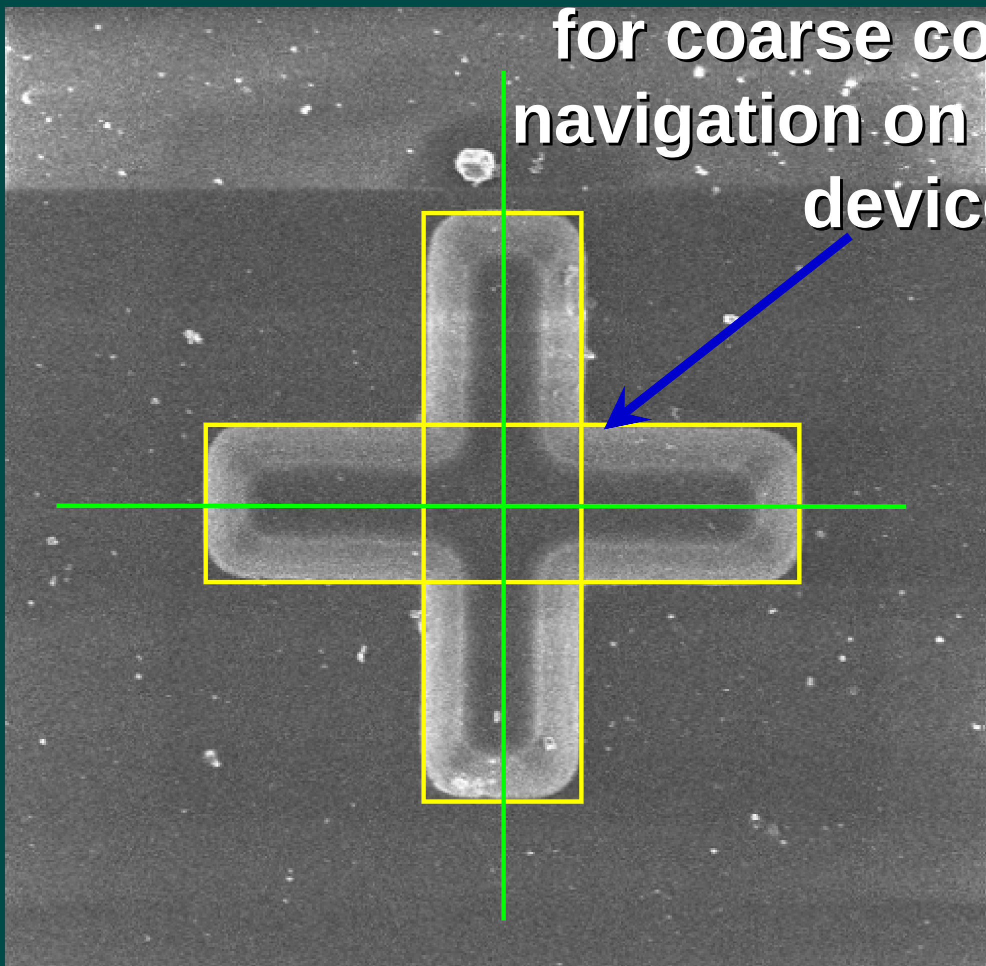
Layout Capture and Node Localization



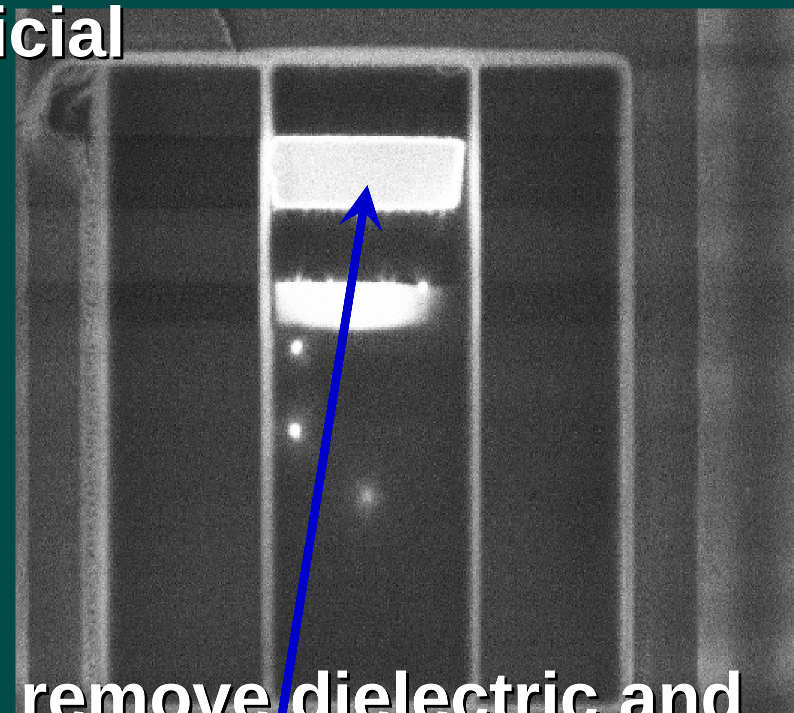




References and Nodes in FIB



Use alignment references for coarse coordinate navigation on sacrificial device



Then remove dielectric and capture precise position of node

Navigation with Local Alignment

- Accuracy of FIB stage is limited – how to navigate on small-linewidth devices?
- Use coordinates for coarse navigation
- Use protective shield as your local reference!
- Try to make contacts in between shield lines, if impossible then bypass the shield

Bypassing Shield

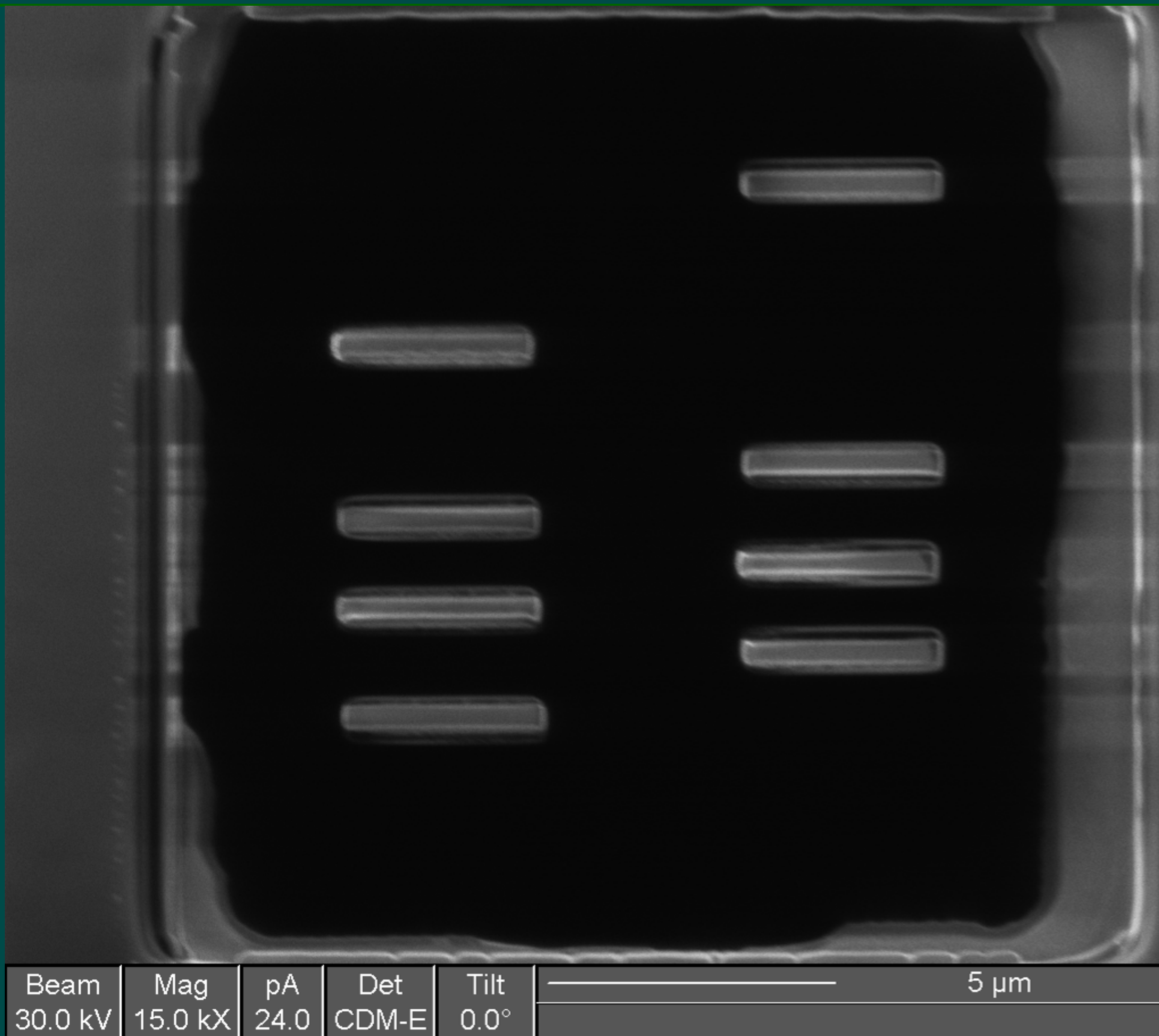
- Bypass entire shield
  - » Best for analog shields, works for some digital
  - » Takes 30 to 60 min. of FIB time per device
  - » Often need follow-up by non-FIB techniques
  - » Shield can be removed to speed up further work

- Bypass protective shield locally
  - » Works on analog and digital shields
  - » On <200nm devices 2 – 3 shield lines may need bypassing to clear space for each contact
  - » Takes 30 to 120 min. of FIB time per bypass

Disabling Shield

- Disable shield control circuitry
  - » Requires detailed analysis of layout
  - » Simulate “OK” shield on input of test circuitry
  - » Cut output of charge pump – disable flash erase!
  - » Cut “security interrupt” outputs, tie to “1” or “0”

Making "Large" Contacts: Direct Line Probing

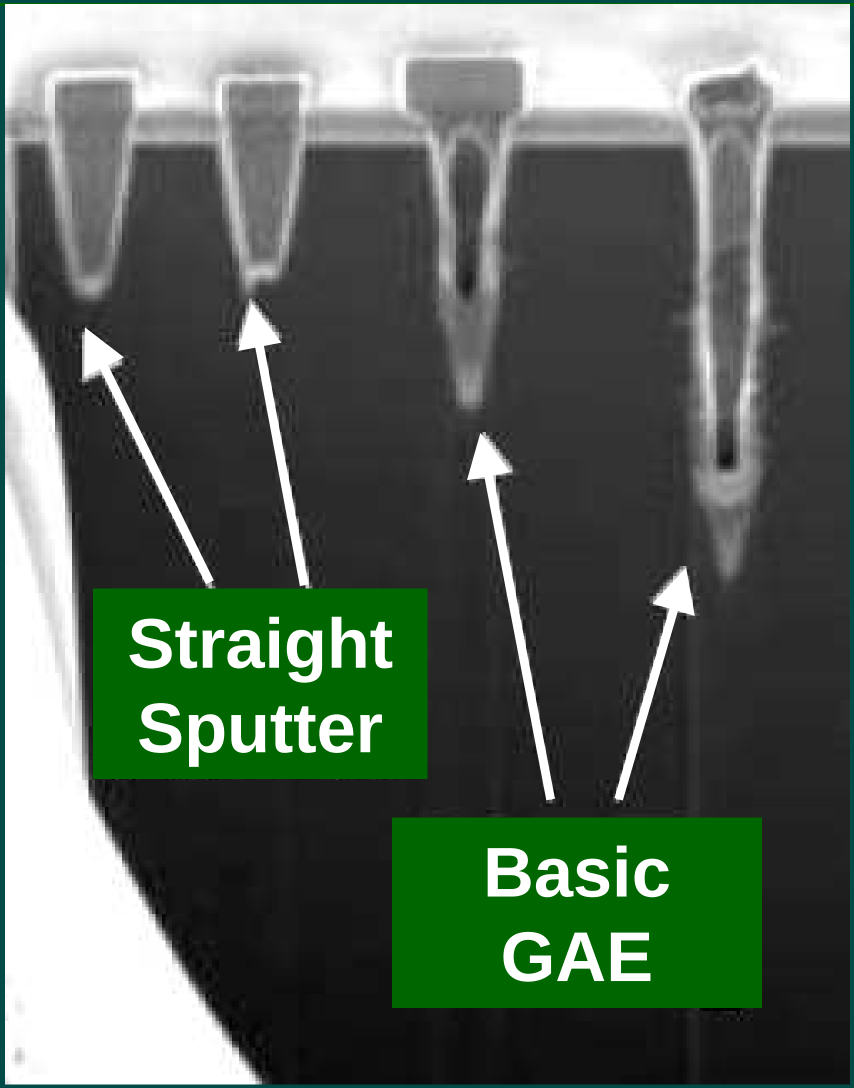


On devices with spacious layout and line-width ≥ 350nm direct probing of internal nodes may be possible.

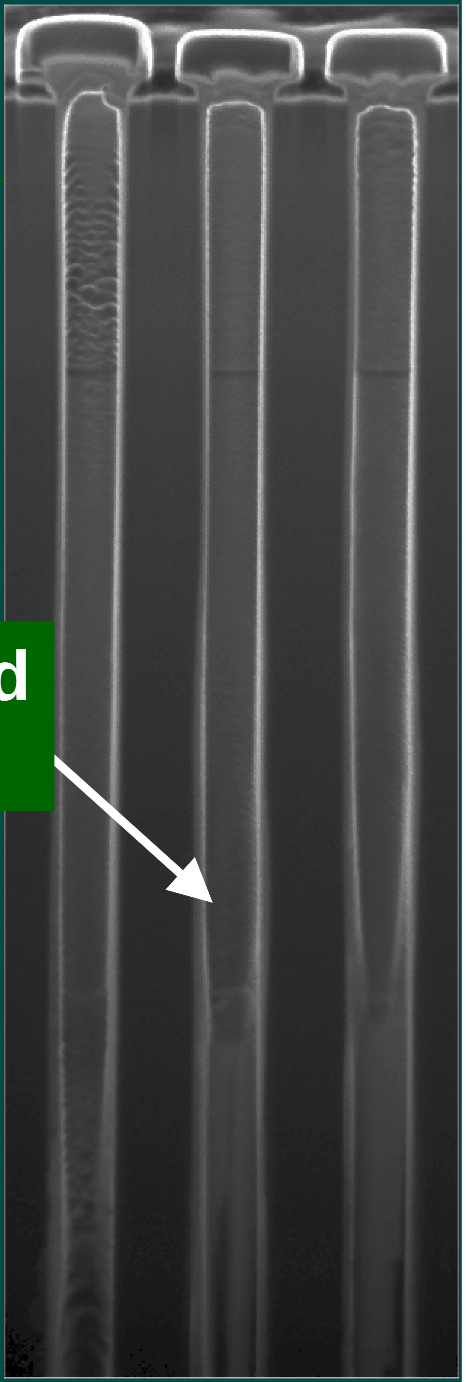
FIB Image of bus opening

www.flylogic.net

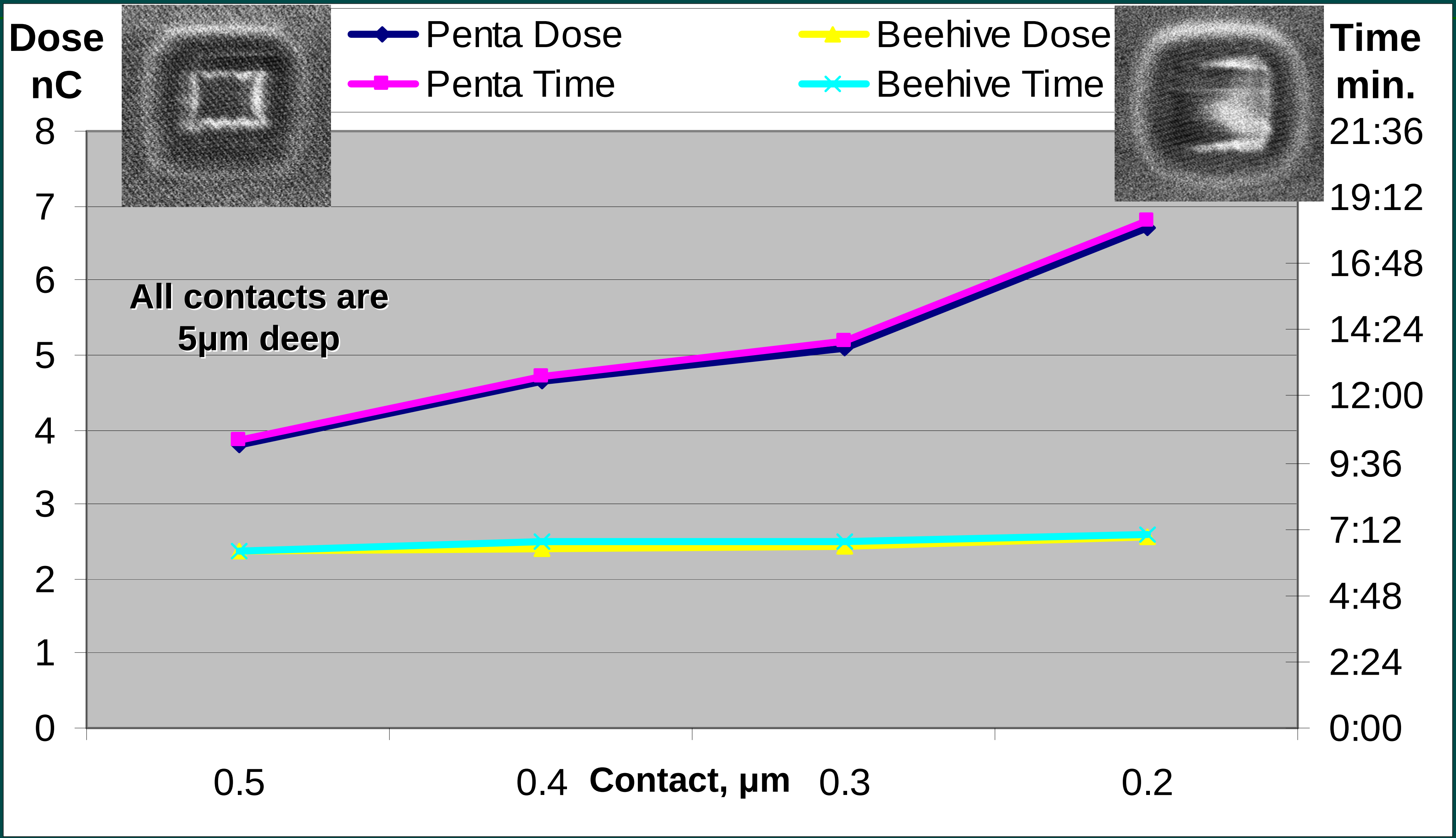
Making Small Contacts: High Aspect Ratio Milling



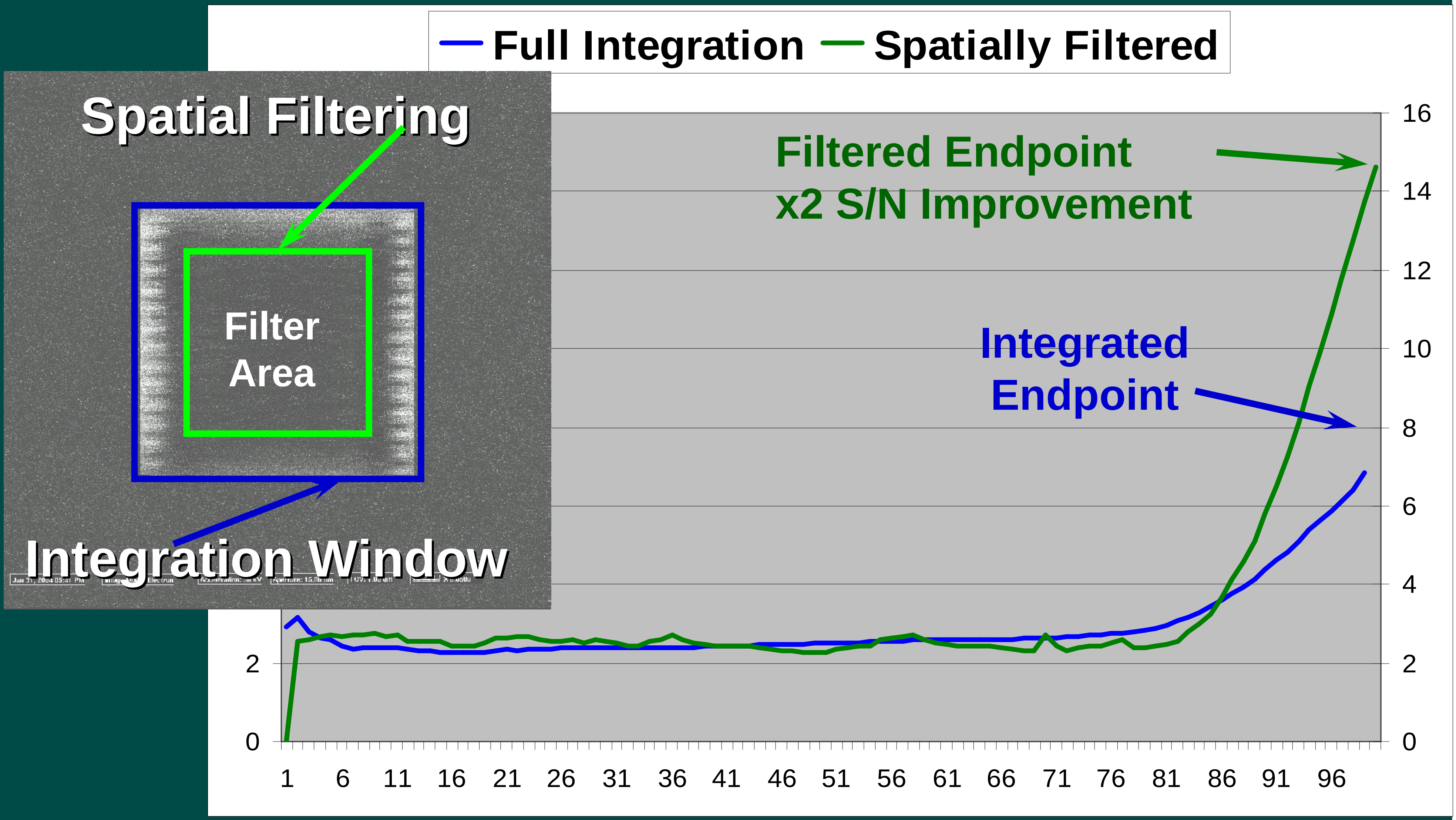
Optimized
GAE



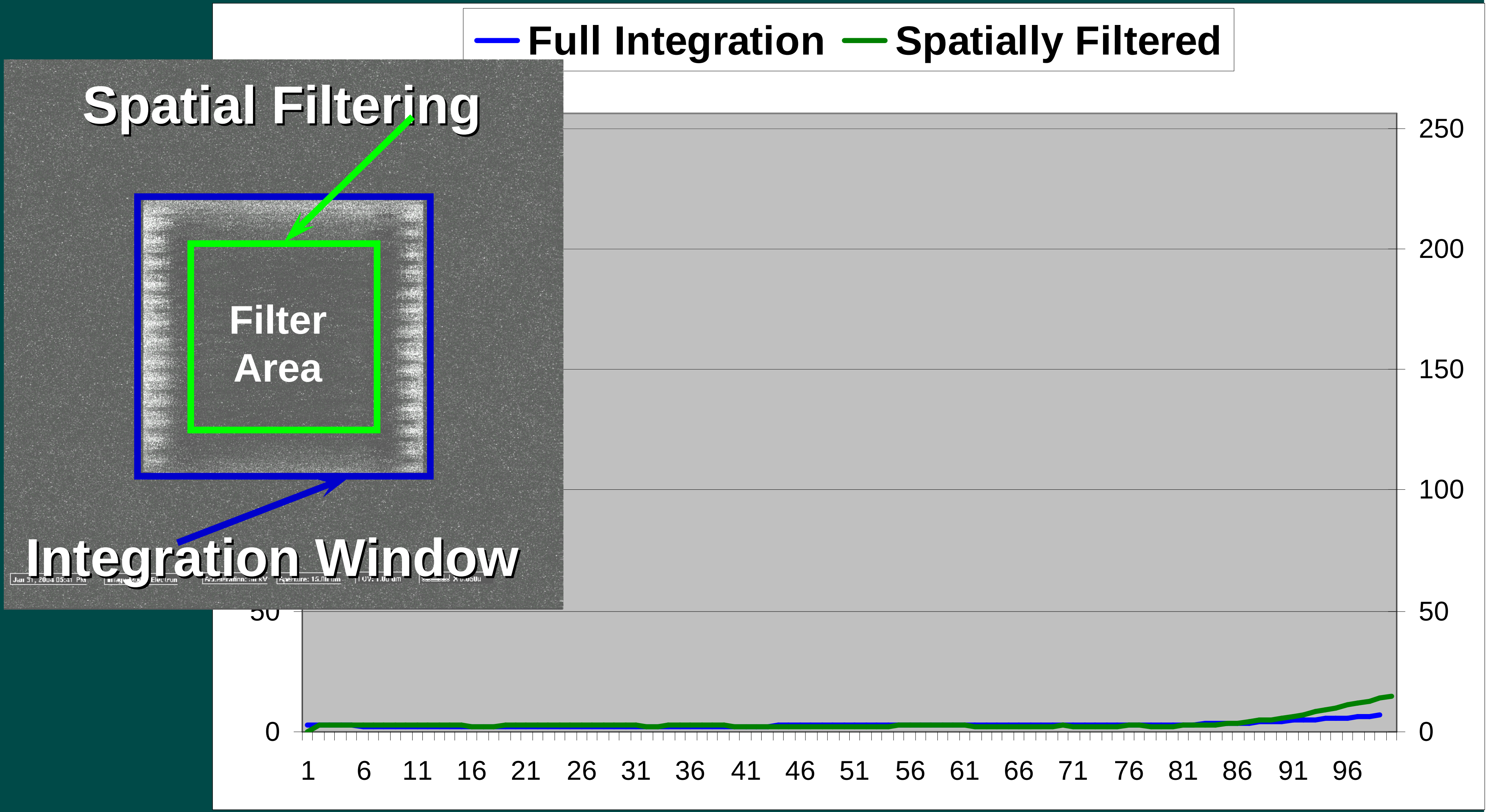
Making Small Contacts: High Aspect Ratio Milling Throughput



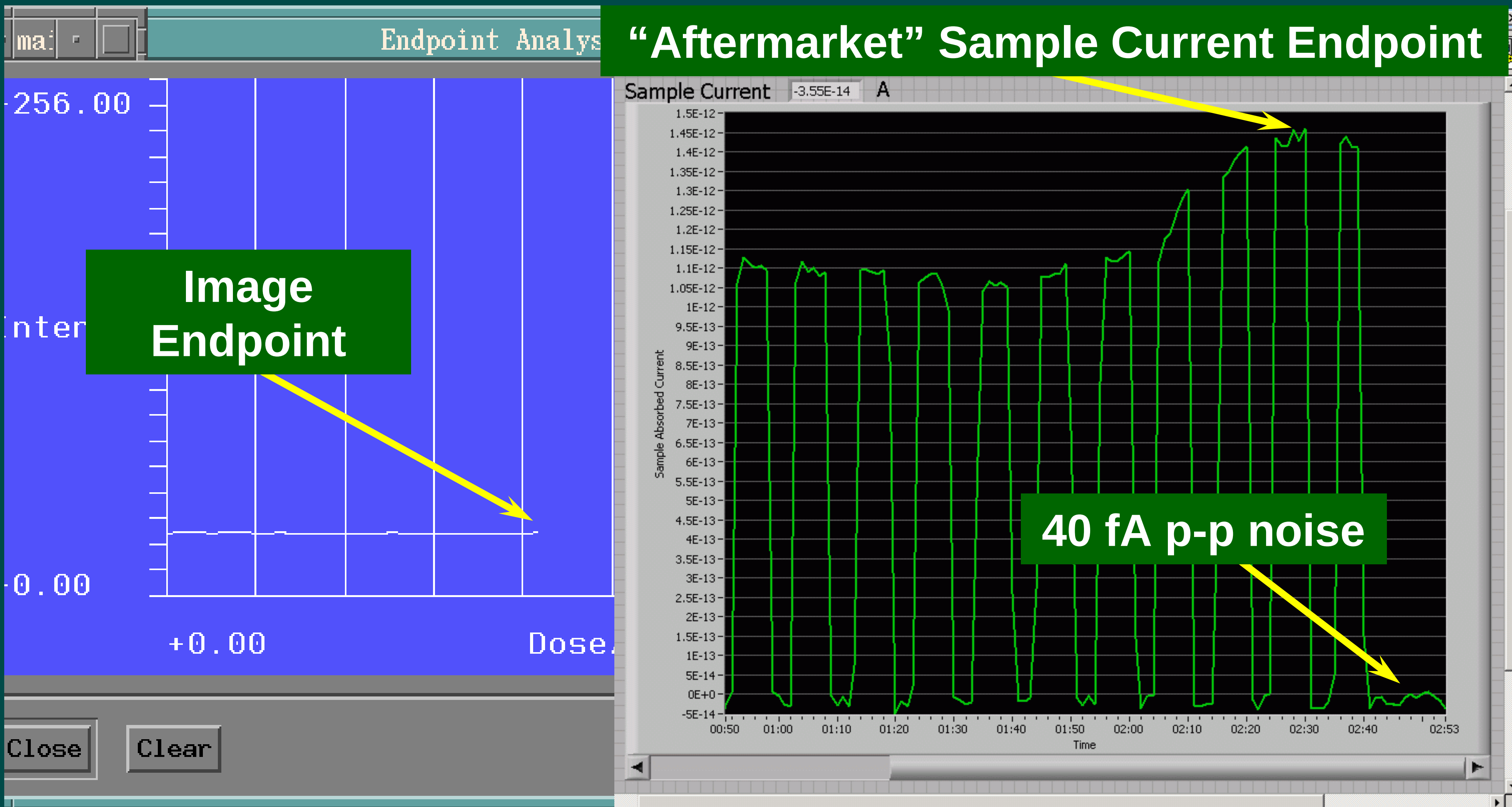
Making Small Contacts:
High Aspect Ratio Endpointing by Image



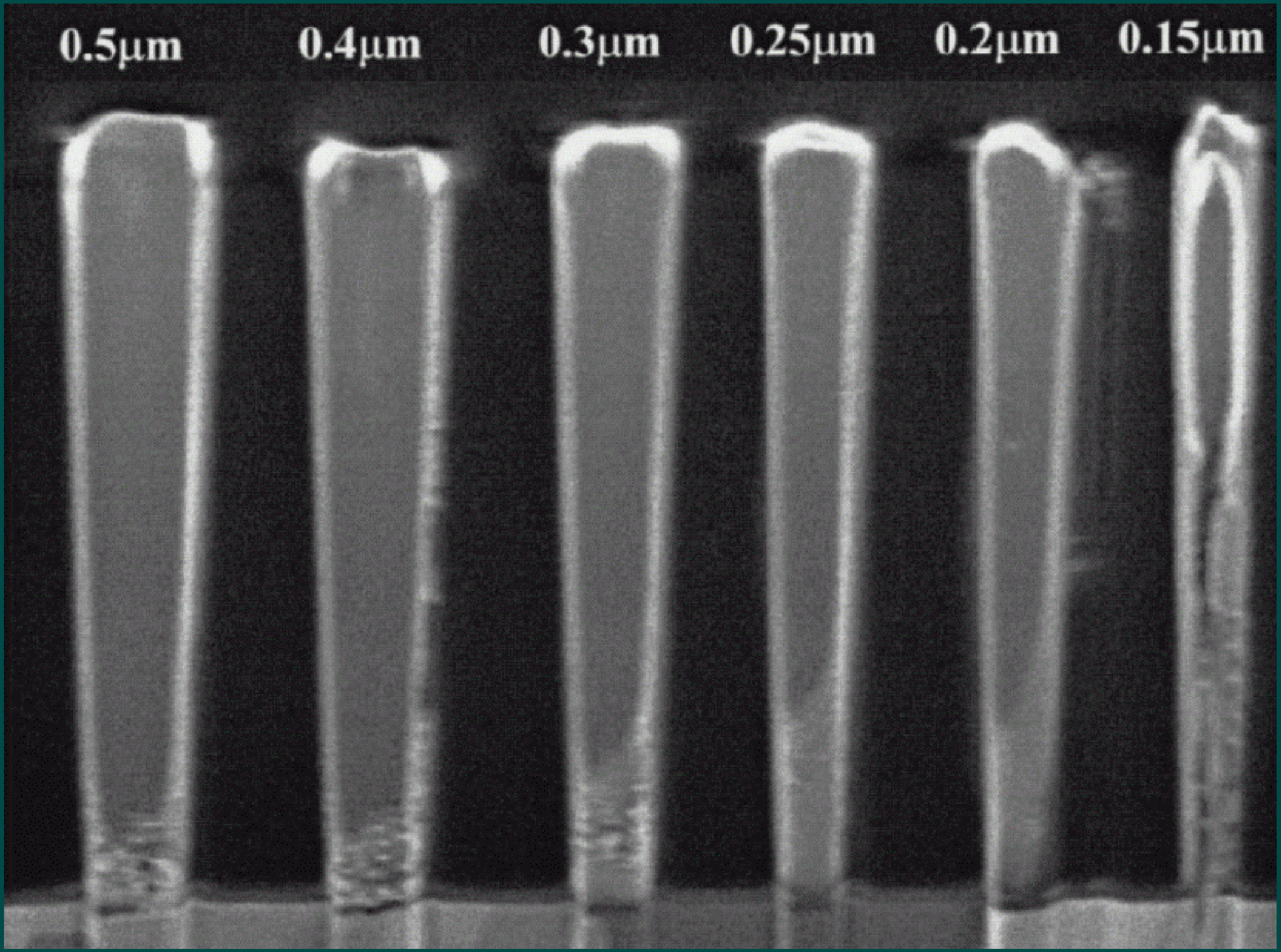
Making Small Contacts:
High Aspect Ratio Endpointing by Image



Making Small Contacts: High Aspect Ratio Endpointing by Current



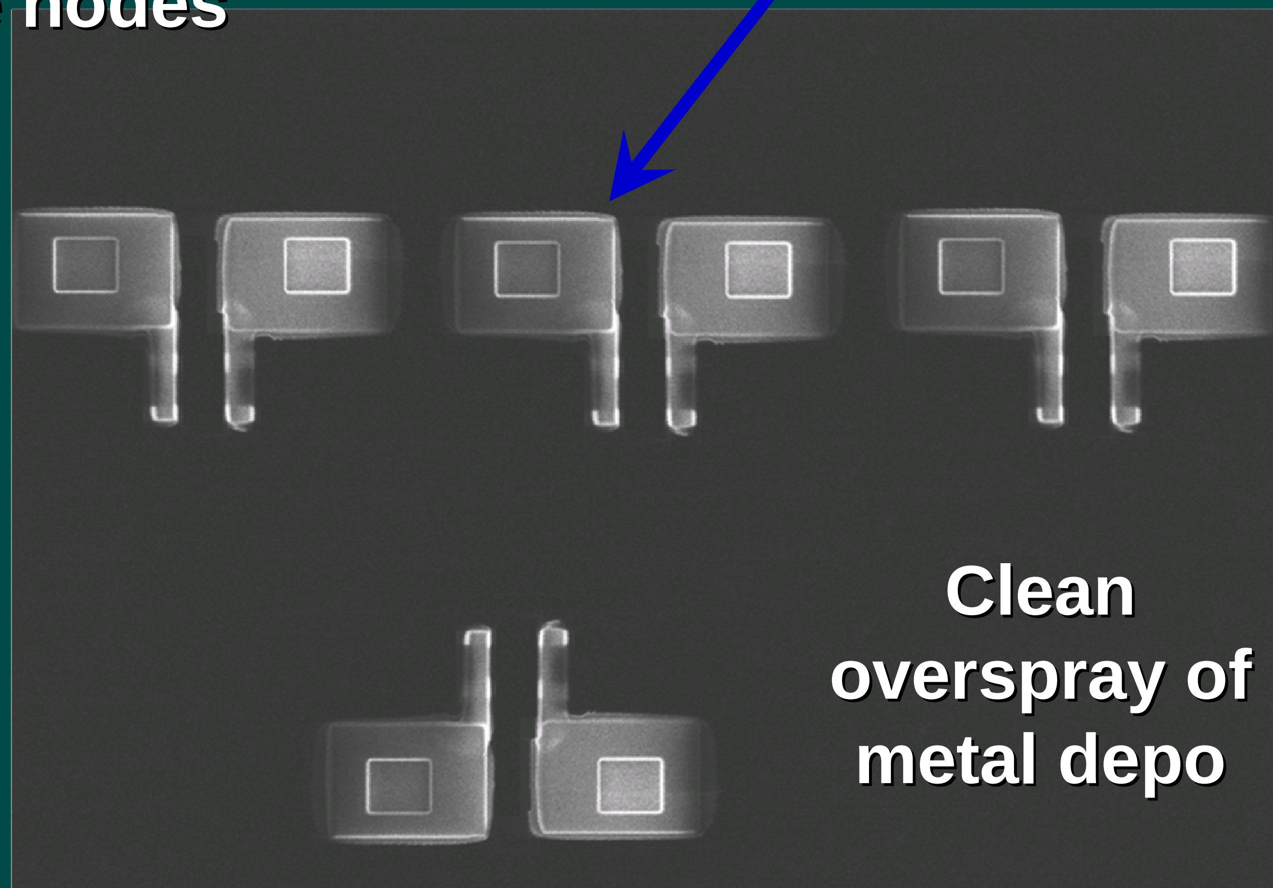
Making Small Contacts: High Aspect Ratio Deposition



Making Contacts and Pads

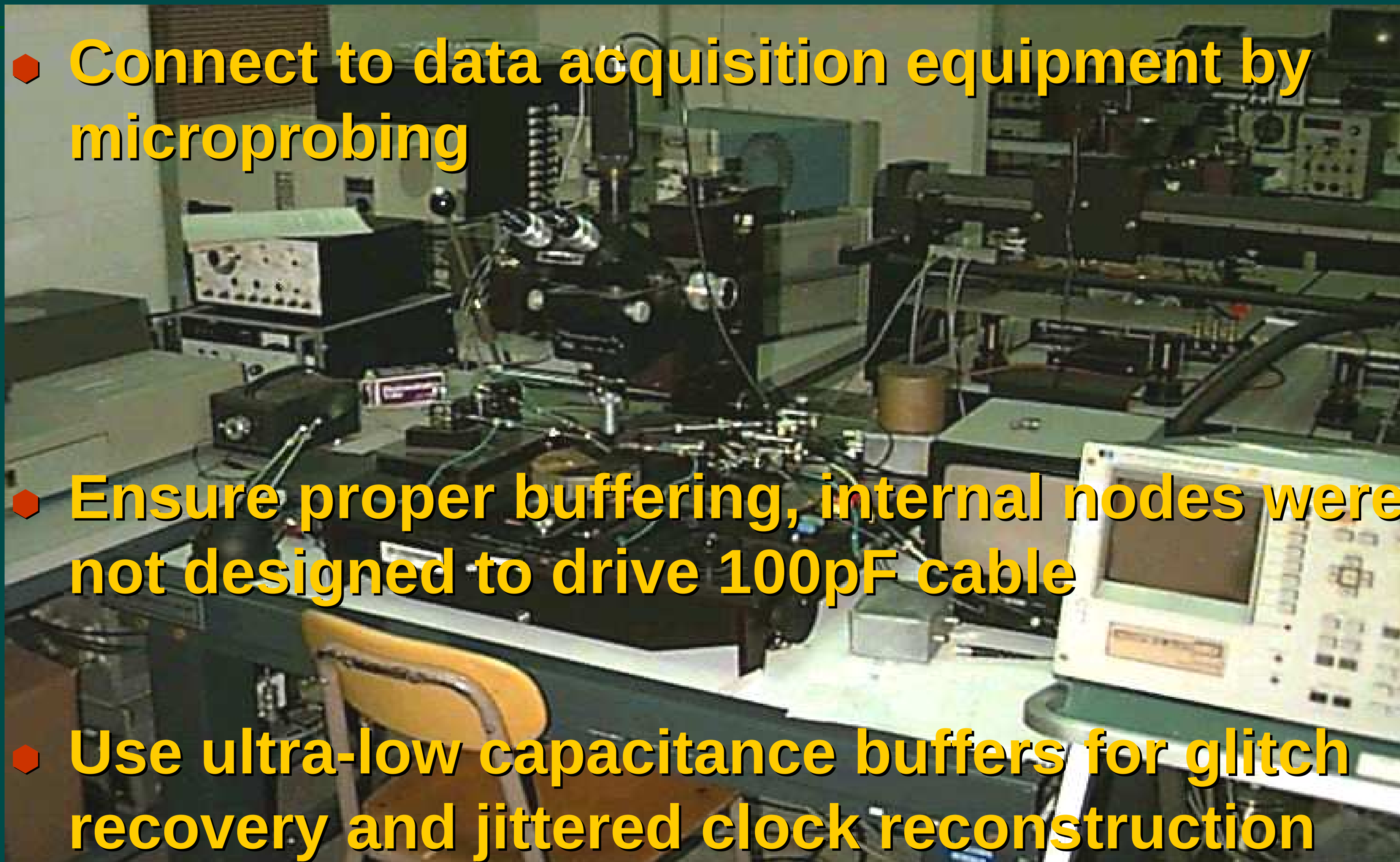
HAR vias connecting to the nodes

Contact pads for probing



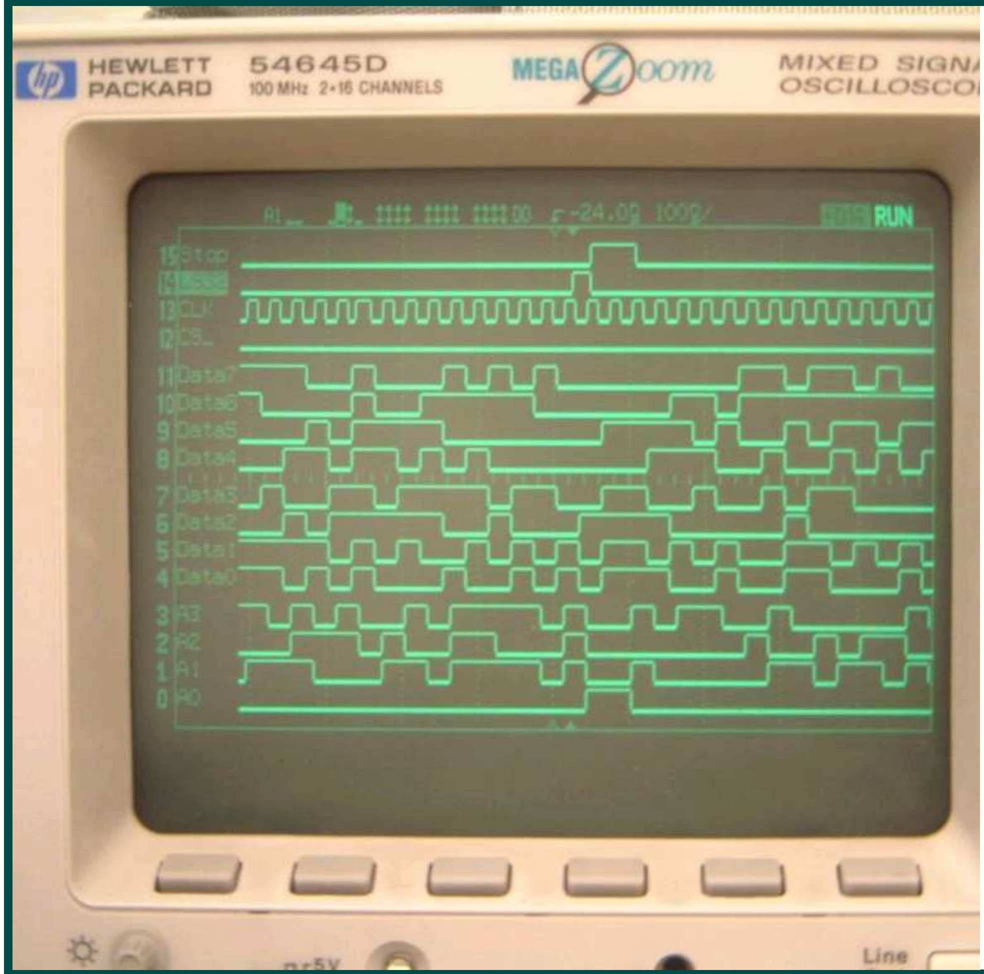
Clean overspray of metal depo

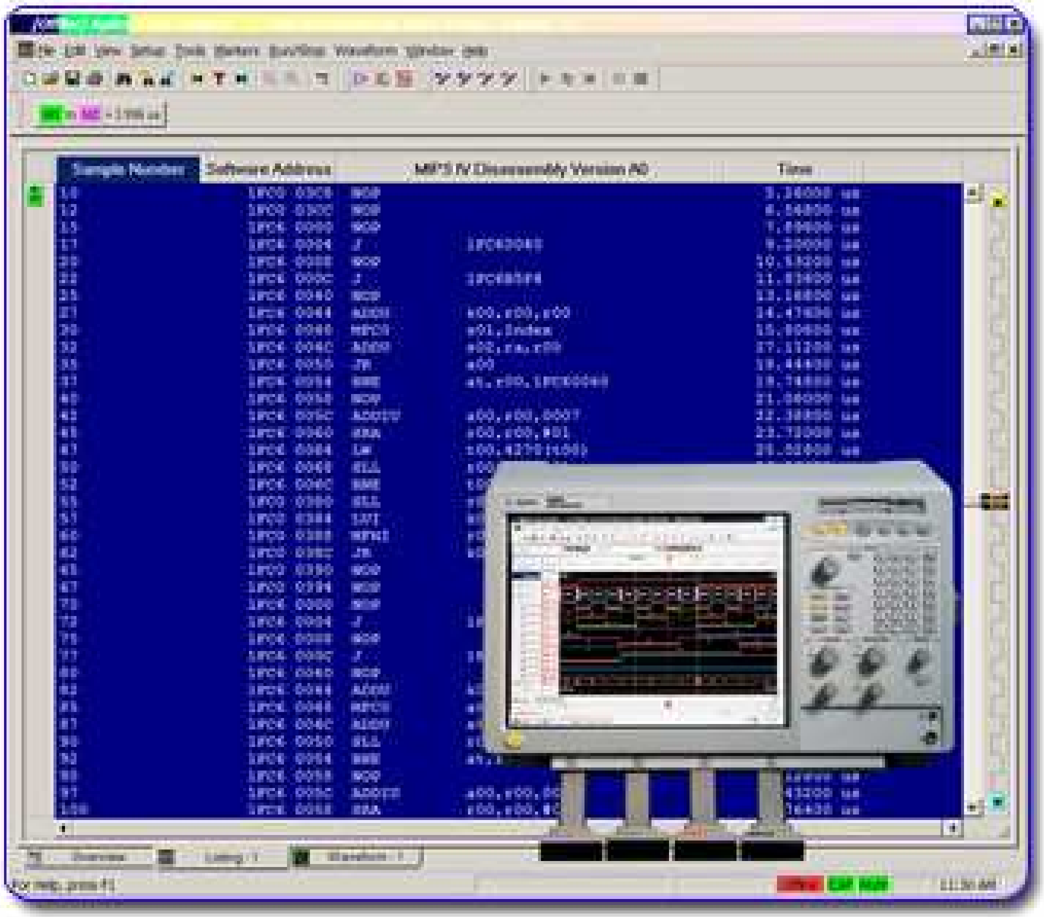
Data Extraction



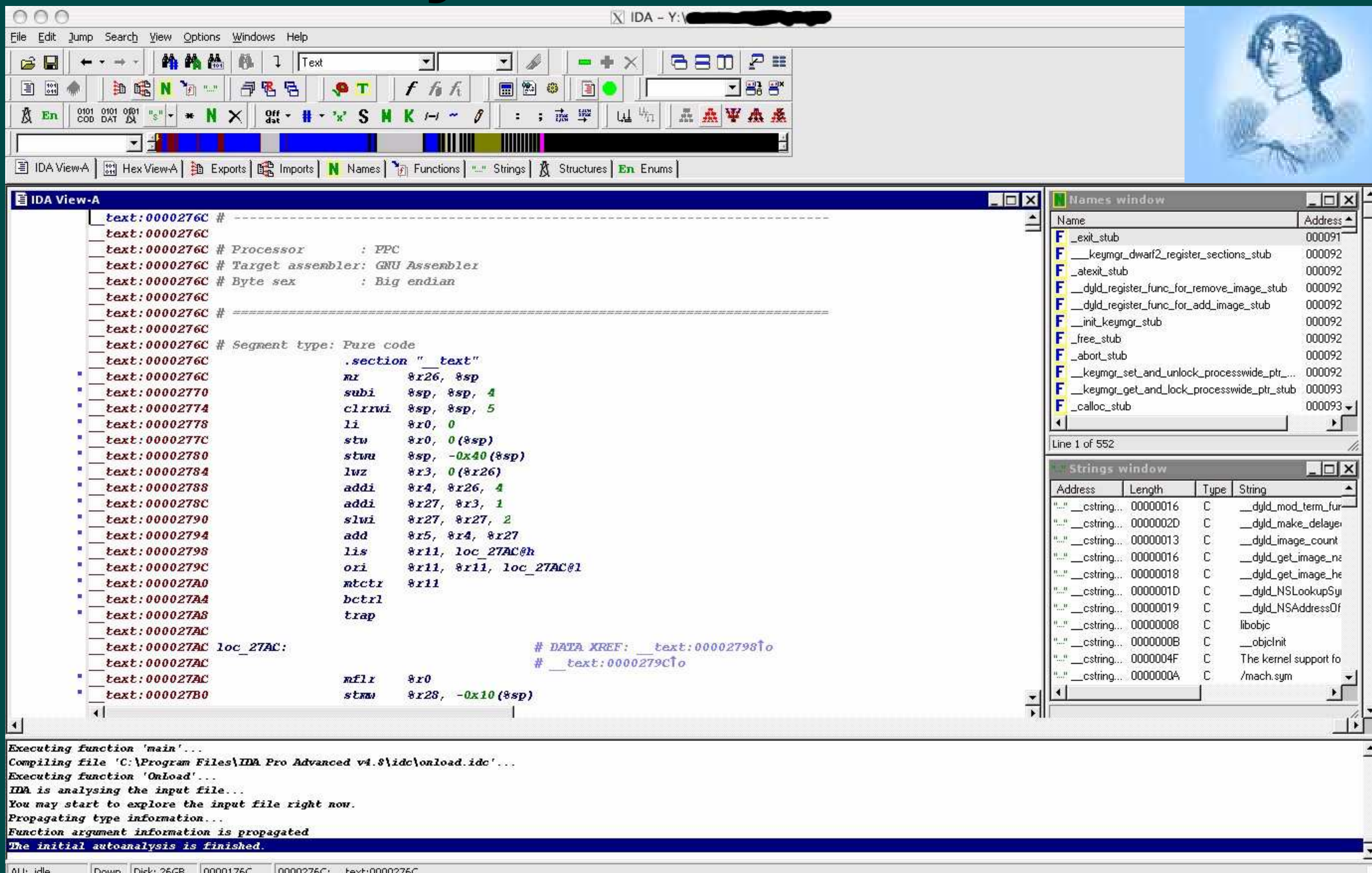
- Connect to data acquisition equipment by microprobing
- Ensure proper buffering, internal nodes were not designed to drive 100pF cable
- Use ultra-low capacitance buffers for glitch recovery and jittered clock reconstruction

Filter, Align, and Convert to HEX Code





Disassemble, decompile, and make yourself at home ☺



Signal injection

- Injection of impulses into data bus can alter execution of embedded code
- Basic application: disruption of “End Of Loop” command in Answer To Reset (ATR) function of smartcard microcontrollers may cause extraction of data memory ☺
- Suitable injection buffers are not available from OEMs of pattern generators or otherwise, must DIY

Limitations of existing FIB technology

- **Accuracy of navigation**
  - » **Targeting nodes by only the coordinates on devices with linewidth < 200nm is unreliable on most FIBs**
- **Aspect ratio of contacts**
  - » **Endpoint of milling on contacts with 20:1 depth/width may require “aftermarket” upgrades**
- **Linewidth (technology node) limitations**
  - » **Making multiple deep contacts smaller then 250 nm is esoteric art, takes very dedicated operator…**

Countermeasures against FIB

- **FIB attacks can be made uneconomical for casual FIB access:**
  - » **Planarise devices and use small linewidth technology**
  - » **Thick copper plane combined with active shields is difficult to cut**
  - » **Use Liquid Crystal Polymer for encapsulation – hard to remove**
  - » **Use combination of analog and digital shields: bypass is difficult**
  - » **Introduce "position jitter" to shield layer on lithography step to prevent local referencing for navigation**
  - » **Orient <u>straight</u> shield lines at 45 degrees angle to the layout**

Summary

- FREUD by FIB methods can't be prevented, but can be made too expensive for casual access
- Basic countermeasures are relatively simple in manufacturing – planarisation of devices, more sophisticated active shields
- Advanced countermeasures become viable as cost of IC manufacturing is reduced: active double-shielding, LCP encapsulation, 45 degrees shield-to-layout orientation and shield position "jittering"

www.partbeamsystech.com