Trigger in HEP (III)

Tiehui Ted Liu
Fermilab

Tropical Seminar on Frontier of Particle Physics 2011 –
Detector and Electronics

Aug 18-22, 2011, Beijing

Outline of lectures

- Trigger in HEP (I): the view from Physics
- Trigger in HEP (II): the view from Instrumentation
- Trigger in HEP (III): the view into the Future

Decided to take a very different approach to the lectures:

cover much less, explain more, with only a few selected topics,
bias towards collider experiments

FEE/Trigger/DAQ go hand-in-hand.
Have to drop a lot interesting topics in my lectures, but Patrick Le Du will cover all the rest. Please see his talks/slides to get a more complete view.

Lecture III

- Trigger in HEP II: the view into the future
  - A good place to look for most up-to-date materials: TIPP 2011 conference
  - A reminder from Lecture II: challenges in track trigger
  - A few selected examples on R&D ideas (Note: only have time to mention some activities at Fermilab. There are many other good ideas/ projects out there at other places. No time to cover. See TIPP2011)
  - Data transfer challenge: VICTOR chip
    - R&D project at Fermilab
  - Pattern Recognition Challenge: 3D AM chip → VIPRAM
    - R&D Proposal at Fermilab
  - Homework for students
  - Data Formatting issues: ATCA vs VME (in the backup slides)
  - Processing power challenge: GPU for low latency trigger in HEP?

This is not a survey, and by no means complete. Just a few examples to give student a rough idea about some of the R&D issues for track trigger…

TIPP 2011 conference

- A good place to look for most up-to-date materials on "Technology and Instrumentation in Particle Physics" (TIPP): June 9-14, 2011, Chicago.
- Just google "TIPP 2011"
- Or directly visit: http://conferences.fnal.gov/tipp11/
- Click on "Program", then "Time-Table" under program information to see daily program and all the talks
  - Many sessions on Trigger&DAQ
  - Many sessions on FEE, Semiconductor, Photons … etc
  - See full list of tracks in next slide

TIPP 2011 Tracks (13)

- Experimental detector systems
- Gaseous detectors
- Semiconductor detectors
- Calorimeters
- Particle identification
- Photon detectors
- Dark Matter Detectors
- Neutrino Detectors
- Astrophysics and space instrumentation
- Front-end electronics
- Trigger and data acquisition systems
- Machine Detector Interface and beam instrumentation
- Instrumentation for medical, biological and materials research

More than 360 talks presented at the conference.
it is a library for good talks on detector & instrumentations

Fermilab

The approach for soliciting abstracts/talks for TIPP 2011

- The conference is not about beauty-contest type of talks
- Talks should start with science motivations, then
  - focus on the challenges
  - how the experiment overcame the challenges
  - experiences in designing & building, lessons learned
  - in particular, what challenges still struggling to overcome
  - and focus on ideas on how to break the barriers (*innovation*)
- More about confession than beauty-contest

People seem to like this approach …

TIPP 2011 received 450 abstracts:
360 oral presentations, 80 posters

Second International Conference on

Technology and Instrumentation in Particle Physics

June 9-14, 2011

TIPP 2011

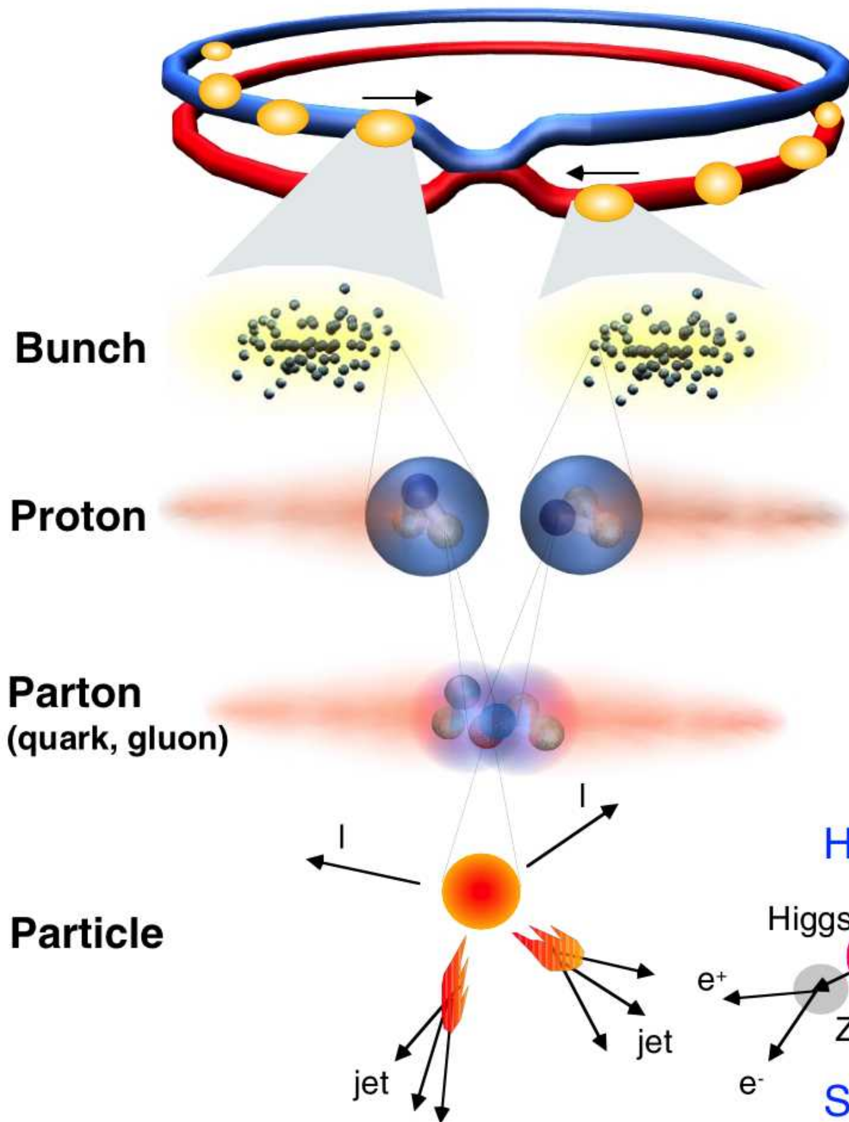
Tons of good reading materials after the lectures

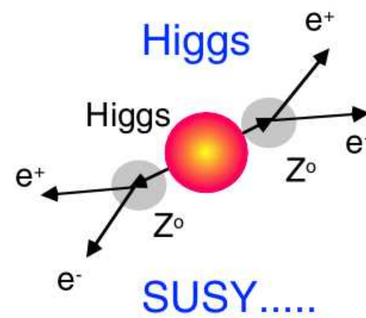
What we learned from Lecture II?

- A brief summary in 24 seconds…

Fermilab

LHC Collisions





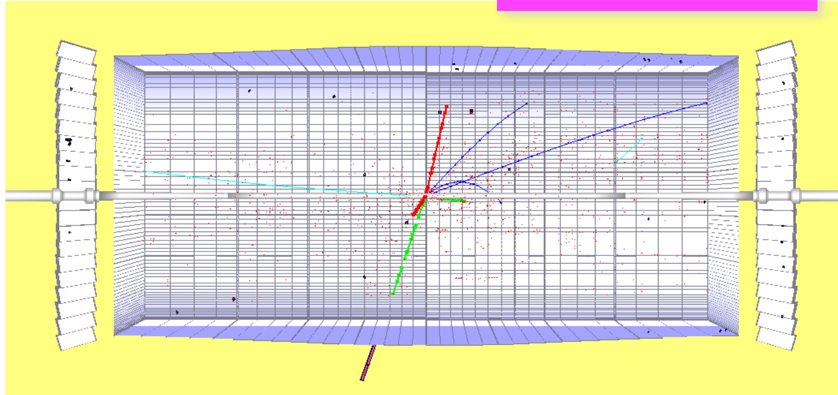
Proton-Proton	2835 bunch/beam
Protons/bunch	10^{11}
Beam energy	7 TeV (7×10^{12} eV)
Luminosity	10^{34} cm^{-2} s^{-1}
Crossing rate	40 MHz

every bunch crossing
~ 25 collisions
~2000 particles produced

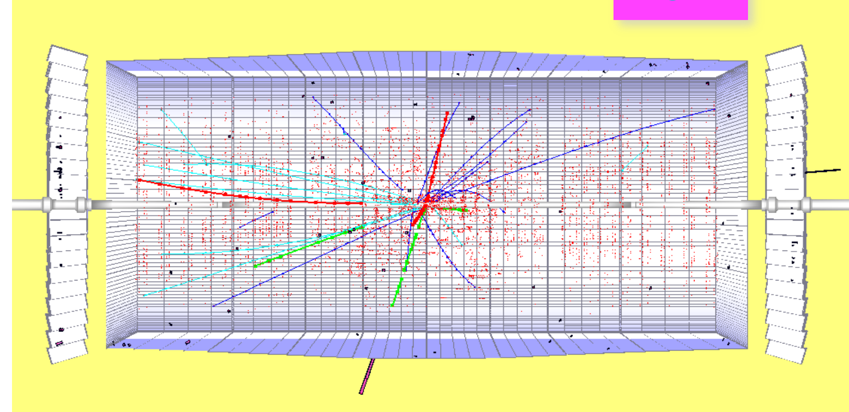
Fermilab

Example of future challenges: CMS Tracking and Triggering

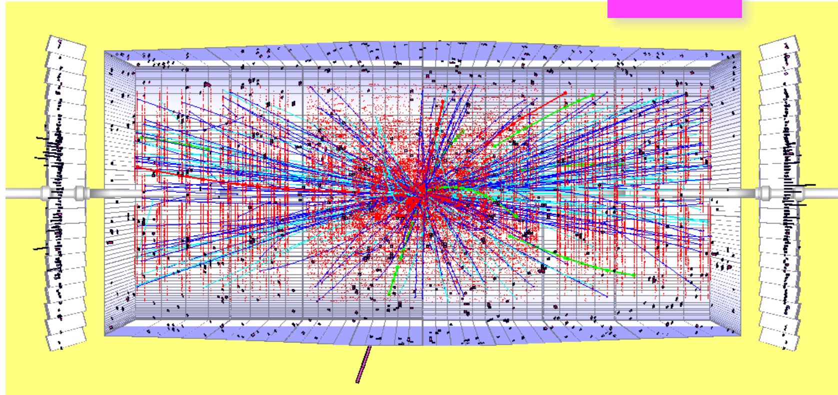
10^{32} cm^{-2} s^{-1}



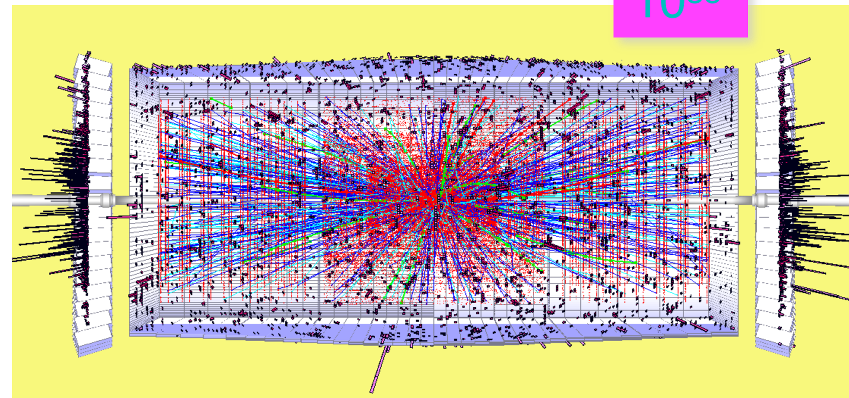
10^{33}



10^{34}

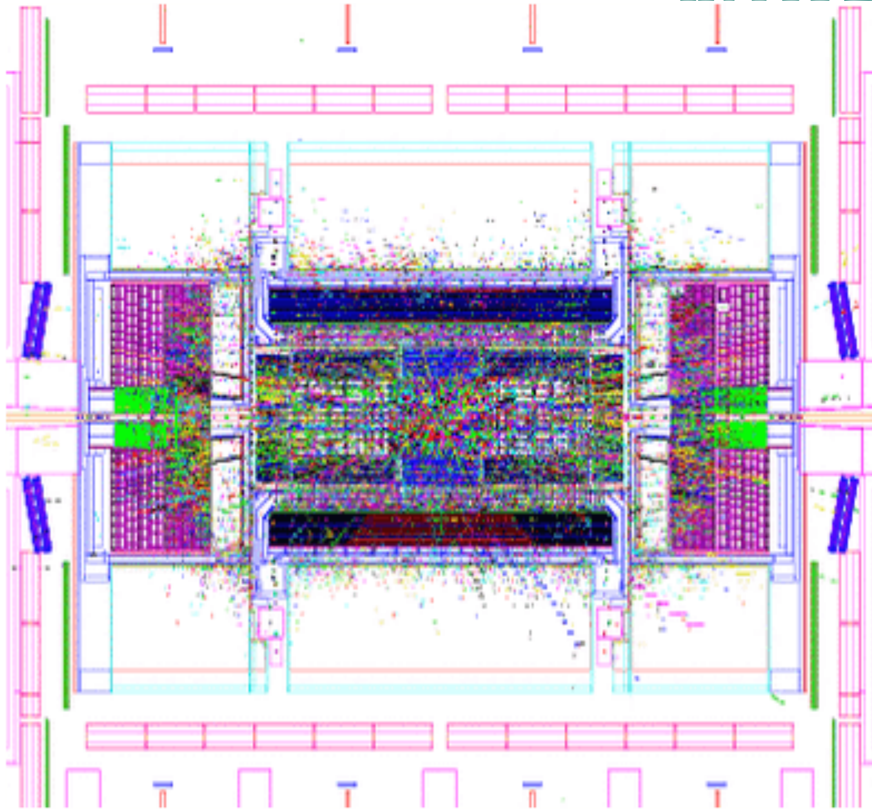


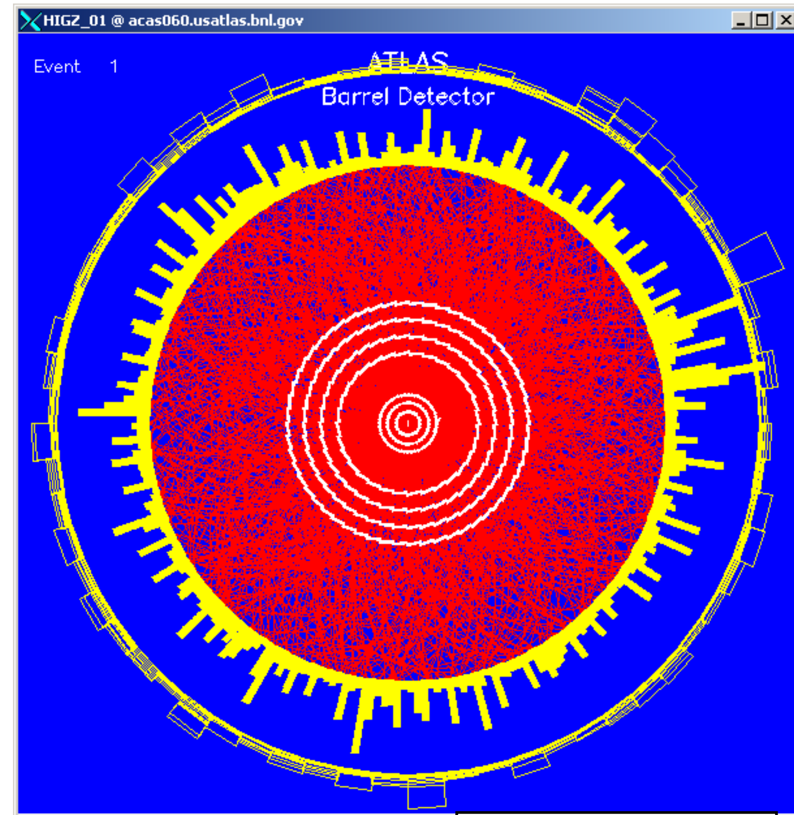
10^{35}



simulation

Expected Pile-up at High Lumi LHC in ATLAS at 10^{35}





$N_{ch}(|y| \leq 0.5)$

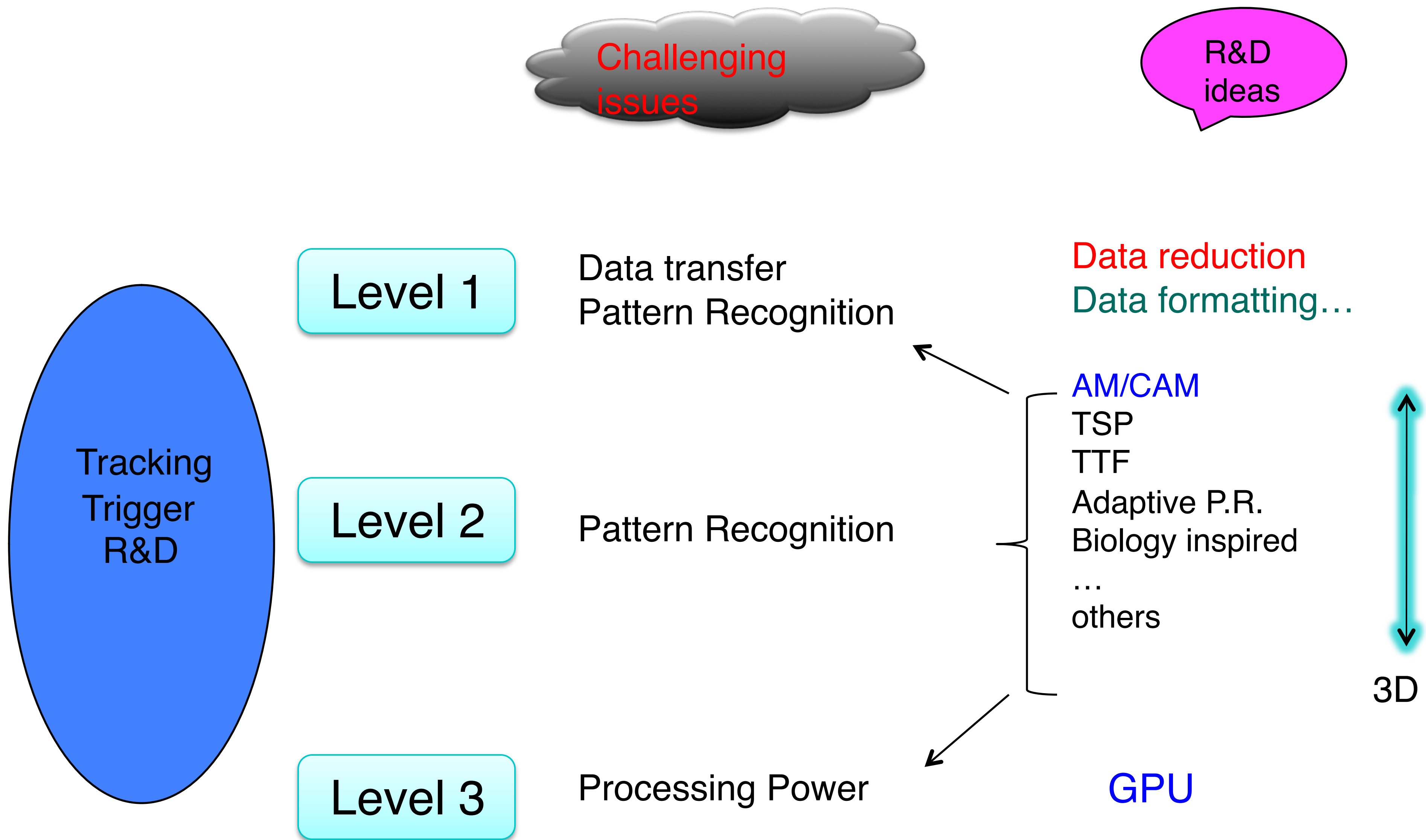
- **230 min.bias collisions per 25 ns. crossing**
- **~ 10000 particles in $|\eta| \leq 3.2$**
- **mostly low p_T tracks**
- **requires upgrades to detectors**

Fermilab

Challenge in Tracking Trigger

- The PAST: hardware-based pattern recognition for fast track triggering has been very successful for HEP
  - CDF SVT: based on AM (Associative Memory) or CAM (Content-Addressable Memory) for pattern recognition
  - CDF SVT: ~400K patterns --> 6M patterns --> >~ 1B needed for LHC
- THE FUTURE: enormous challenges in implementing pattern recognition for tracking trigger at LHC (L1&L2), due to
  - much higher occupancy and event rates at the LHC
  - detectors much more massive
  - much larger number of channels in their tracking volumes
  - (Likely similar issues for some high-intensity frontier experiments)
- There is a clear need to develop/improve the hardware-based pattern recognition technology to advance the state-of-the-art
  - One example: AM/CAM R&D for HEP
  - Or think harder to come up smart/crazy ideas …

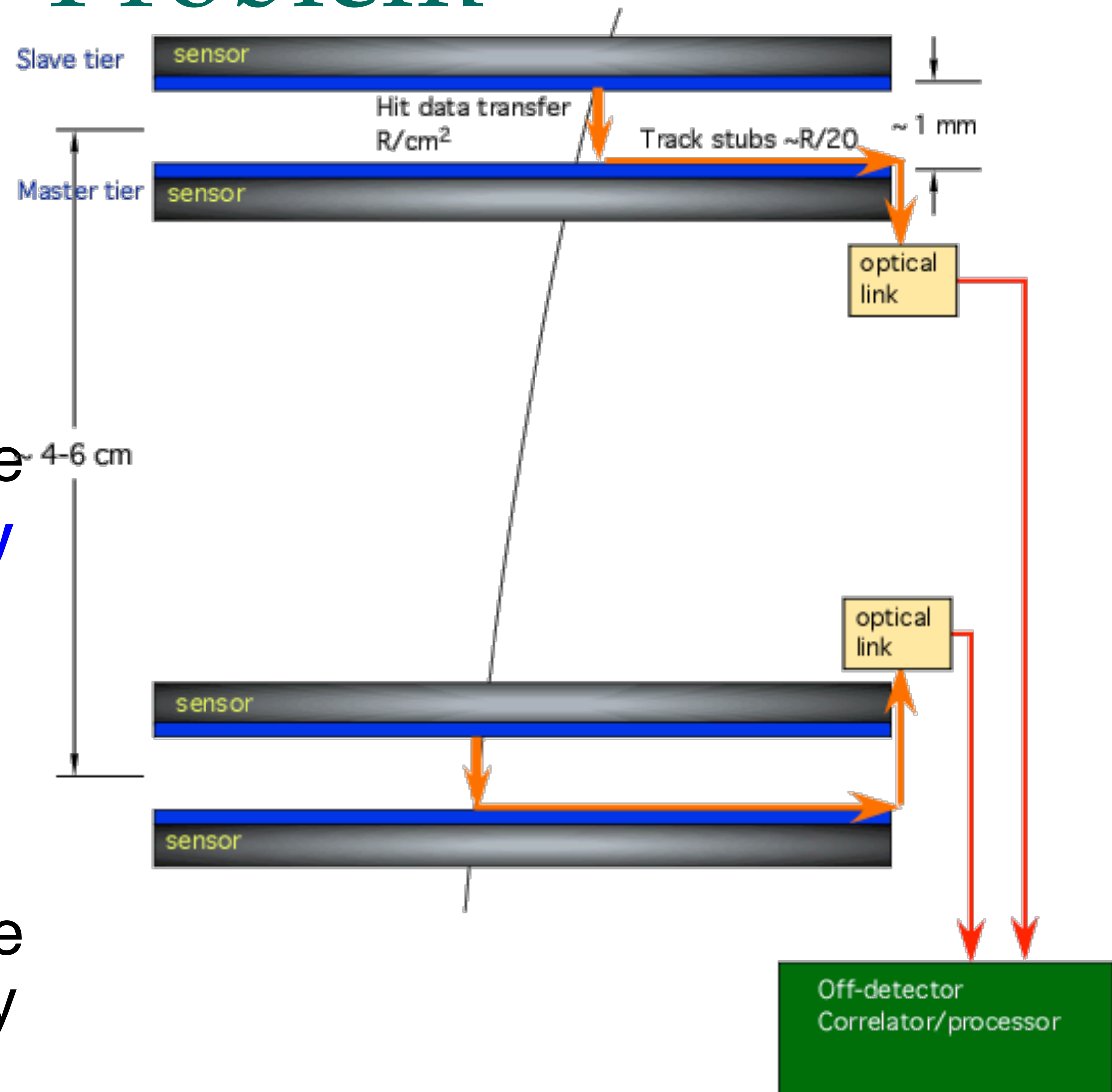
Fermilab



The ultimate physics reach of LHC will critically depend on the ultimate tracking trigger capabilities of its experiments

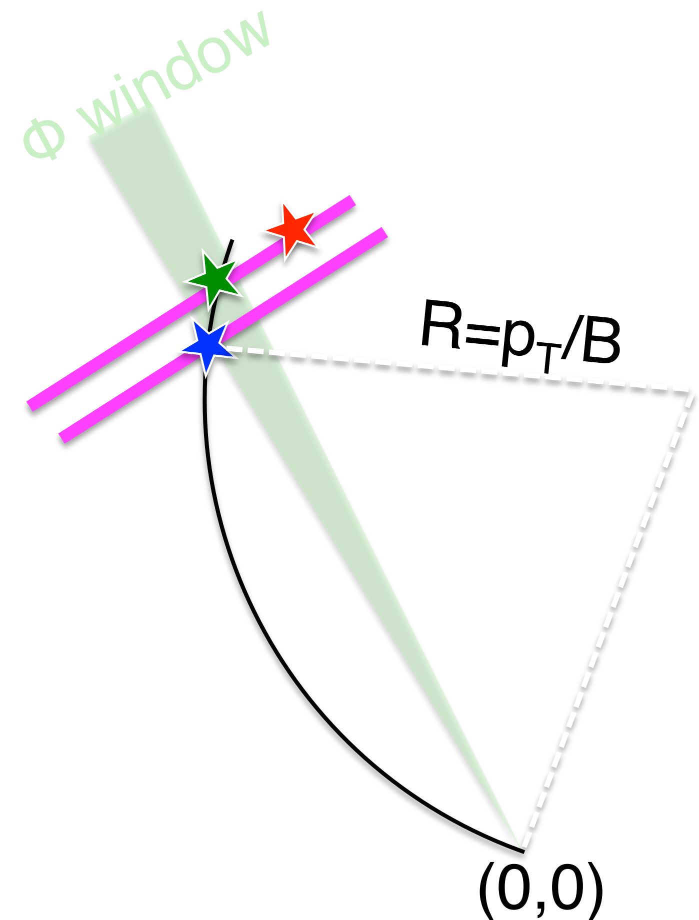
The Track Trigger Problem

- Need to gather information from 10^8 pixels in $200 m^2$ of silicon at 40 MHz
- Power & bandwidth to send all data off-detector is prohibitive
  - **Local filtering necessary**
  - **Smart pixels needed to locally correlate hit P_t information**
- Studying the use of 3D electronics to provide ability to locally correlate hits between two closely spaced layers



One Idea for CMS tracker design

- At 200 interactions/ crossing
  - 3×10^{13} bits/second
  - Too much data to move off detectors
- Trigger on p_T by looking for pointing coincidences in planes separated by ~1mm
  - Infinite $p_T \Rightarrow 90°$
  - 2 GeV/c $\Rightarrow 83°$

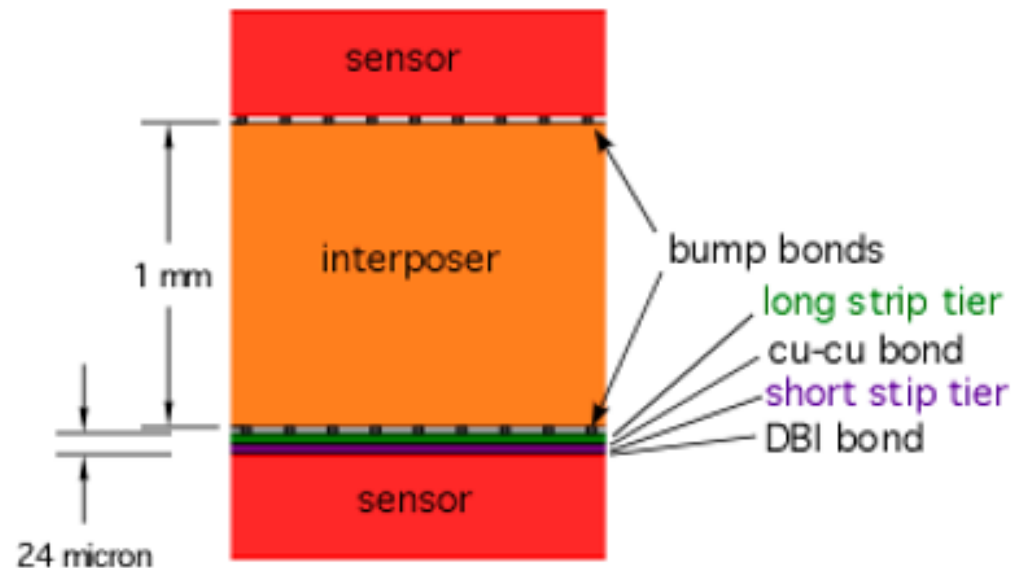


Push only data of interest off detectors

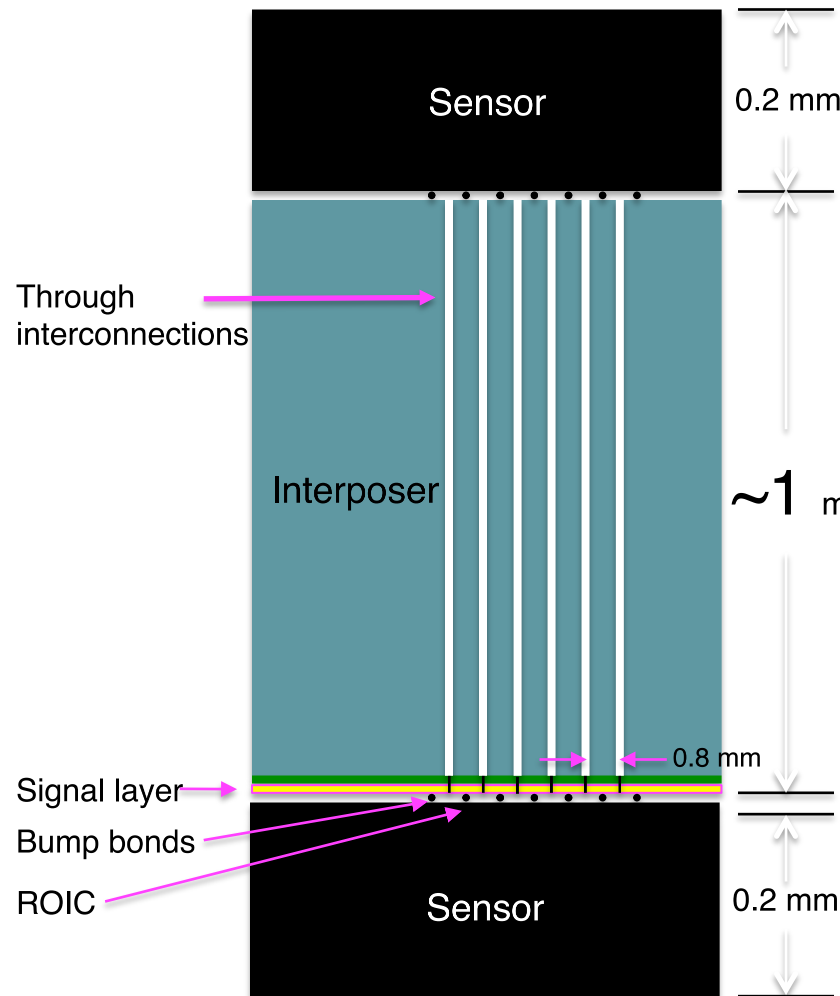
3D Interconnection

Key to design is ability of a single IC to connect to both top & bottom sensor

- Enabled by “vertical interconnected” (3D) technology
- A single chip on bottom tier can connect to both top and bottom sensors – locally correlate information
- Analog information from top sensor is passed to ROIC (readout IC) through interposer



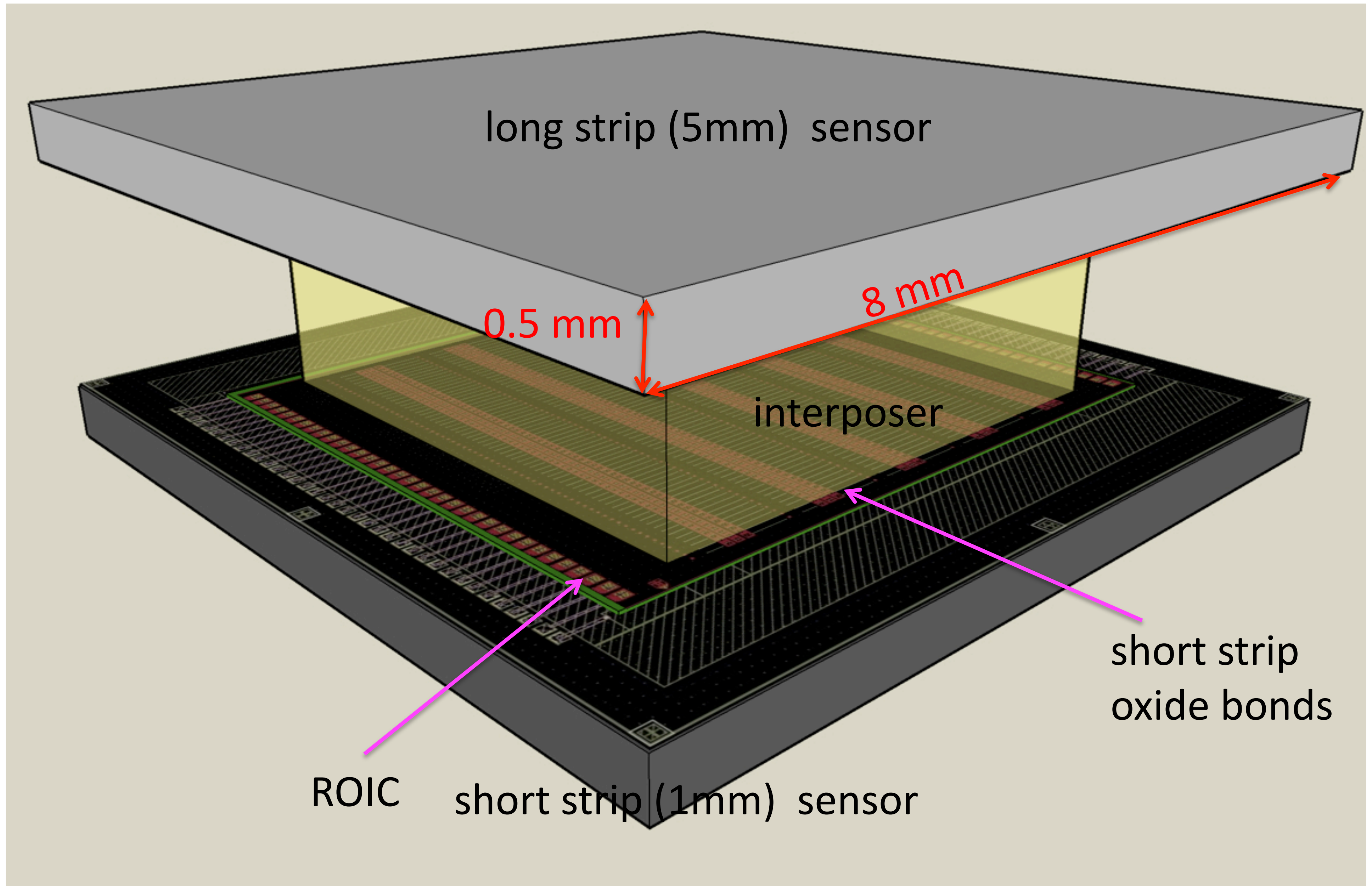
Stack Details



- Vertical information flow from outer to inner stack layers
- Readout chip (ROIC) connected to inner sensor
- Low mass interposer
  - transmits analog signals from upper sensor
  - bump bond connections
- Through Silicon Vias used to connect ROIC to bonding pads

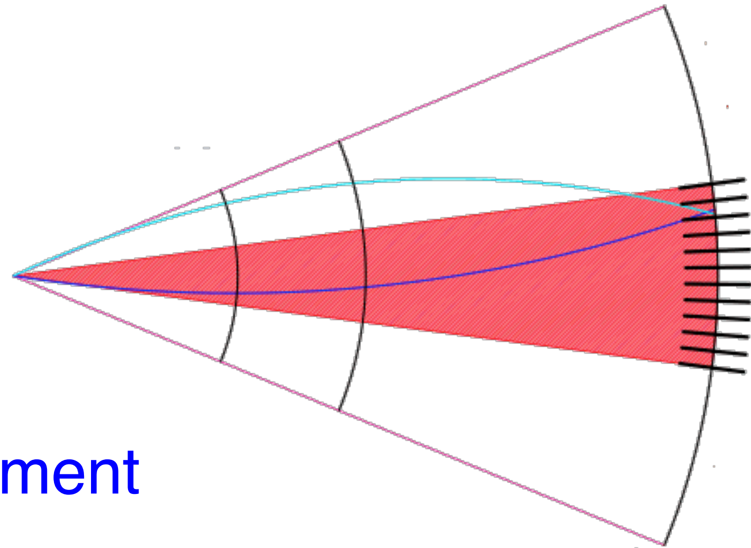
Lenny Spiegel's slides from TIPP 2011

Demonstration Model



Trigger Considerations

- 15° sectors
- 3 layers
- pT min ≈2.5 GeV



Conceptual development

Once the data are transferred out, one can then format the data and perform pattern recognition. One possible approach is to use the associate memory …

Note: any new silicon tracker has to be carefully designed for triggering.

Current ATLAS/CMS silicon-based tracker were not designed for triggering.

CDF Silicon detector was design for triggering

Fermilab

3D Technology in 30 seconds

- 3D technology: the integration of thinned and bonded silicon integrated circuits with vertical interconnects between IC layers
  - Vertical interconnects: Through-silicon-vias (TSVs)
  - Applications: memories, pixel arrays, microprocessors & FPGAs
- Performance can be improved by reducing interconnect R/L/C for higher speed and density…
- Freedom to divide functionality among tiers to create new designs that are simply not possible in 2D
  - Useful when a task can be partitioned into multiple sections that are physically and logically separable, and the interconnects among them are straightforward

Moore's law is approaching severe limitations

3D could be the next scaling engine

Not just as merely an extension of Moore's law,

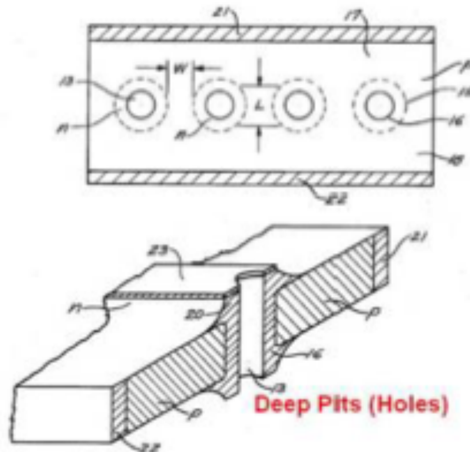
also provides novel design opportunities

Through Silicon Via (TSV)

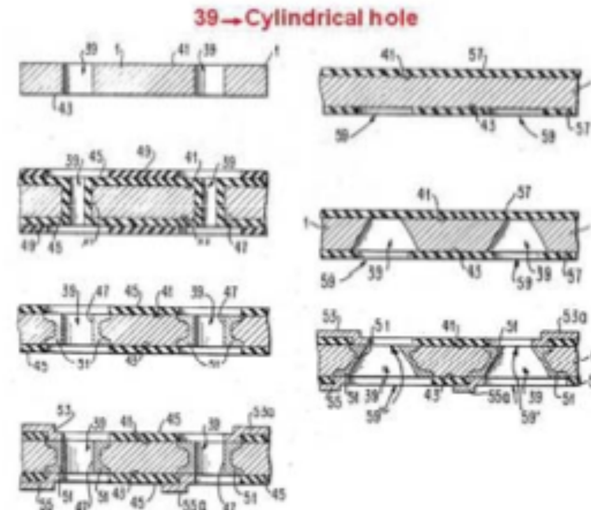
A Solution Without a Problem for… Half of a Century

W. Shockley…

…and M. Smith & E. Stern



U.S. Patent # 3,044,909: "Semiconductive Wafer and Method of Making the Same", 1958



U.S. Patent #: 3,343,256 "Methods of Making Thru-Connections in Semiconductor Wafers", 1964

CONFIDENTIAL 2
© Synopsys 2010
SYNOPSYS
Predictable Success

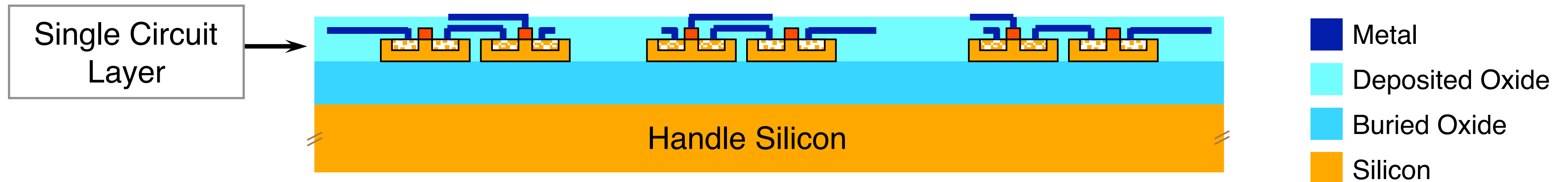
Examples of commercial applications of 3D Technology

- Increase density dramatically
  - Example: 3D DRAM stacking (control/interface tier + memory cell tiers)
  - Footprint or size reduction has been the main driving factor
  - Available commercially in embedded, wireless, and memory devices
- Increase memory access bandwidth dramatically
  - 3D integration of memory layers onto processor chip
  - Eliminate the slower and higher-power off-chip buses (tens of ~ mm) by replacing them with high-bandwidth and low-latency short vertical interconnections (~ tens of um)
  - Potential to remove some "fundamental bottlenecks" in computing

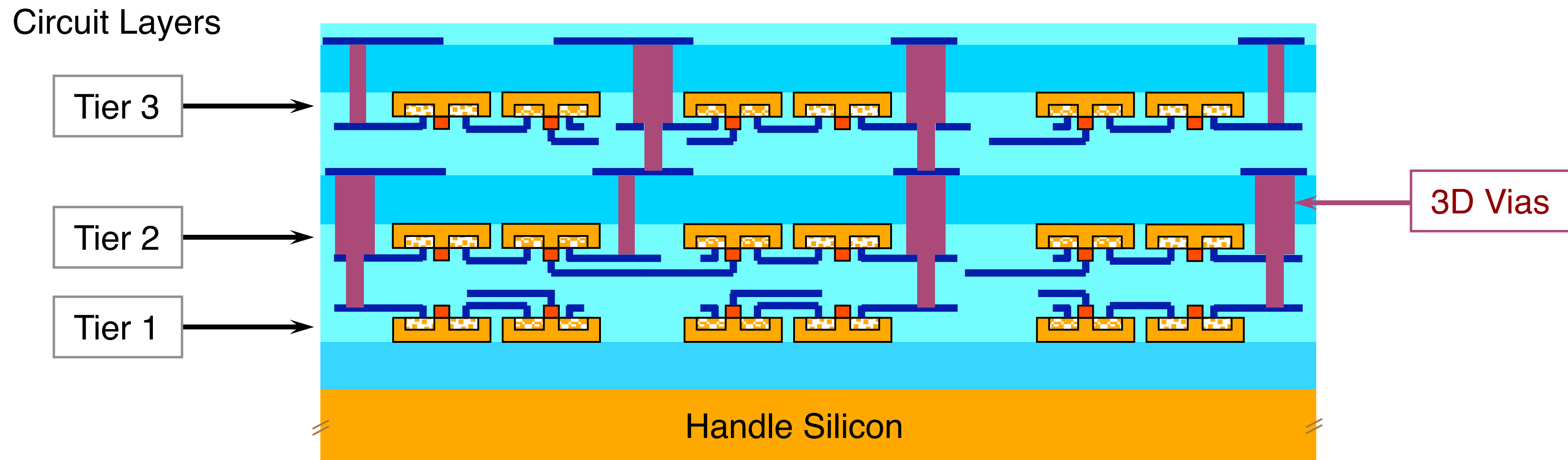
Both examples are relevant to AM R&D in 3D (see later)

Routing in 3D can be efficient, esp. if functional elements are arranged such that the interconnects among tiers are mostly vertical

2D versus 3D Circuits



2D Integrated Circuit Cross-Section



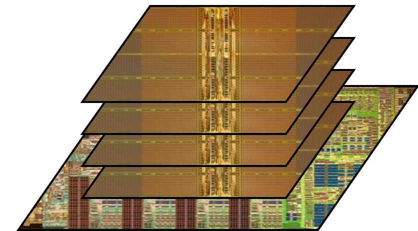
3D Integrated Circuit Cross-Section

Fermilab

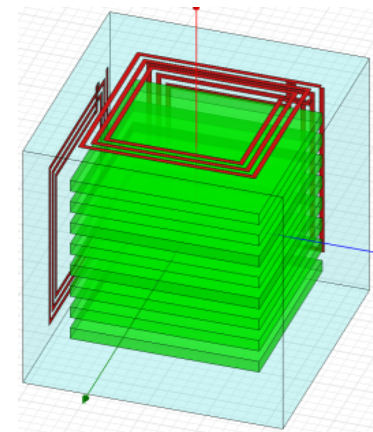
3DIC Value Propositions

Fundamentally, 3DIC permits:

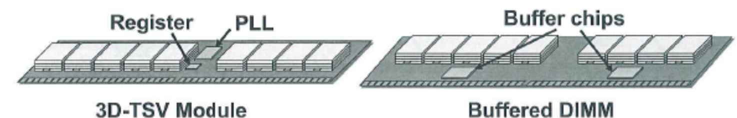
- Shorter wires
  - consuming less power, and costing less
  - The memory interface is the biggest source of large wire bundles



- Heterogeneous integration
  - Each layer is different!
  - Giving fundamental performance and cost advantages, particularly if high interconnectivity is advantageous



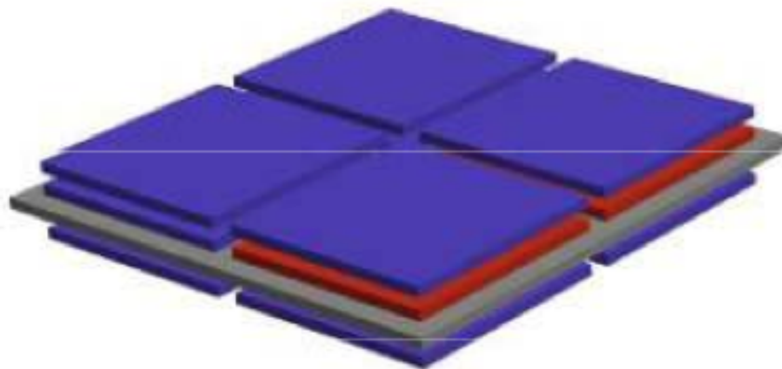
- Consolidated "super chips"
  - Reducing packaging overhead
  - Enabling integrated microsystems

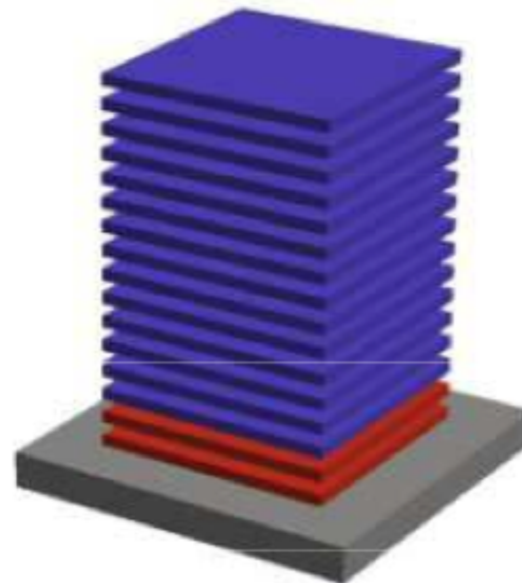


Many Flavors Of “> 2D” IC Integration

Designer’s Imagination Is the Only Limit…

- Homogeneous vs. heterogeneous
- Side-by-side vs. stack… vs. combinations
- Face-to-face, face-to-back,…
- Passive vs. active interposer
- Single-sided vs. double-sided





CONFIDENTIAL 31
© Synopsys 2010

SYNOPSYS® Predictable Success

Many Flavors Of “> 2D”…

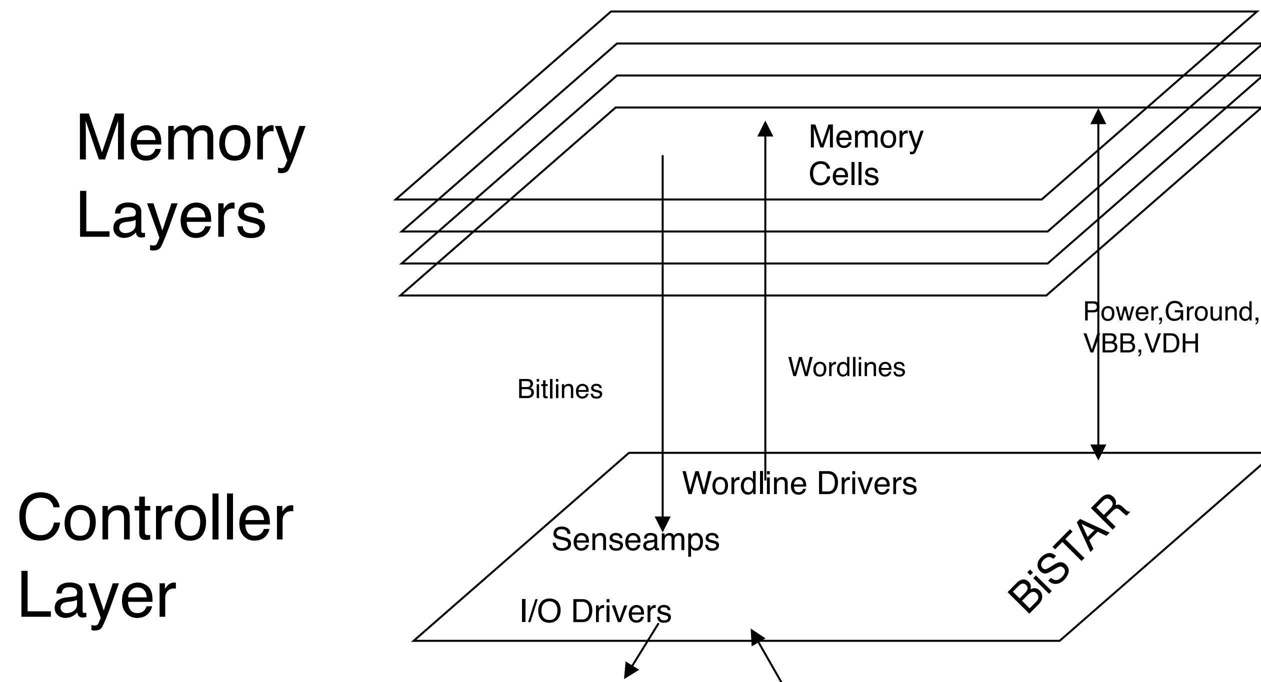
Marketing Imagination Is the Only Limit…

CONFIDENTIAL 30
© Synopsys 2010

Source: www.bk.com 2010

SYNOPSYS® Predictable Success

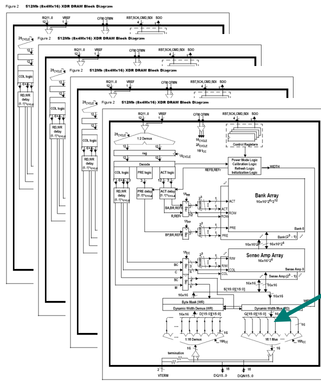
“Dis-Integrated” 3D Memory

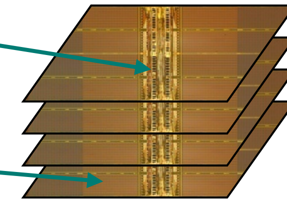


Tezzaron Memory

Memory on Logic

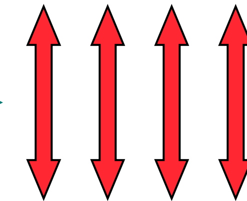
Conventional | TSV Enabled





Less Overhead

Flexible bank access



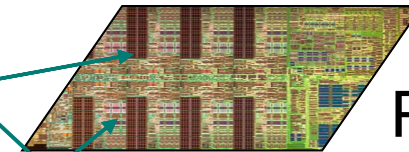
x32 to x128

Less interface power

3.2 GHz @ >10 pJ/bit

→ 1 GHz @ 0.3 pJ/bit

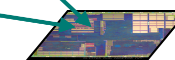
N x 128 "wide I/O"



Processor

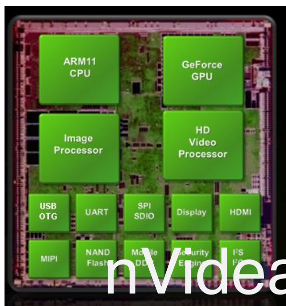
Flexible architecture

or

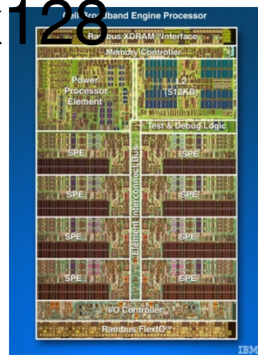


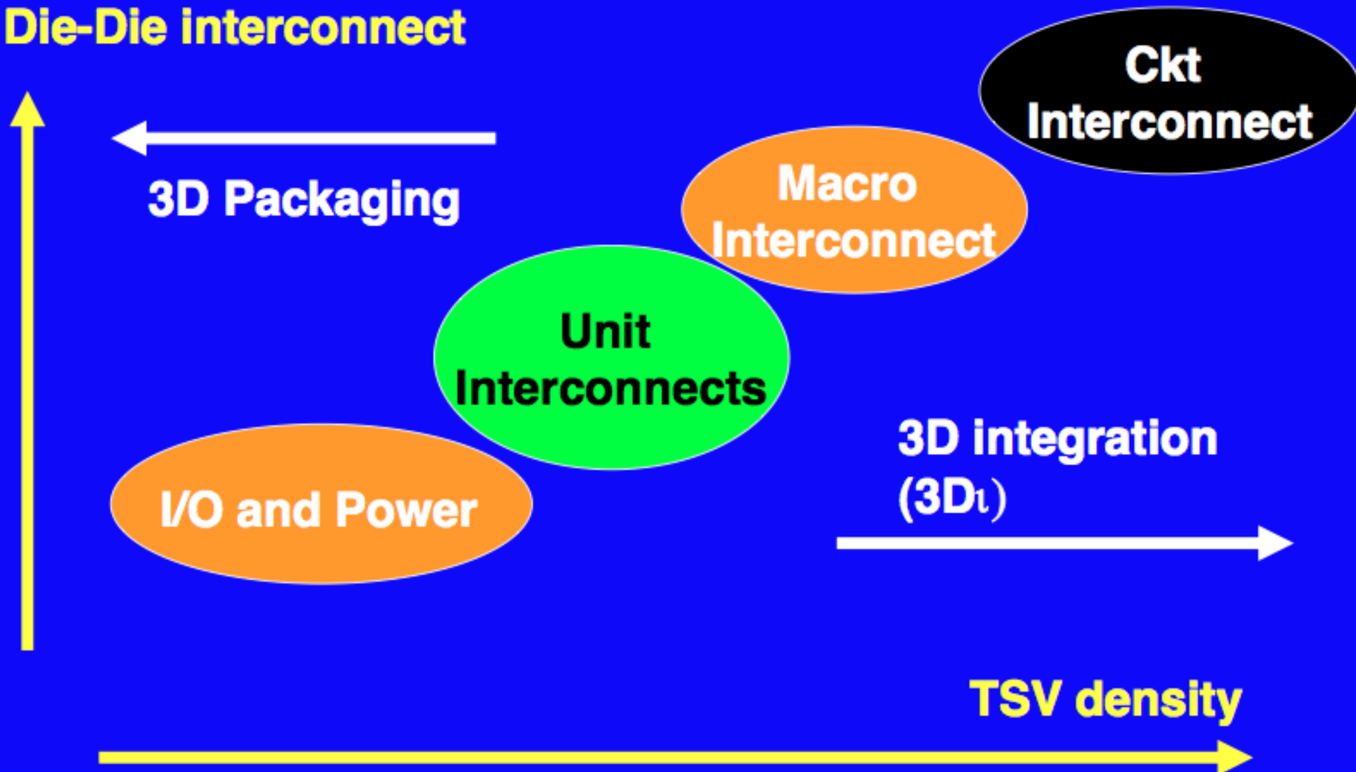
Mobile

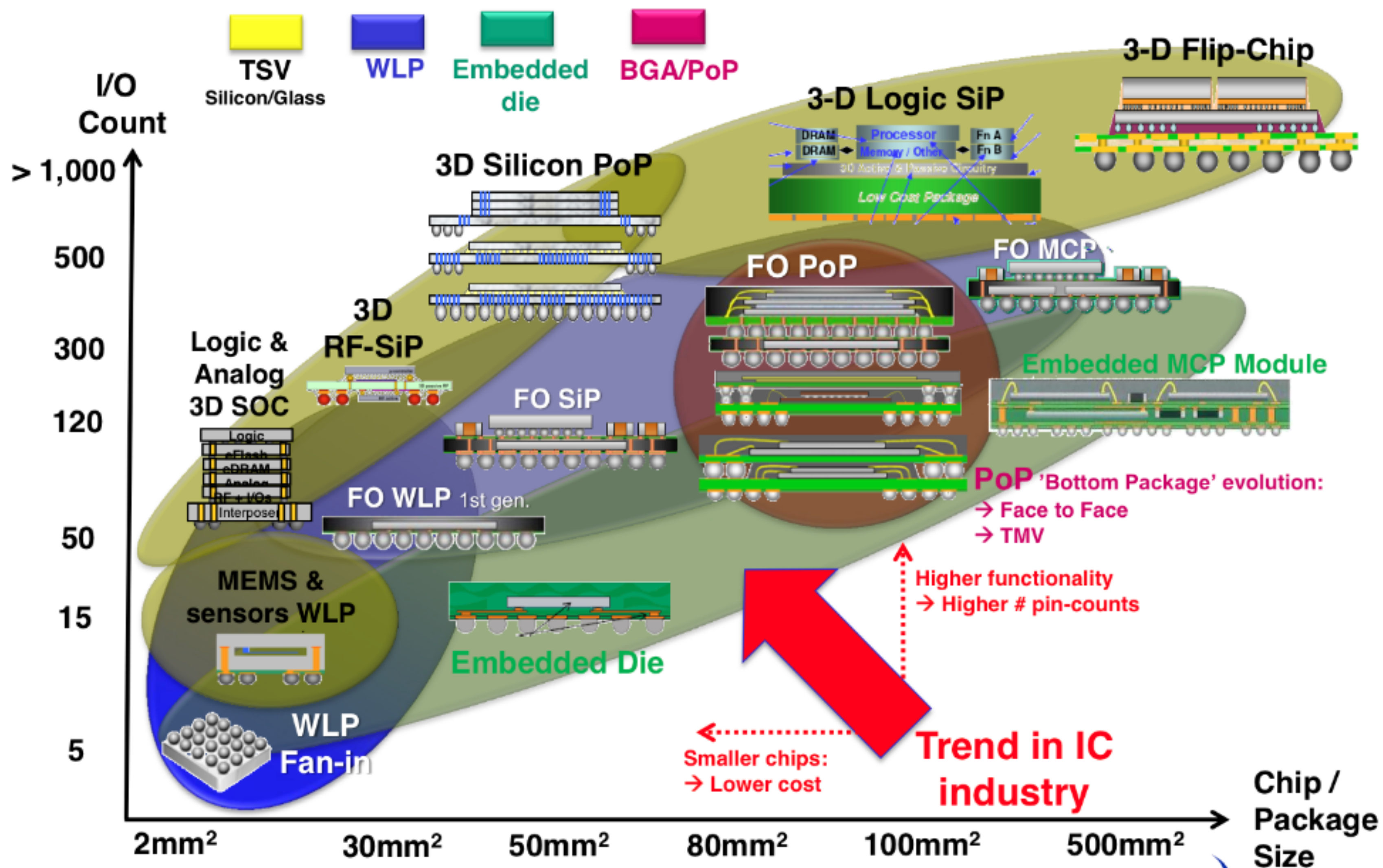
Short on-chip wires



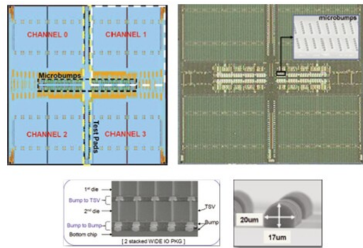
or

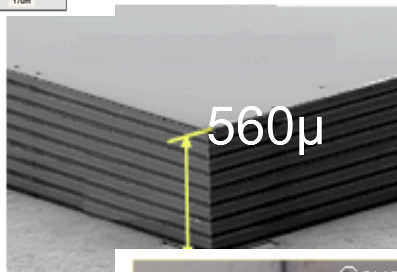


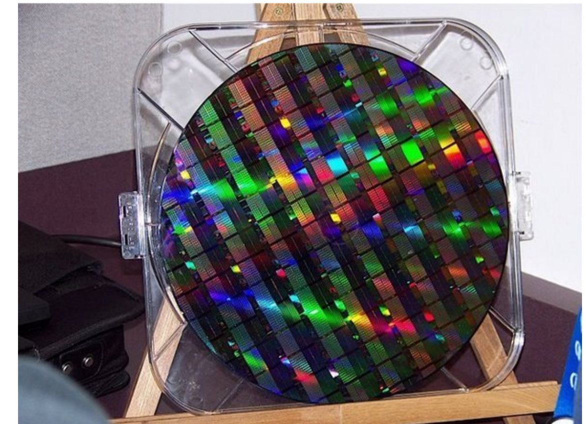

3D Packaging Vs Integration
Die-Die interconnect
3D Packaging
Ckt
Interconnect
Macro
Interconnect
Unit
Interconnects
I/O and Power
3D integration
(3Dι)
TSV density
3D ASIP conference December 2010 (S.S. Iyer)


Current / Future Tool-Box for 3D Packaging
TSV
Silicon/Glass
WLP
Embedded die
BGA/PoP
I/O Count
> 1,000
500
300
120
50
15
5
3-D Flip-Chip
3-D Logic SiP
3D Silicon PoP
FO PoP
FO MCP
3D RF-SiP
Logic & Analog 3D SOC
FO SiP
Embedded MCP Module
FO WLP 1st gen.
PoP 'Bottom Package' evolution:
→ Face to Face
→ TMV
MEMS & sensors WLP
Embedded Die
Higher functionality
→ Higher # pin-counts
WLP Fan-in
Smaller chips:
→ Lower cost
Trend in IC industry
Chip / Package Size
2mm²
30mm²
50mm²
80mm²
100mm²
500mm²

How Real is 3D?



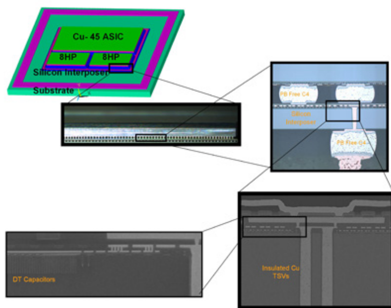


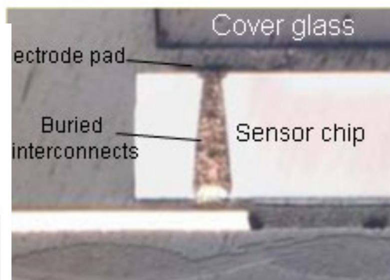


Samsung
16Gb NAND flash (2Gx8 chips), Wide Bus DRAM

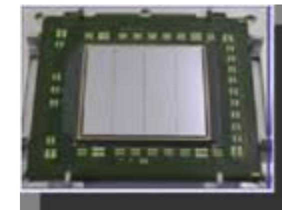
Micron
Wide Bus DRAM

Intel
CPU + memory

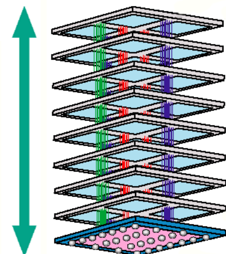


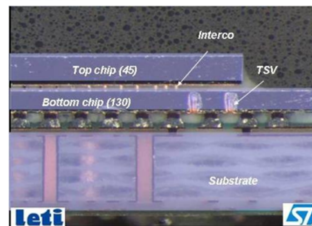


OKI
CMOS Sensor

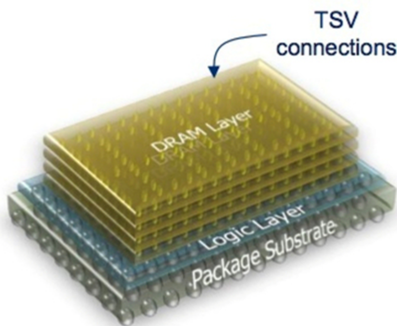


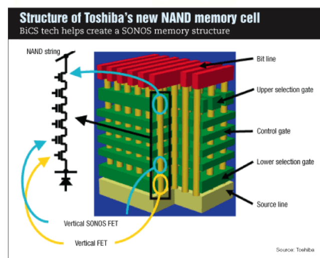
Xilinx
4 die 65nm interposer

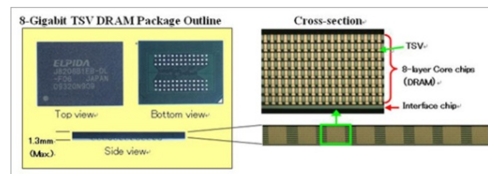




Raytheon/Ziptronix
PIN Detector Device





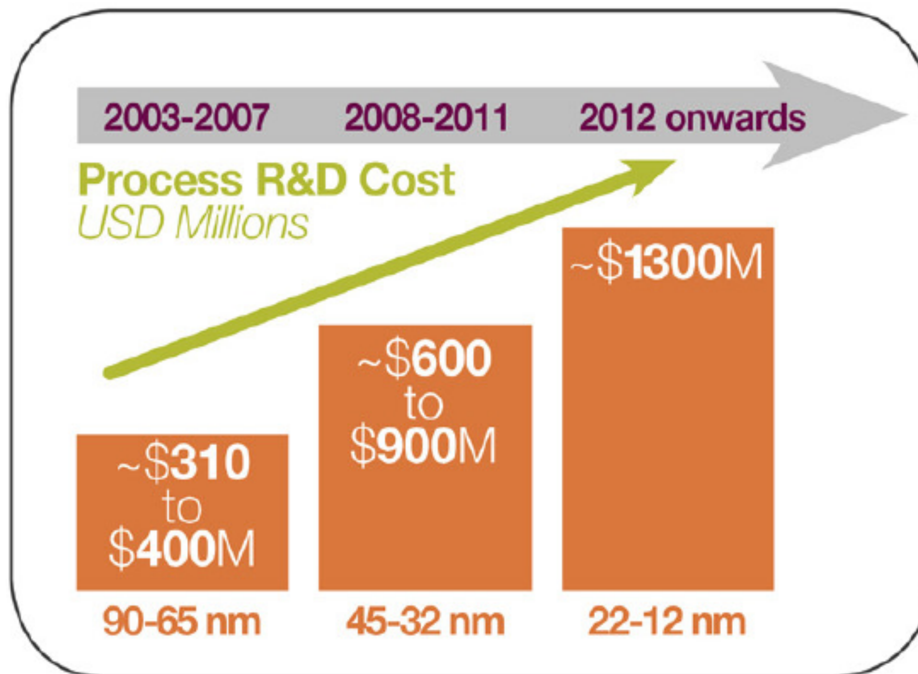


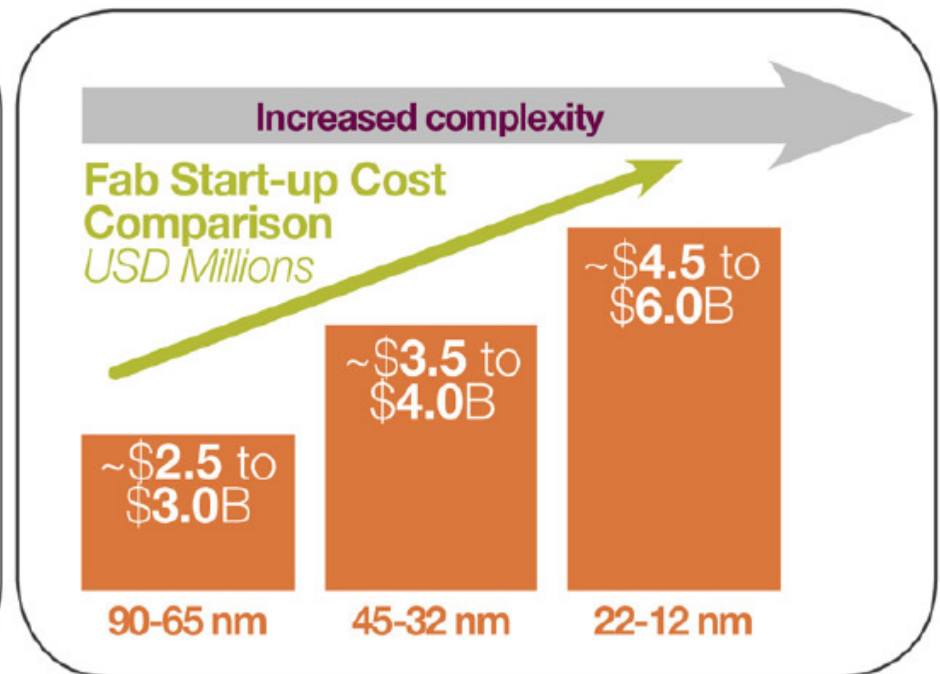
IBM
RF Silicon Circuit Board / TSV
Logic & Analog

Toshiba
3D NAND

The Dilemma of the Semiconductor Industry

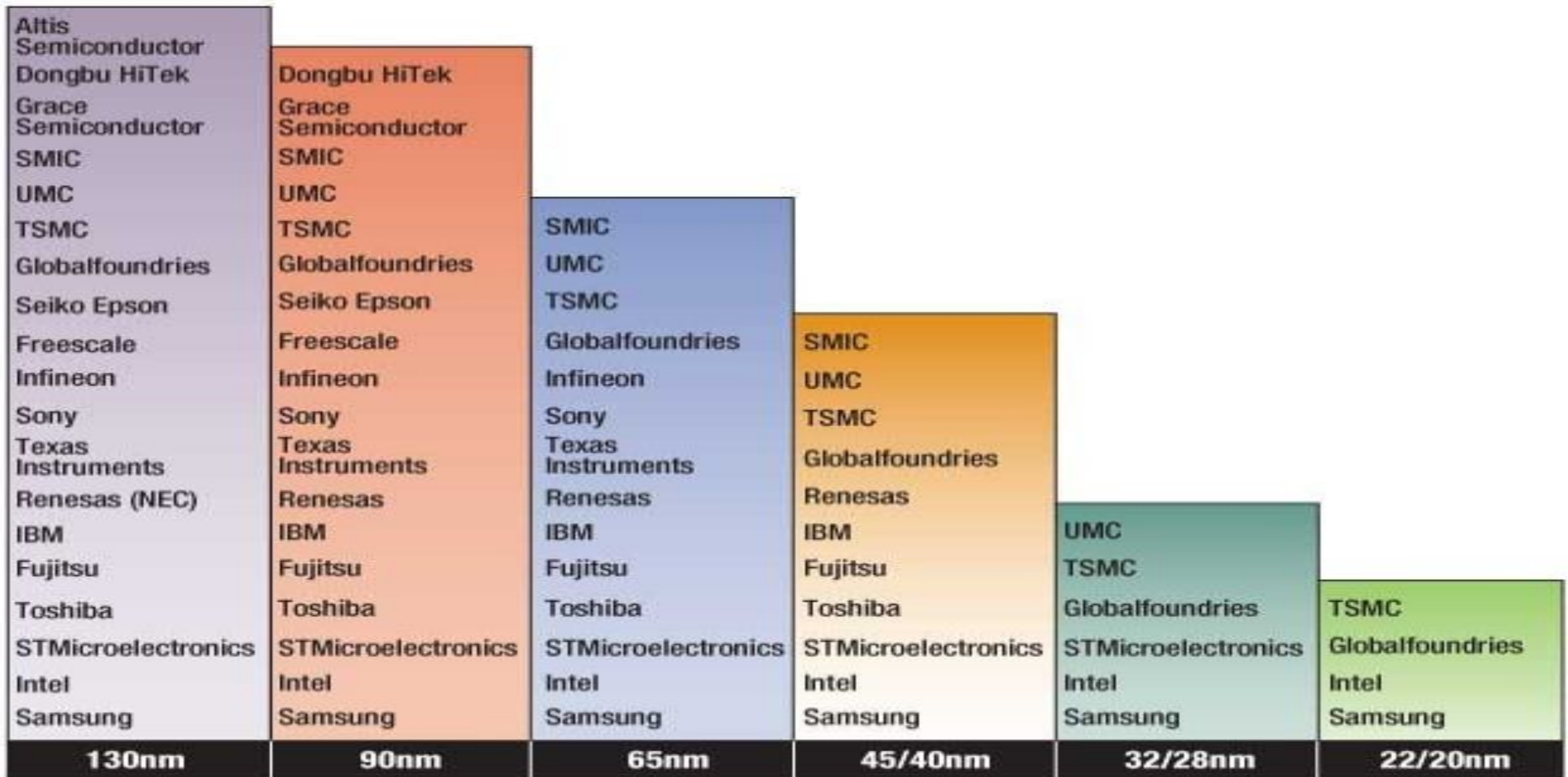
- Chip-makers need to keep pace with technology and focus on design
- …while chip manufacturing and technology R&D continue to grow in cost and complexity



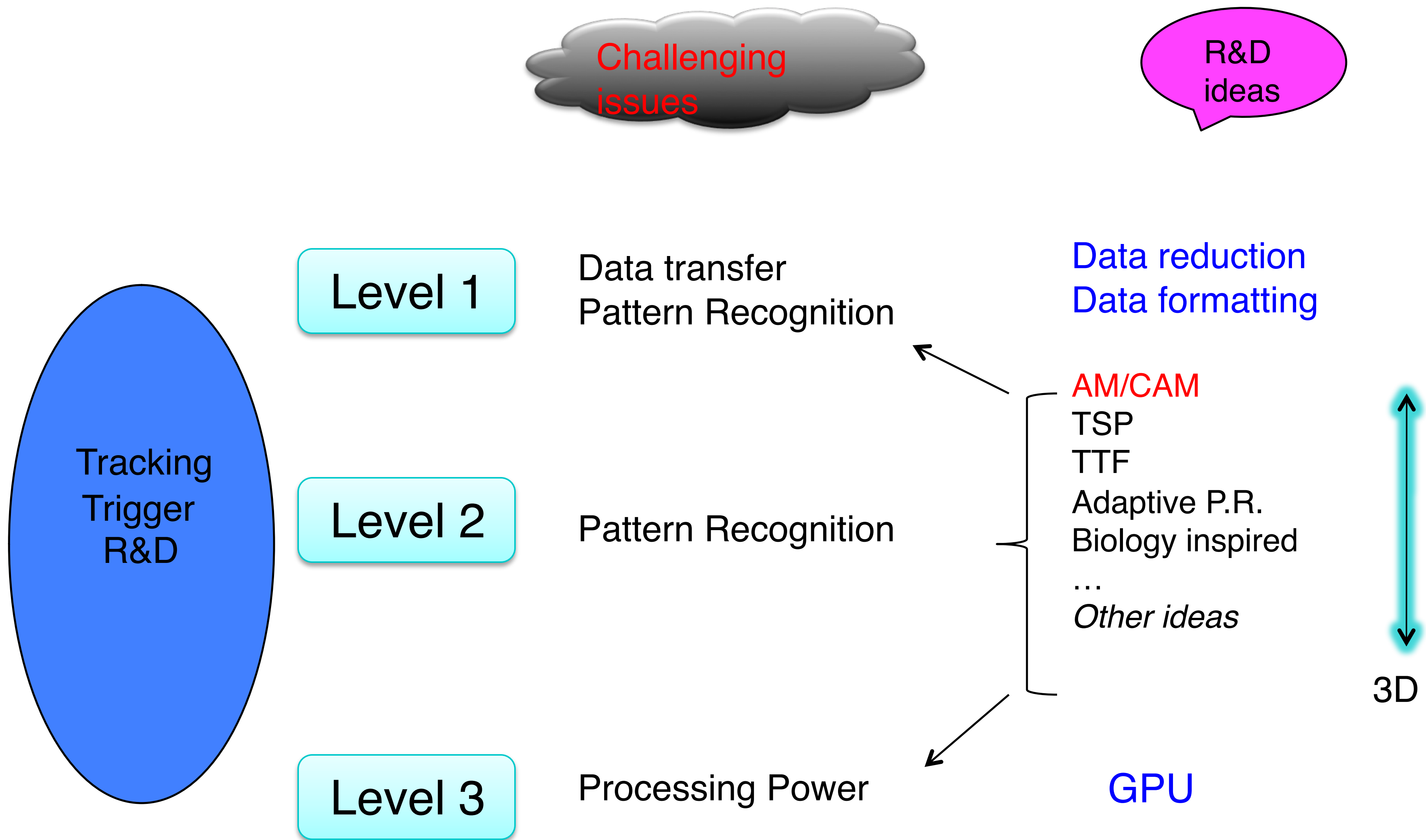


Courtesy: GlobalFoundries

Severe Reduction in Number of Fabs



(Source: IHS iSuppli)

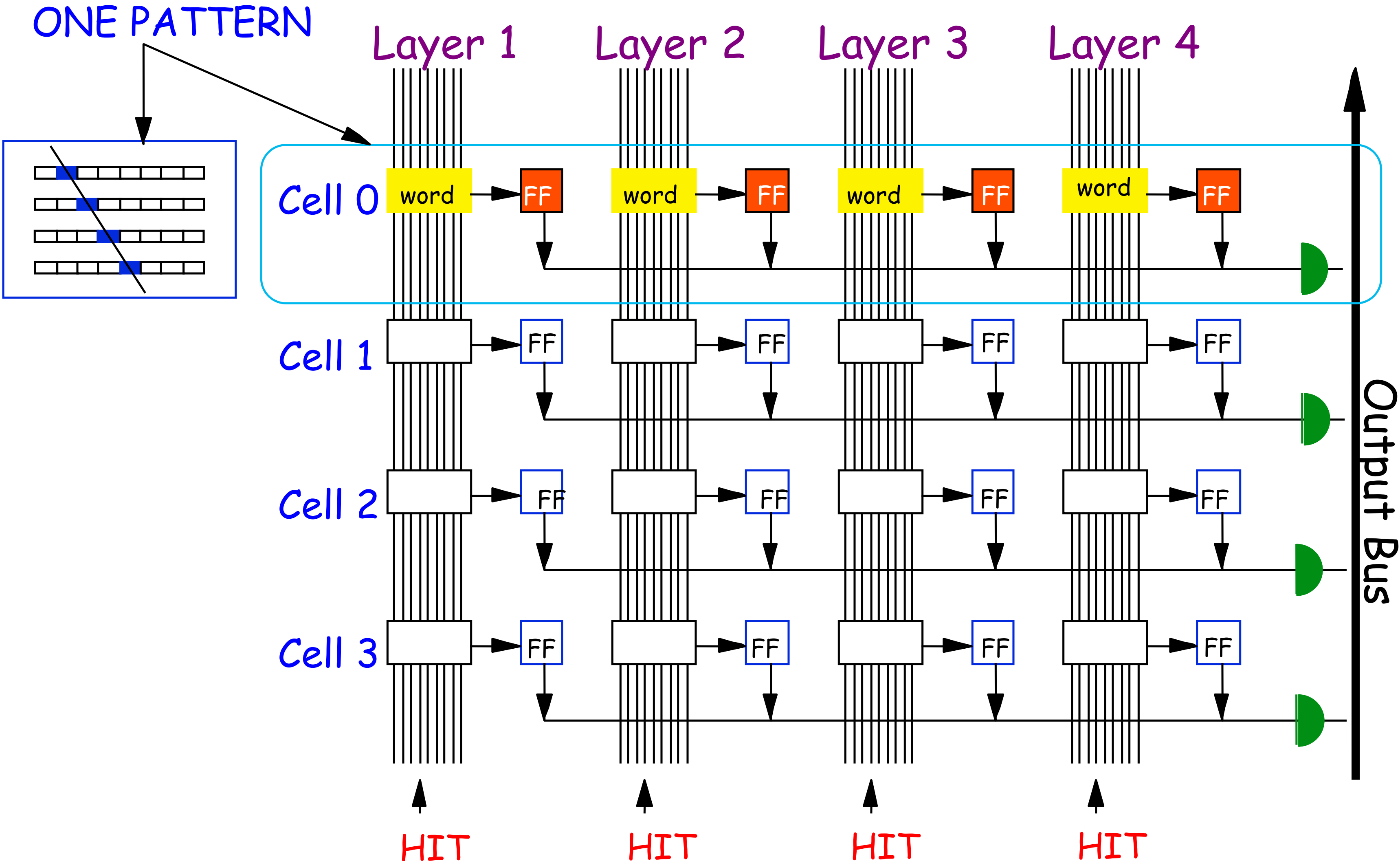


The ultimate physics reach of LHC will critically depend on the ultimate tracking trigger capabilities of its experiments

Will just use AM (Associate Memory) R&D as an example in this lecture

Fermilab

Back to the basics of Associative Memory



AM State-of-the-art

- For HEP, the best AM is the AMchip03 from INFN
  - Designed for HEP in 2004 (CDF/SVT, ~ 10^3 chips needed for the system)
  - 180 nm technology (standard cell)
  - 5K patterns for 6 tracking layers, ~16 bits per layer → 5K x 6 x 16
  - Or about 0.5 Million CAM bits in 1 cm^2 and runs at 40 MHz

- Commercial CAMs not designed for HEP

- The challenge for HEP: how to increase the AMchip patterns (&speed) significantly? > 100 in density, > 3 in speed
  - Optimization in 2D, for density, speed and power etc
  - On-going R&D effort with 90 nm now and 65nm later

Question: Can 3D technology help and how?

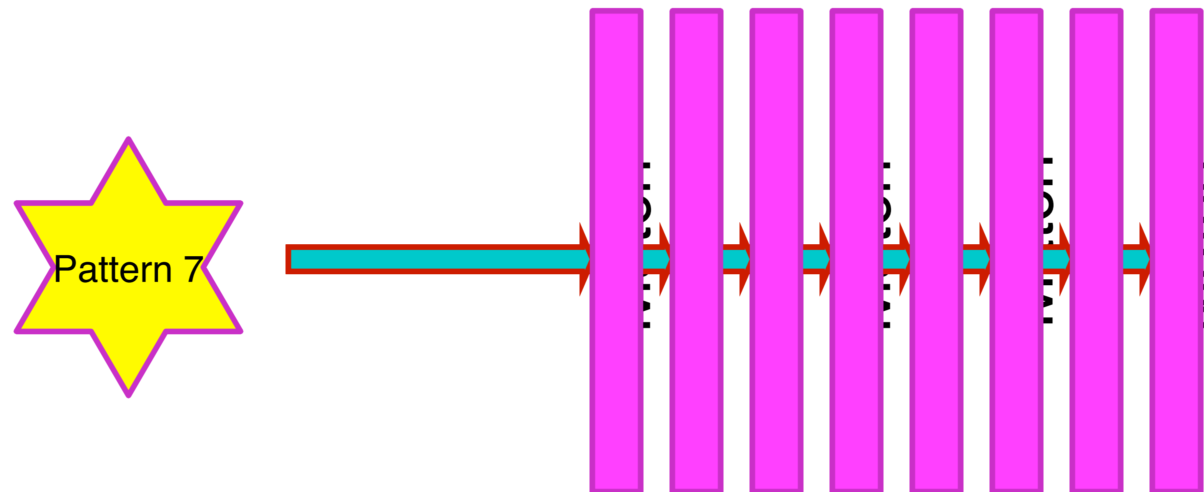
Fermilab

3D and Pattern Recognition

- To increase the AM pattern density
- The simplest approach: using 3D simply as stacking tool
  - To stack *n* AM chips together, to gain x *n* in pattern density
  - The KISS method: Keep It Simple Stupid
  - Example: 10 (optimized in 2D) x 4 (3D stacking) ~ 40 gain
  - Most likely limited by power/thermal issues
- *True 3D: can still keep it simple enough*
  - To revisit the fundamental architecture of AM
  - *To change it for more dramatic enhancement*
  - More flexibility to deal with power/thermal issues
  - More work involved
- To increase the bandwidth between AM and Track Fitting
  - Integration of AM and Track Fitting into one package/chip
  - AM+FPGA+DRAM+SRAM combo
  - “SVT in a chip”

How CAM works

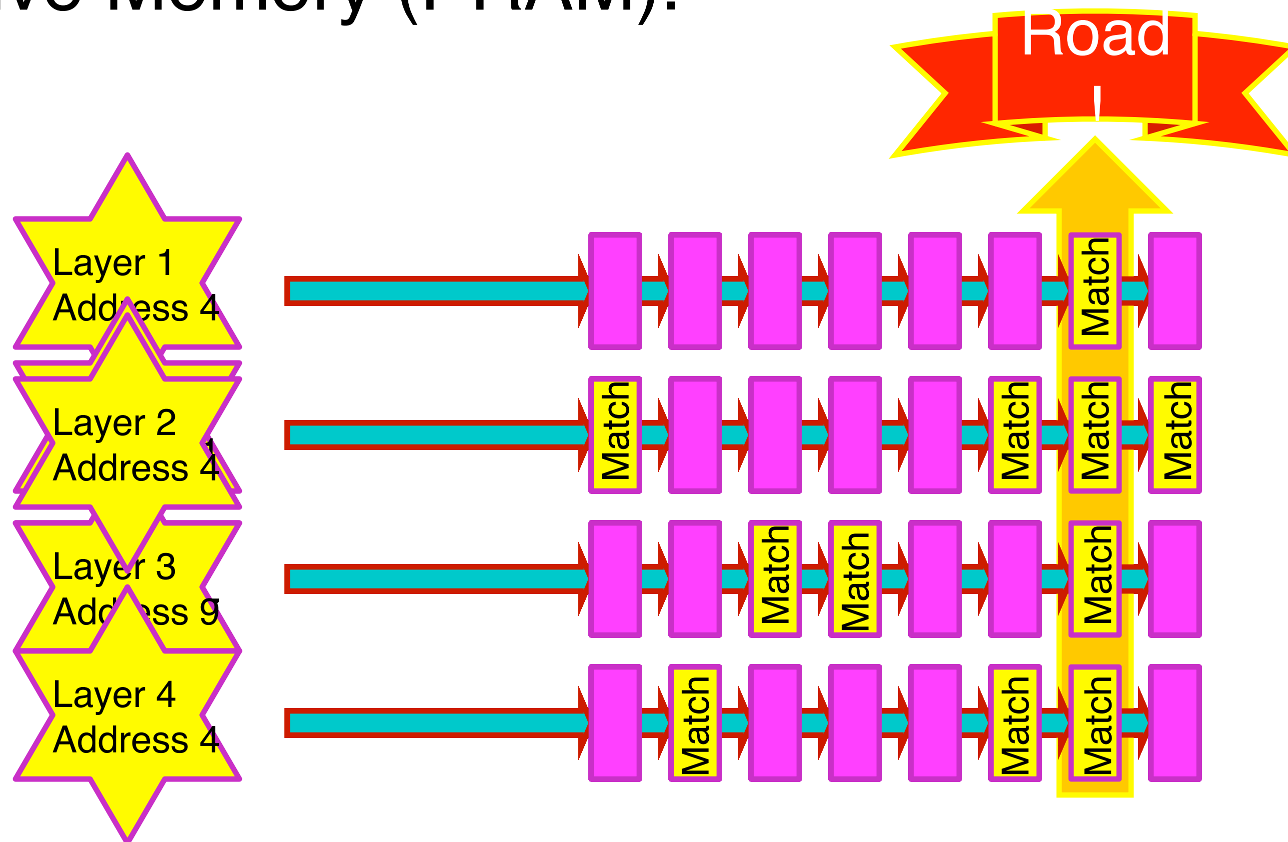
- A CAM (Content Addressable Memory) is a classical digital system building block



•One pattern at a time
•There is no memory of previous matches

How PRAM works

- A PRAM on the other hand is a Pattern Recognition Associative Memory (PRAM).



Jim Hoff's slides from TIPP 2011 (Trigger&DAQ session)

2D Implementation

(shown for 6 detector layers)

Each Column: A Classic CAM dedicated to ONE Detector Layer

The Extra "Stuff" that makes it a PRAM

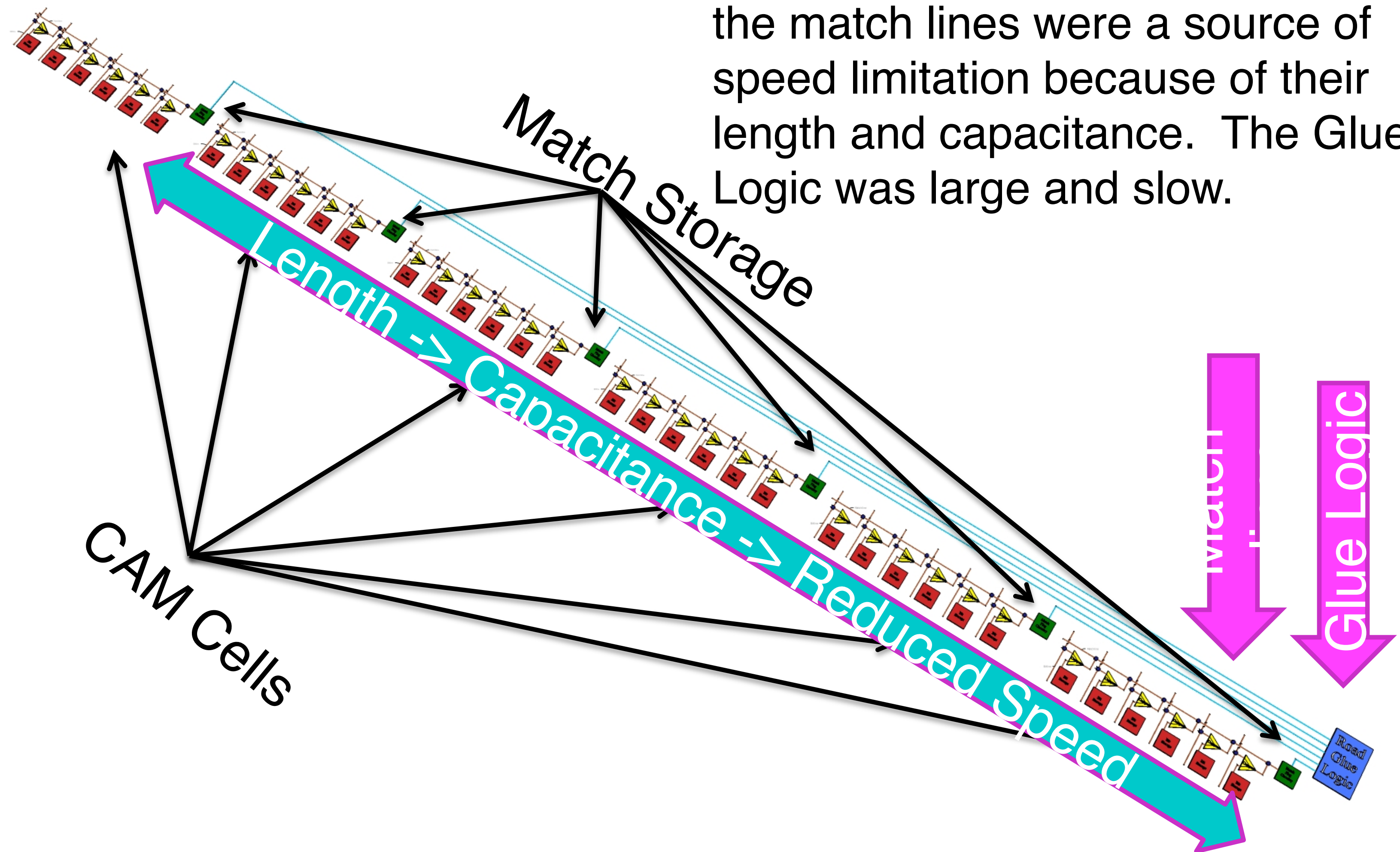
Each Row: One Full Road

CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic

A Single PRAM Cell

(in 2 dimensions)

In the older version of the AMchip, the match lines were a source of speed limitation because of their length and capacitance. The Glue Logic was large and slow.



The Challenge

To increase the patterns density by 2 orders of magnitude while increase the speed by more than a factor of 3

How can 3D help to improve the density & speed?

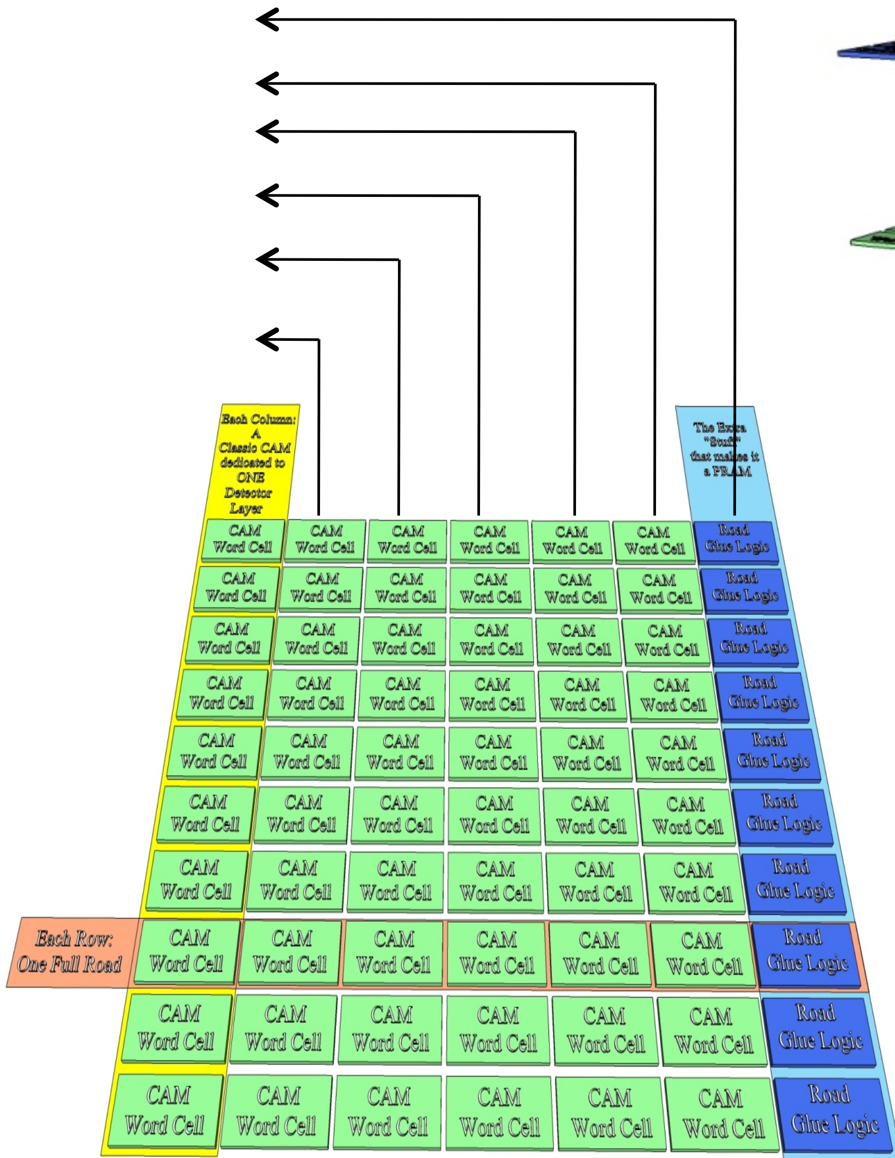
Each Column: A Classic CAM dedicated to ONE Detector Layer

The Extra "Stuff" that makes it a PRAM

Each Row: One Full Road

CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic
CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	CAM Word Cell	Road Glue Logic

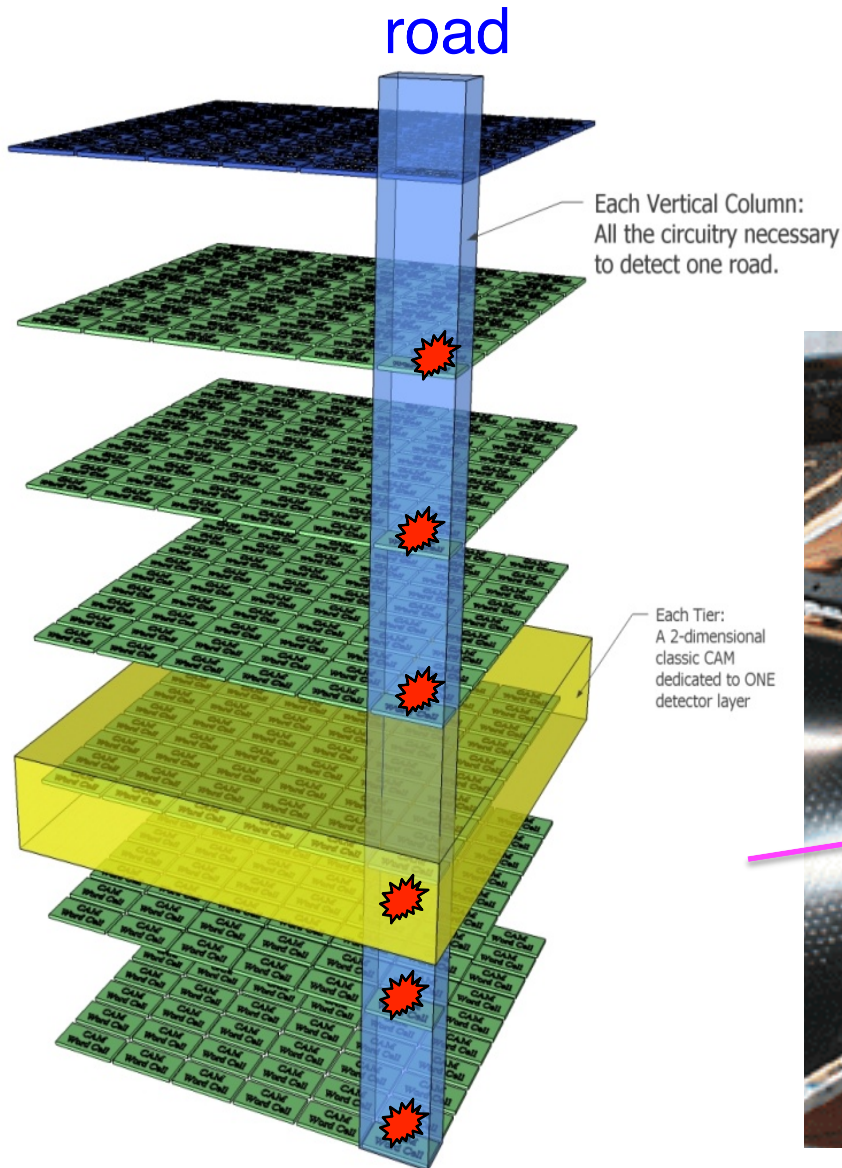
From 2D to 3D



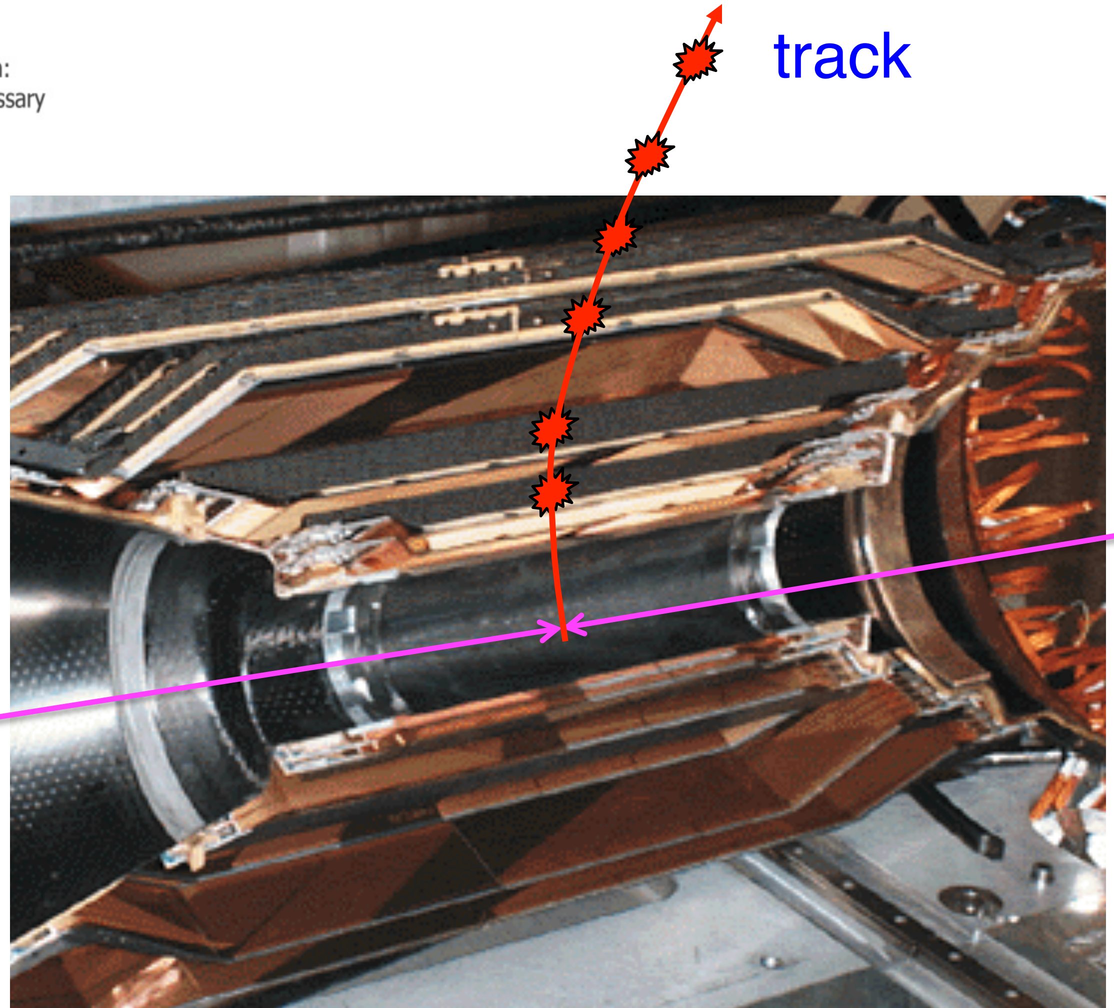
Each Vertical Column: All the circuitry necessary to detect one road.

Each Tier: A 2-dimensional classic CAM dedicated to ONE detector layer

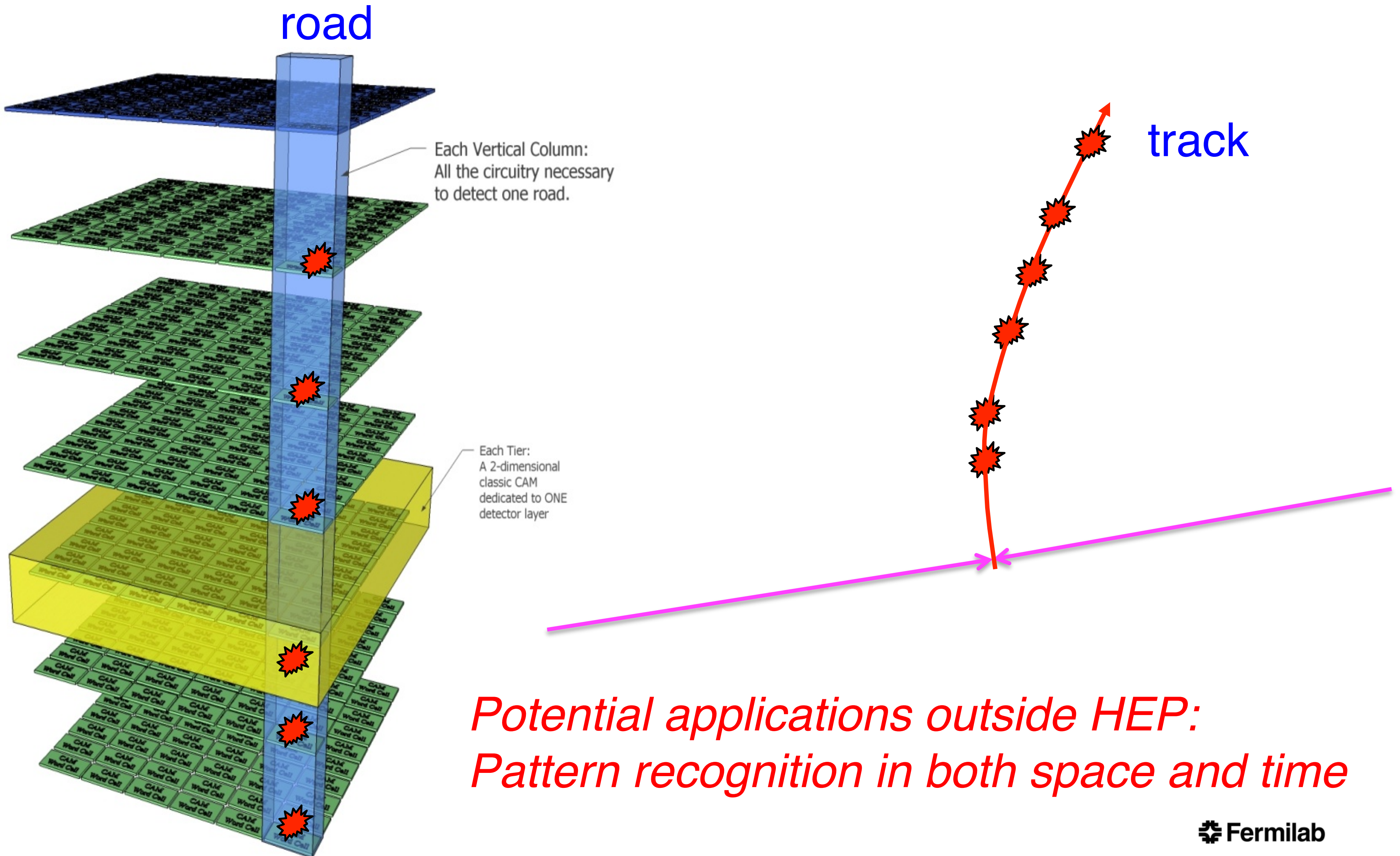
VIPRAM
(Vertically Integrated Pattern Recognition Associative Memory)



Pattern recognition for tracking is naturally a task in 3D

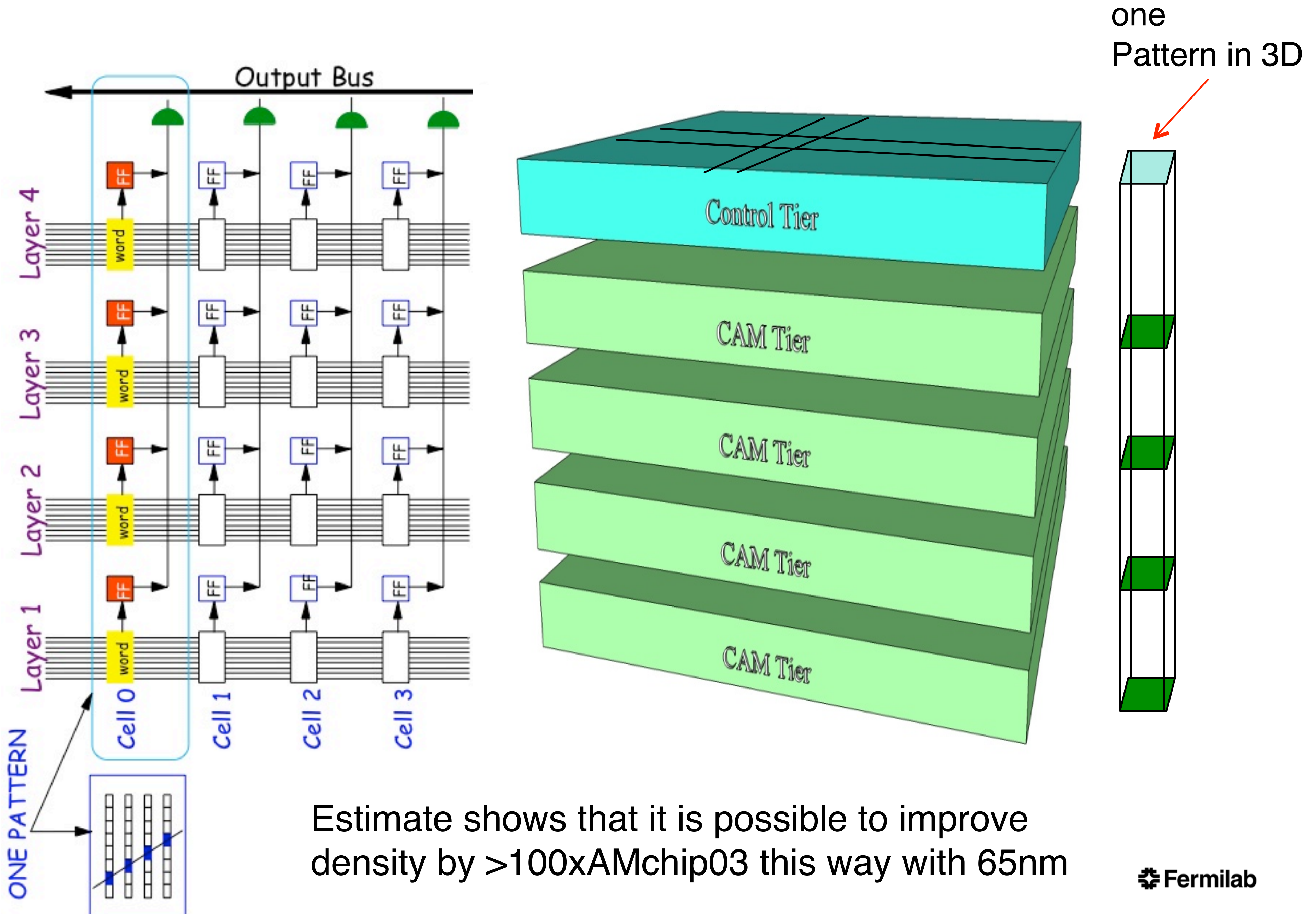


Advantages of VIPRAM Architecture



Potential applications outside HEP: Pattern recognition in both space and time

3D VIPRAM architecture

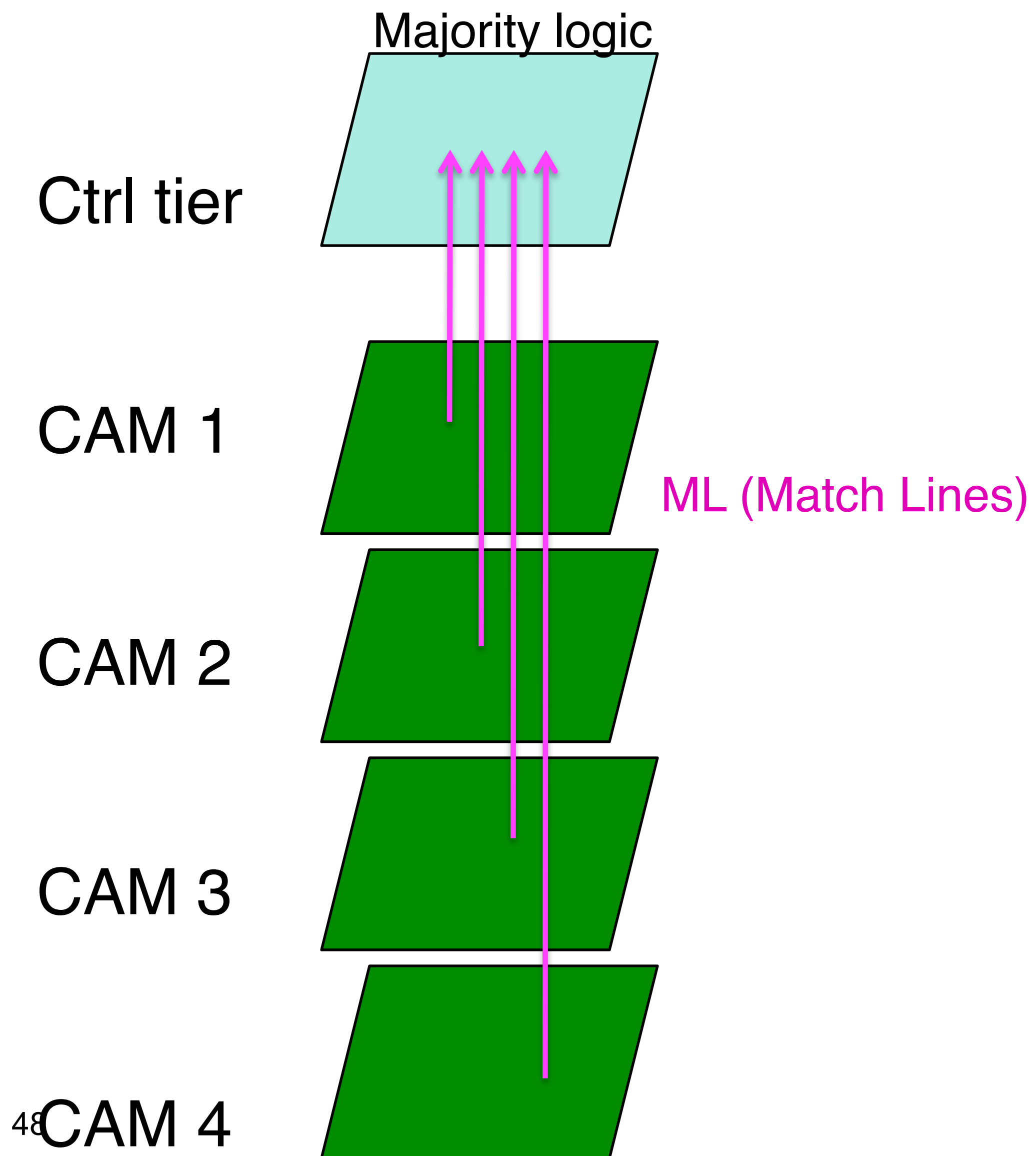


Estimate shows that it is possible to improve density by >100xAMchip03 this way with 65nm

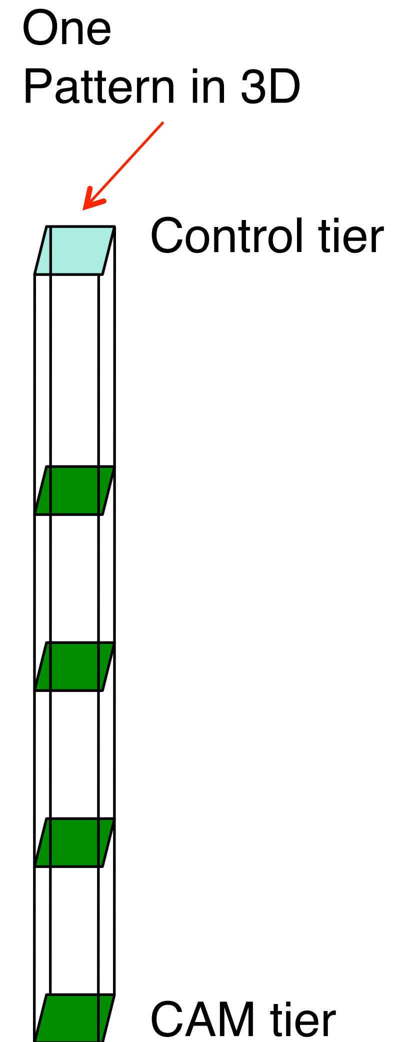
Fermilab

One key issue:

How to communication between the control and each CAM tier, given that the CAM tiers are physically identical?

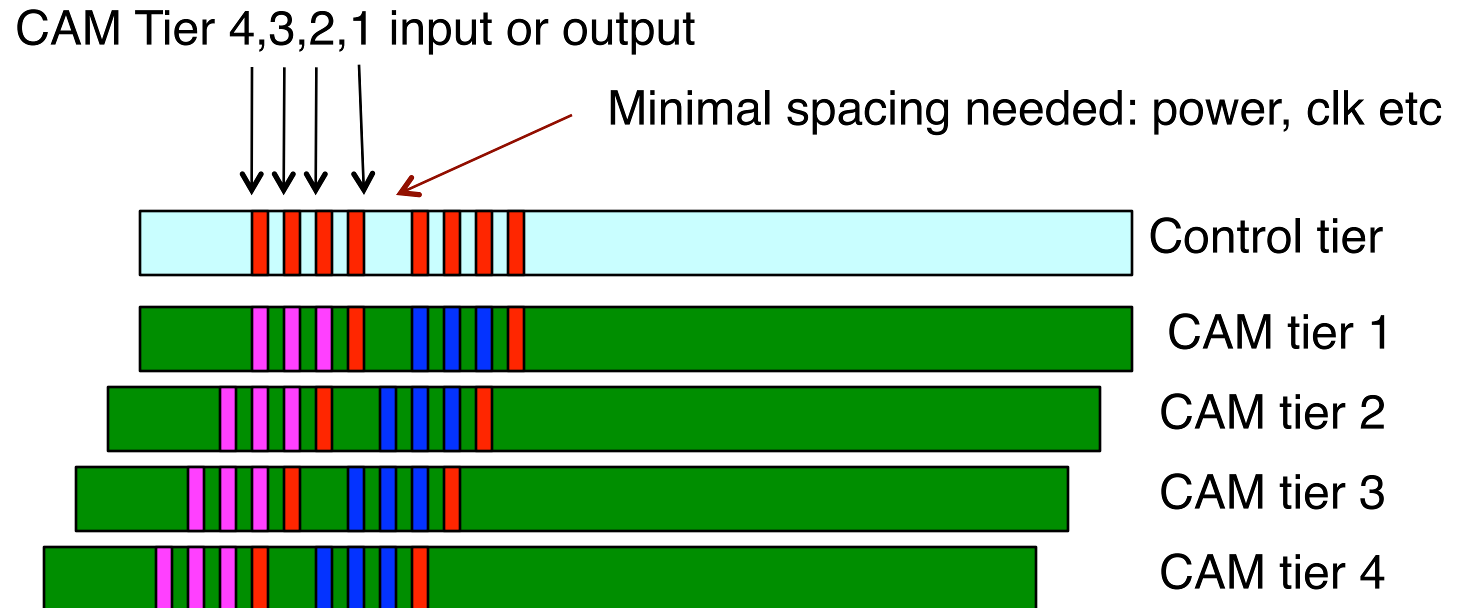


ML (Match Lines)



Fermilab

Offset stacking idea– is this feasible?

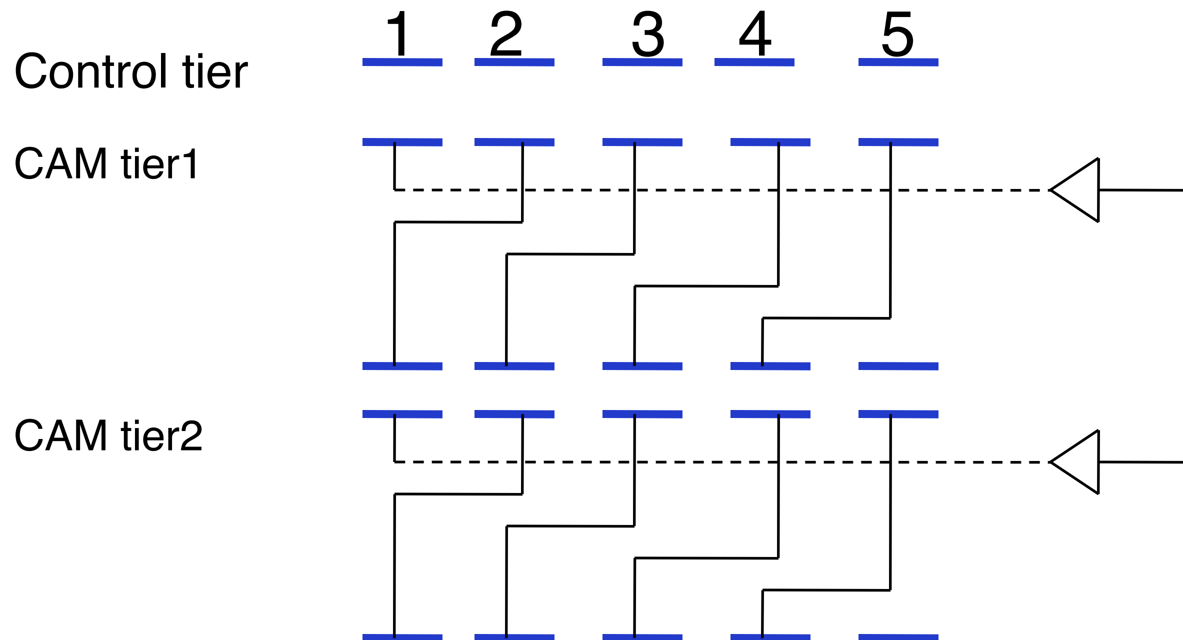


- The example for 4 identical CAM tiers, offset in one direction
- Every vertical connection has 3 extra connections on CAM tier
- Point to point communication done by offset (to/from Control)
- Power, clock etc lines have all 4 connected together

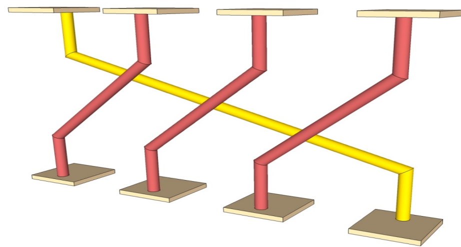
No extra transistor needed, pure geometrical solution.
But requires offset at wafer stacking stage…

Turns out Bob Patti at Tezzaron had a simpler idea to solve this problem long ago -- patented in 1999

- The idea was used for 3D DRAM stacking, to solve the same problem we are having, using "diagonal via"



- One example case
  - Diagonal via structure for 4 CAM tier case



The same can be done for all input and output signals,
No extra transistor is needed.
This trick solves the tier communication problem in a simple and clean way.

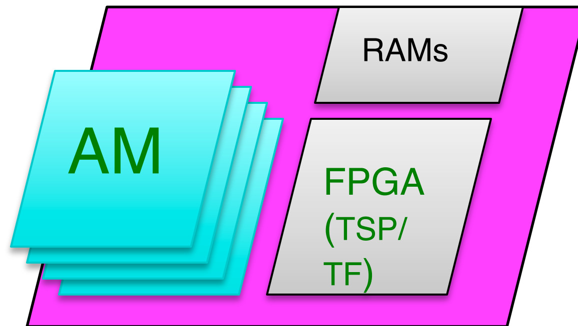
Price to pay:
a set of vias per signal
Number of vias = number of layers/tiers

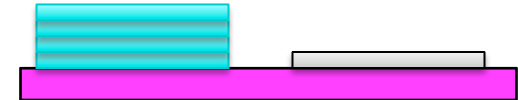
Control Tier Signal for CAM Tier 4
Control Tier Signal for CAM Tier 3
Control Tier Signal for CAM Tier 2
Control Tier Signal for CAM Tier 1
Control Tier
CAM Tier 1
CAM Tier 2
CAM Tier 3
CAM Tier 4

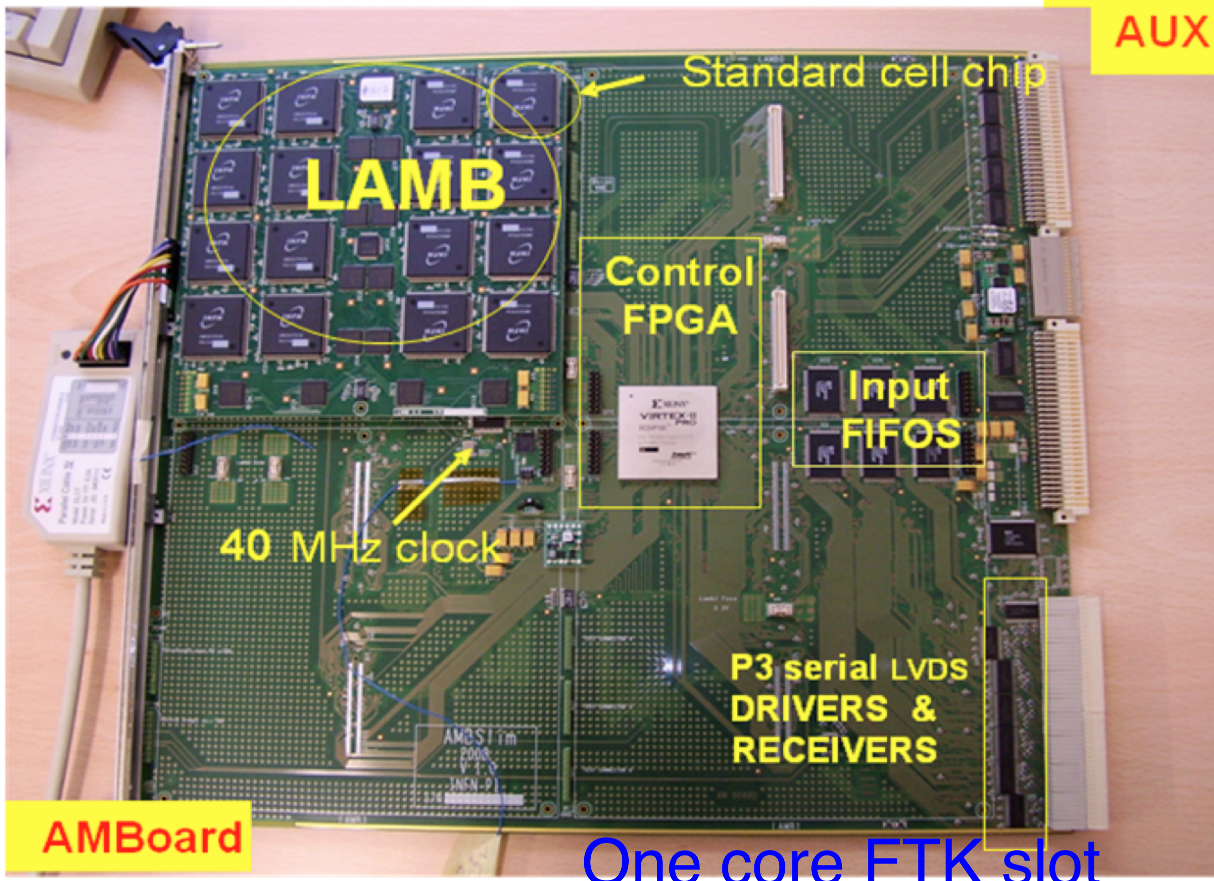
Integrate AM and TF stages into one chip

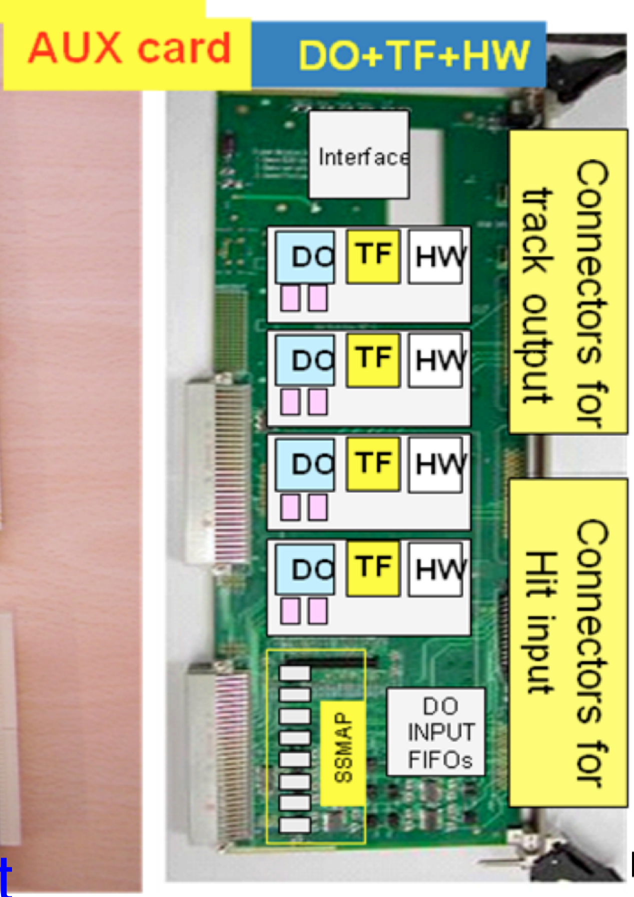
→ "Original SVT wedge in one chip"

- Bandwidth between AM stage and Track Fitting stage is another major challenge
  - As AM pattern size increases, need to transfer large number of fired roads and associated full resolution hits from AM stage into the TF stage
  - The larger the AM pattern size, the more demand
  - Highly desirable if the two stages can be integrated
  - Board/system level design could be much simplified
  - Potentially large cost saving (esp. system level)
- 3D Technology could help here
  - Similar to enhancing CPU memory access bandwidth
  - Would make the chip much more flexible (within & outside HEP)
  - Generic R&D



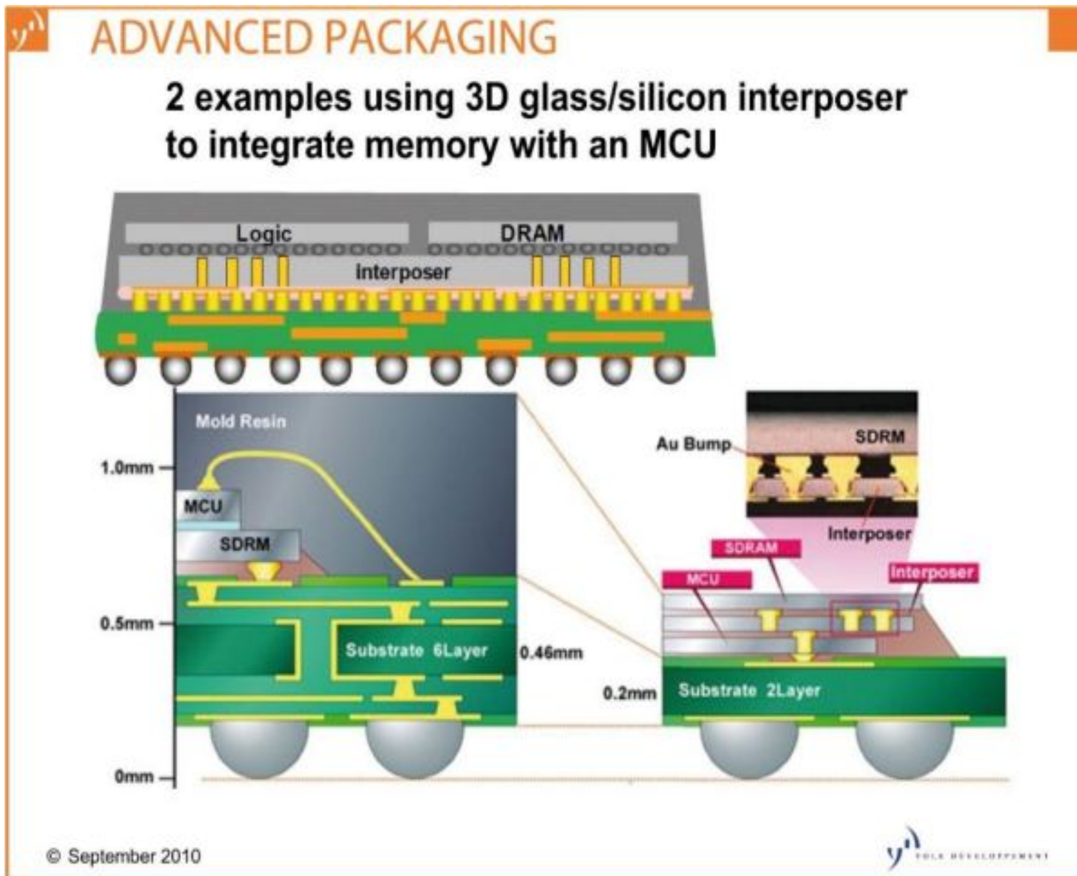




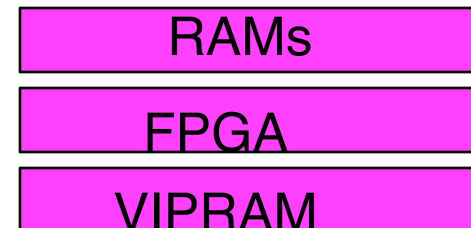


One core FTK slot

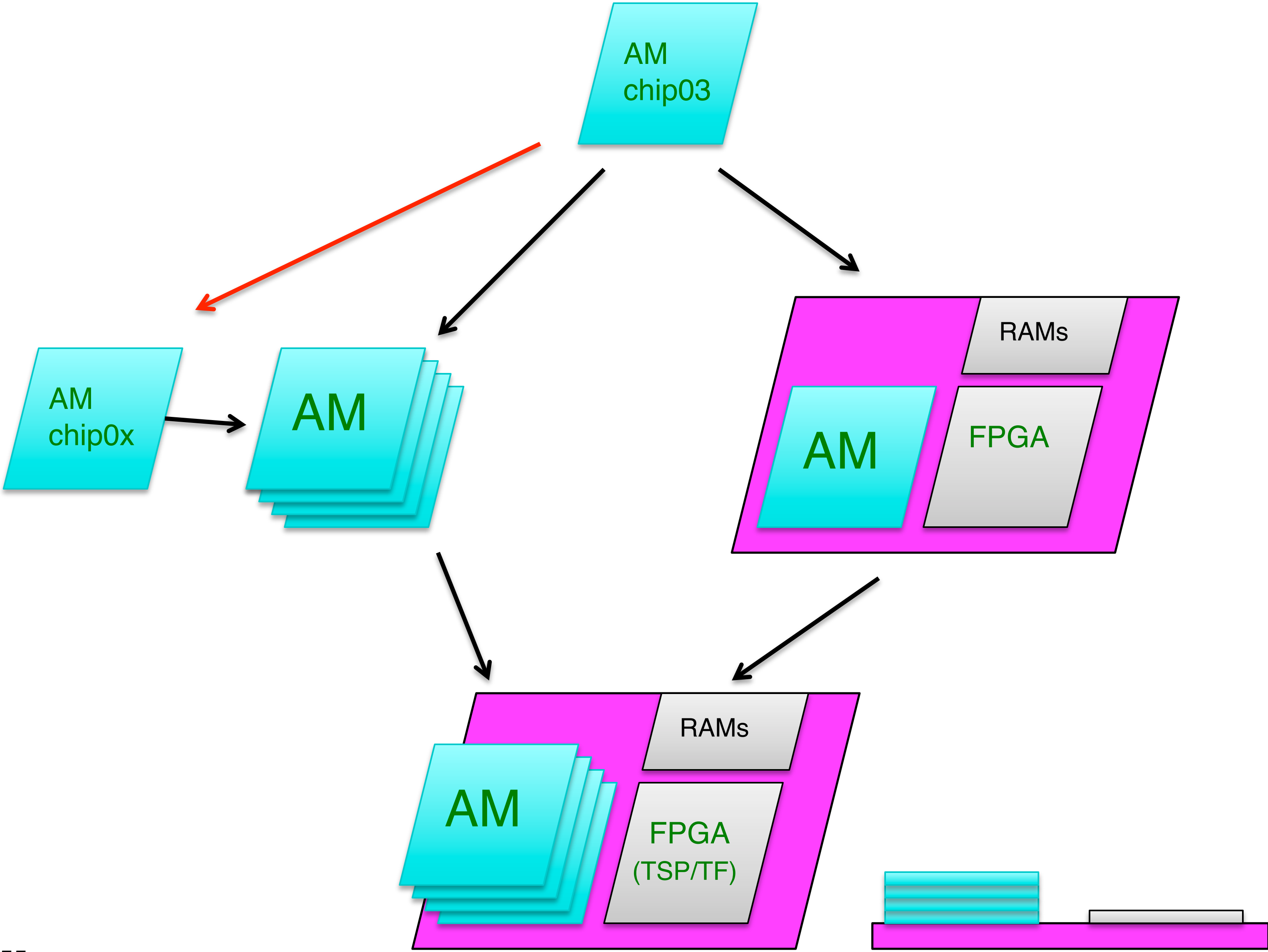
Examples



Or directly integrate AM with FPGA/RAMs using 3D vertical interconnect: Perhaps possible in the future

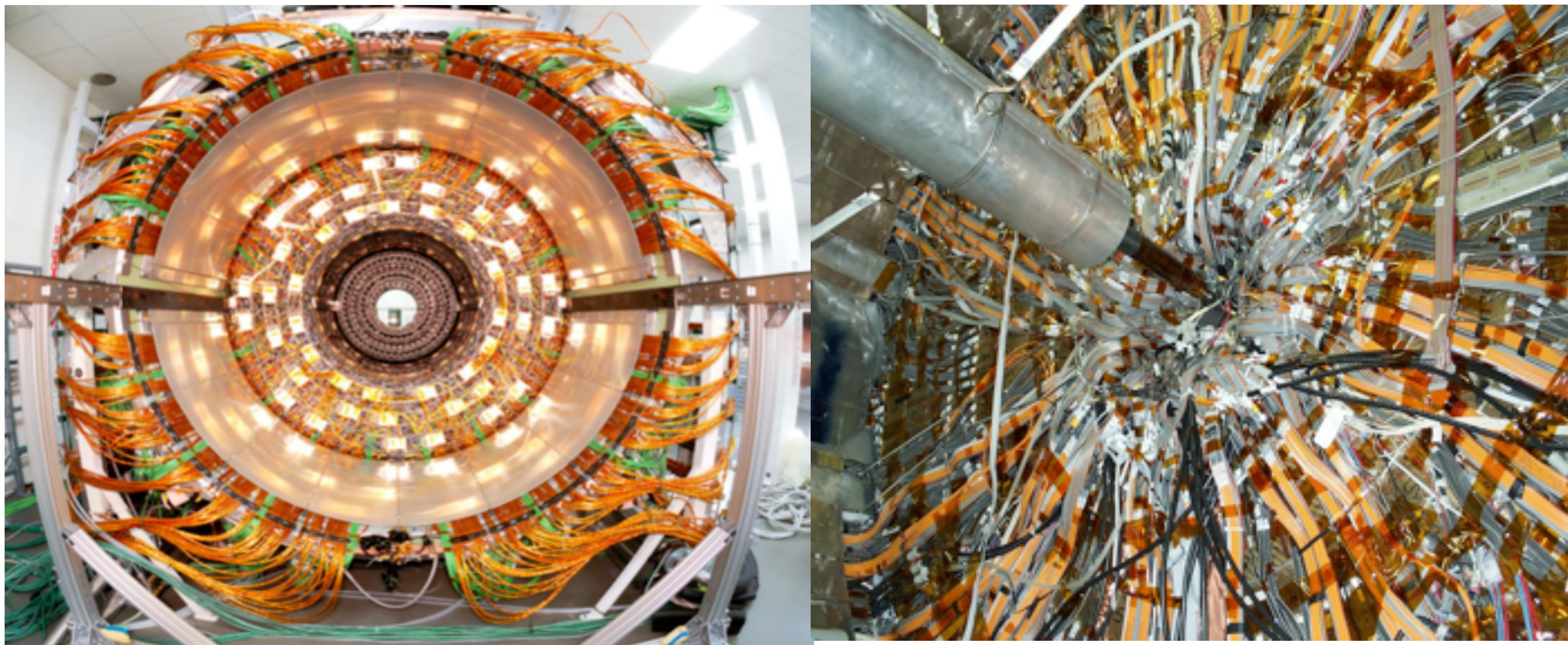


Integrate AM with FPGA+RAMs: Possible Development Paths



Fermilab

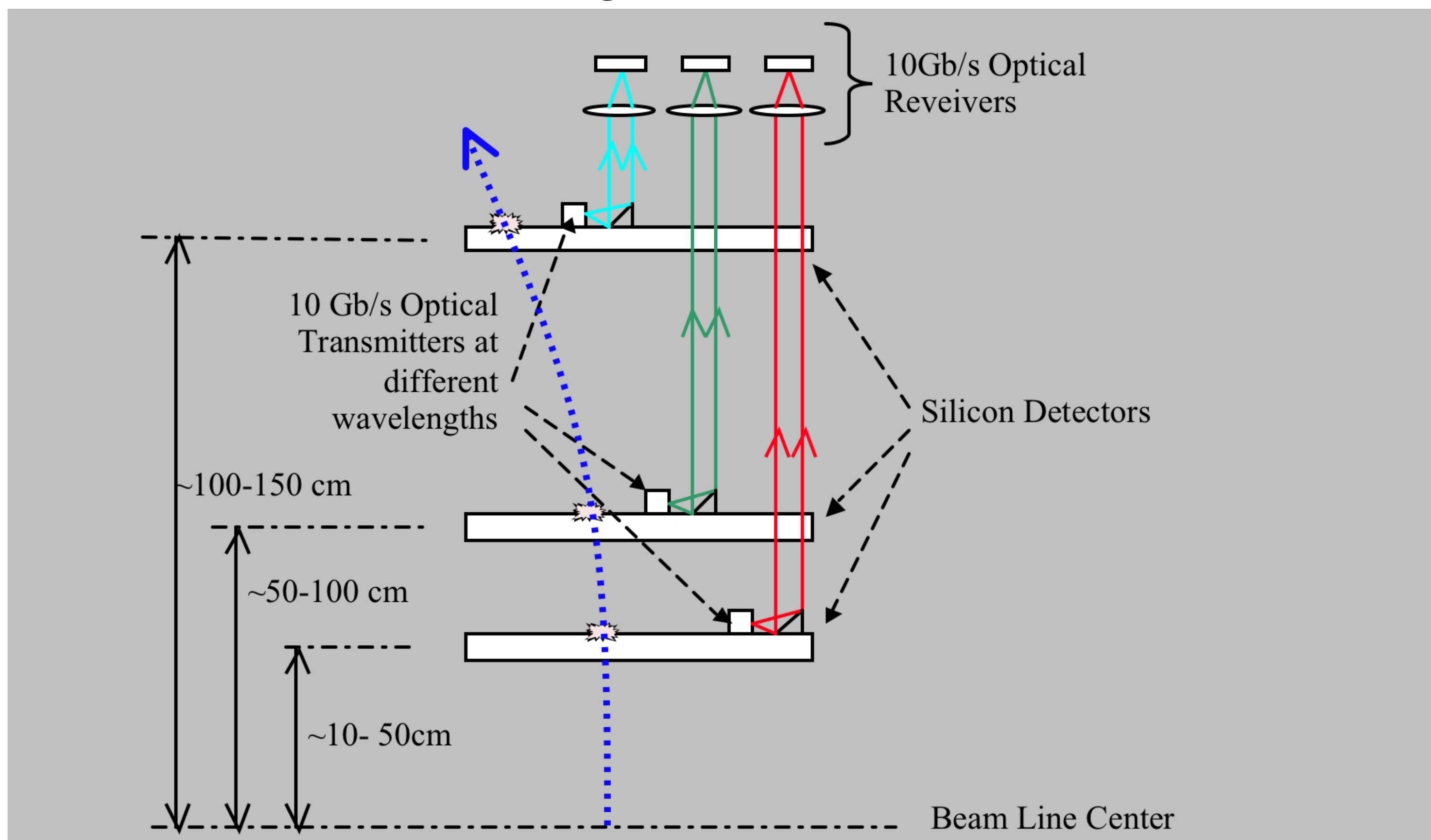
"Cables R Us"



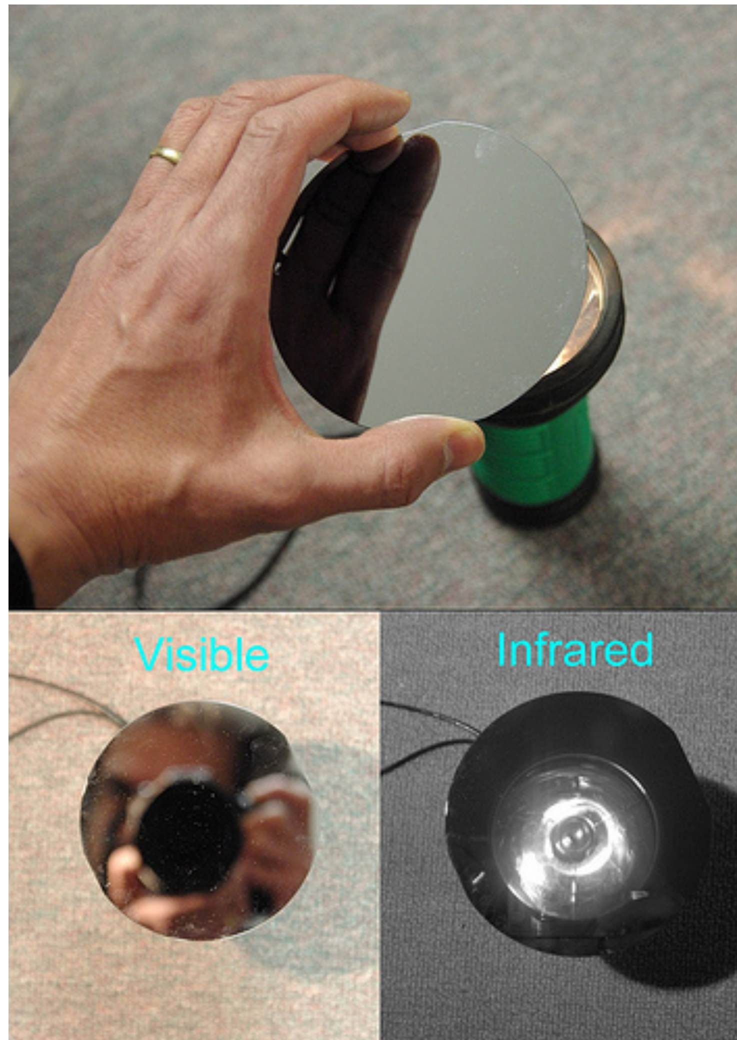
Food for thought: possible free-space optical interconnection, cable-less?

- A conceptual sketch of a free-space optical link for trigger & readout. The data links will operate in the infrared range, for which silicon is transparent ... (other free-space link ideas out there… see TIPP 2011)

Is this idea interesting? What are the possible issues?



Silicon transparent in IR



Homework for students

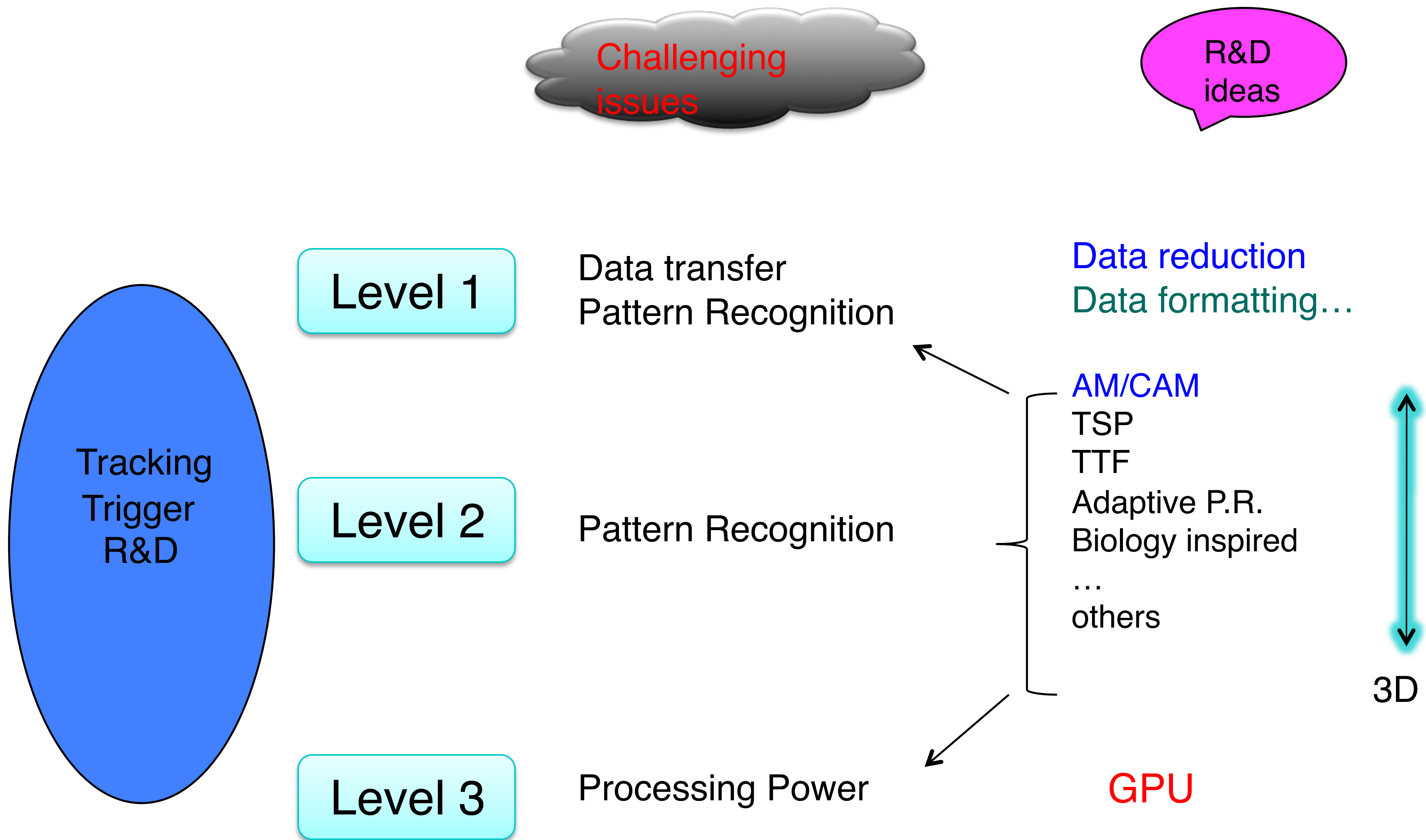
- Identify a technical challenge in your work (or use challenges mentioned in this lecture)
  - Try you best to come up with some crazy ideas to address the challenge
  - Then try to kill the ideas, by yourself first, and ask others to help
  - See if you can come up with one idea that cannot be killed easily…

- The right question to ask:

  It is not whether the idea is crazy or not, rather, it is about

  whether the idea is crazy enough or not.

Even if you cannot come up with any good ideas in the end,
you will learn A LOT in the process.



The ultimate physics reach of LHC will critically depend on the ultimate tracking trigger capabilities of its experiments

GPUs (Graphic Processing Units) have evolved into highly parallel, multi-threaded, multicore processors with remarkable computational power and high memory bandwidth, driven mostly by the high demand of real-time 3-D graphics. The GPUs also come with software environment that allows developers to use C/C++ as a high-level programming language, making it highly accessible to the general user. The combination of highly parallel architecture, high memory bandwidth as well as the user-friendly software environment makes GPUs a potentially promising technology for effective real-time processing for future high energy physics experiments.

Motivation

- Power of GPUs has increased rapidly due to demands of 3D graphics
  - Highly parallelized architecture
  - High memory bandwidth
- Many applications of GPUs outside of imaging
  - Commercially available → cheaper than dedicated hardware
  - Application programming interfaces like nVidia's CUDA C ease development of software for new applications

Photograph of GTX 285 GPU, courtesy of nVidia.

- ***Are GPUs suitable for low-latency environments, like a HEP trigger?***

GPU vs CPU Computation

CPU (Intel Core i7-930)

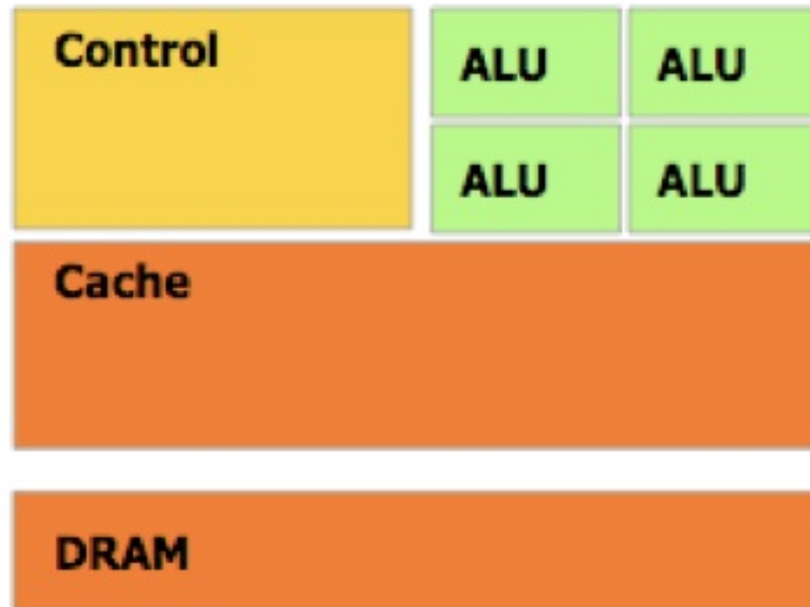
- Limited number of simultaneous calculations possible
  - 1 microprocessor
  - 4 cores
  - 8 threads
- Large cache size
  - 8 MB

GPU (nVidia GeForce GTX 285)

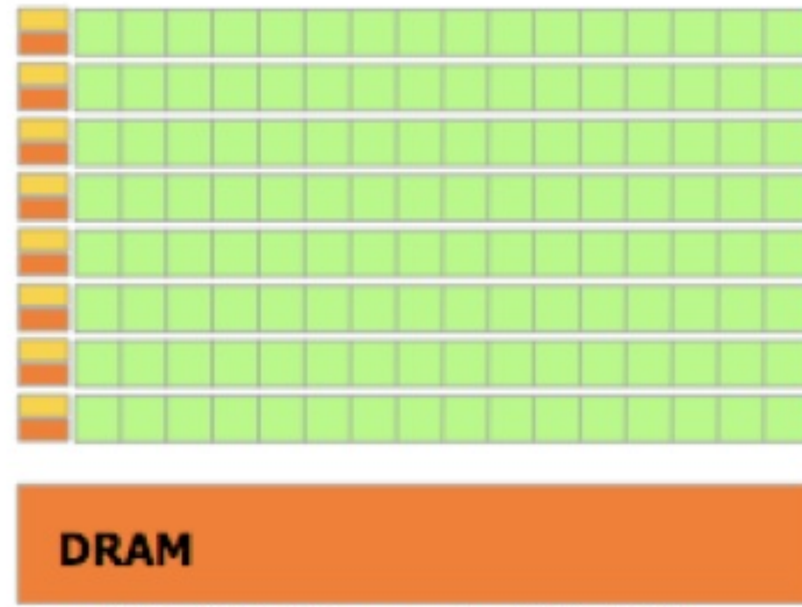
- Designed for running many instances of same routine simultaneously
  - 30 microprocessors
  - 240 cores
  - 1024 x 30 threads (max)
- Small cache size
  - 8 kB / microprocessor

GPU vs CPU Computation

CPU (Intel Core i7-930)



GPU (nVidia GeForce GTX 285)



From nVidia CUDA C Programming Guide (v 3.2)

GPU vs CPU Computation

CPU (Intel Core i7-930)

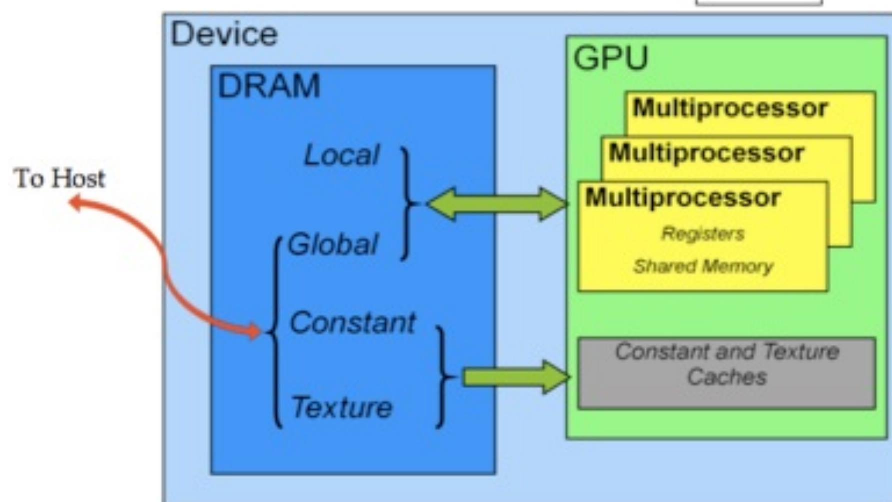
- Limited number of simultaneous calculations possible
  - 1 microprocessor
  - 4 cores
  - 8 threads
- Large cache size
  - 8 MB
- Sits directly on motherboard
  - Latency scale set by number/ speed of operations

GPU (nVidia GeForce GTX 285)

- Designed for running many instances of same routine simultaneously
  - 30 microprocessors
  - 240 cores
  - 1024 x 30 threads (max)
- Small cache size
  - 8 kB / microprocessor
- Communicates with CPU through PCIe bus
  - Latency scale set by host (CPU) ↔ device (GPU) communication

Fermilab

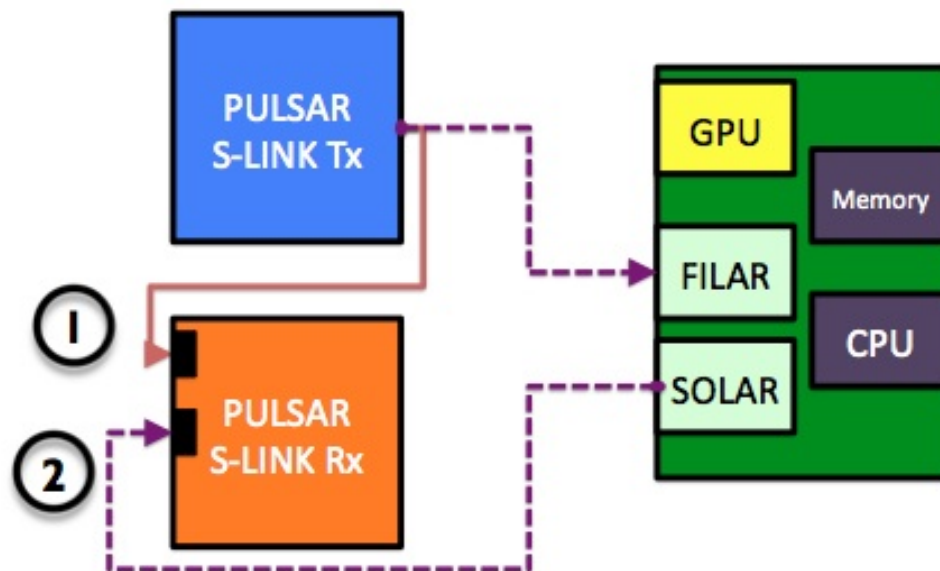
GPU Memory Structure



From nVidia CUDA C Best Practices Guide (v 4.0)

- Various memory locations for storing/accessing data
  - Global Memory
    - Most available space
    - Read/Write
    - Slow access
  - Constant/Texture Memory
    - Smaller storage space
    - Read Only
    - Cacheable on multiprocessors (faster access)
  - Registers/Shared Memory
    - Limited storage space
    - Read/Write
    - Fast access for individual threads for thread blocks

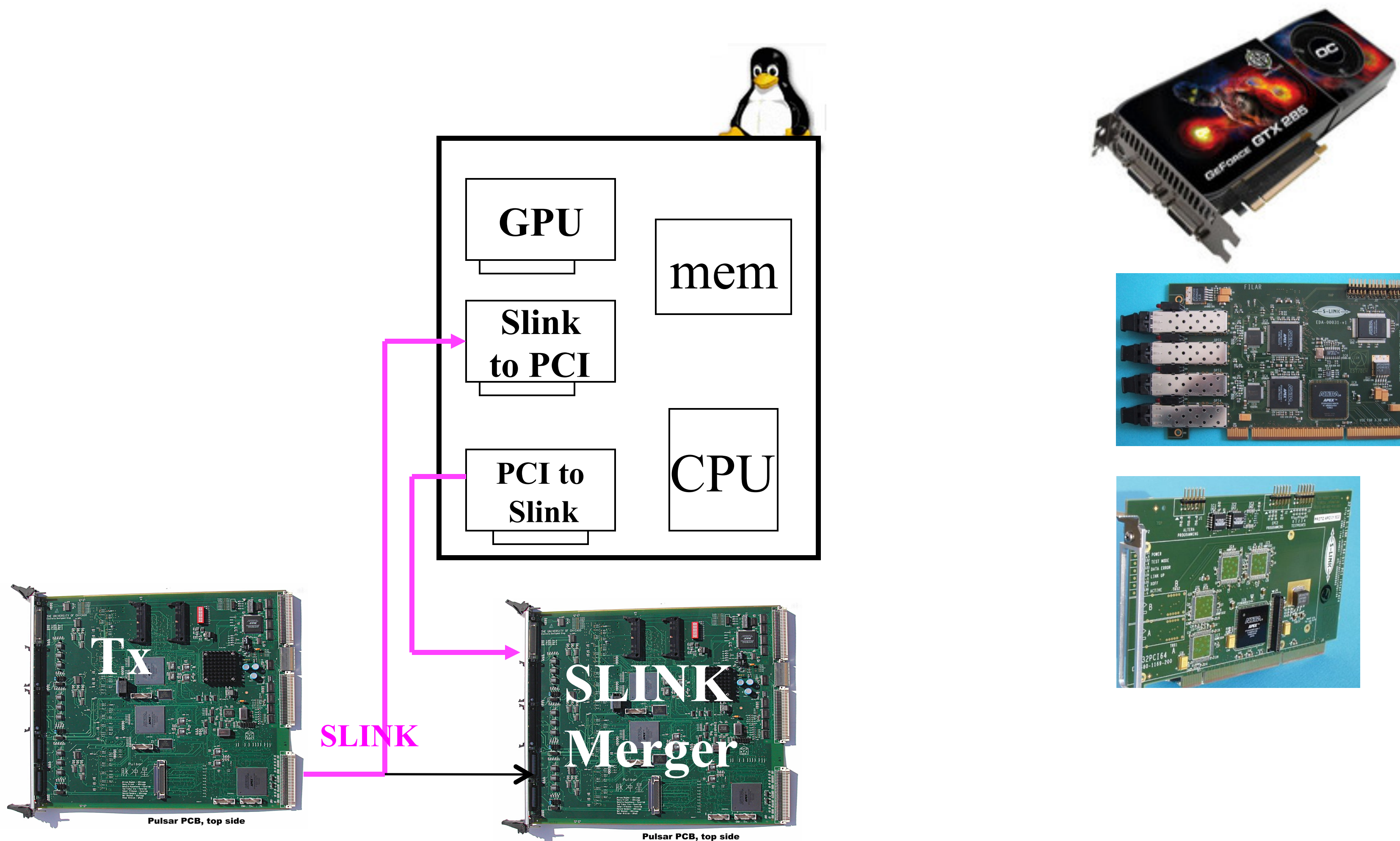
Experimental Setup: Data Flow



Steps in PC

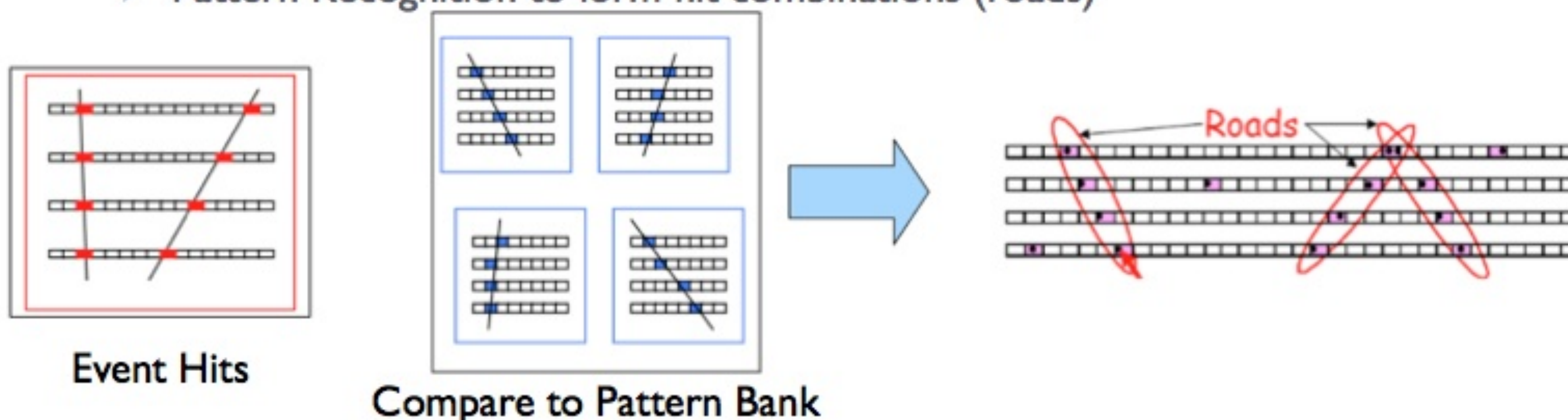
- Receive input data
- Copy input to GPU
- Perform calculations
- Copy results from GPU
- Send output

Trigger Test stand *at CDF*



The Computation: Linearized Track Fitting

- Want to run algorithm that would be used in HEP trigger
- CDF Silicon Vertex Trigger (SVT) finds displaced vertices at L2
  - Pattern Recognition to form hit combinations (roads)



- Perform track-fitting inside roads using simple scalar product

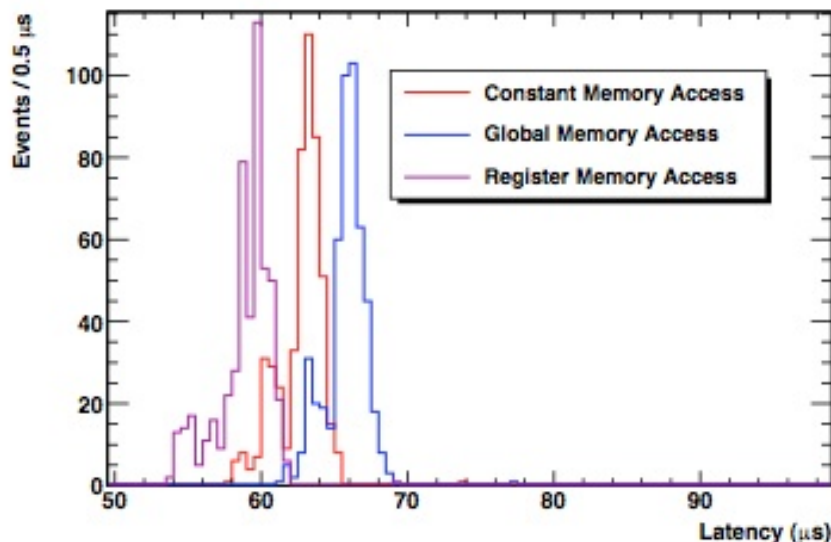
$$p_i = \vec{f}_i \cdot \vec{x} + q_i$$

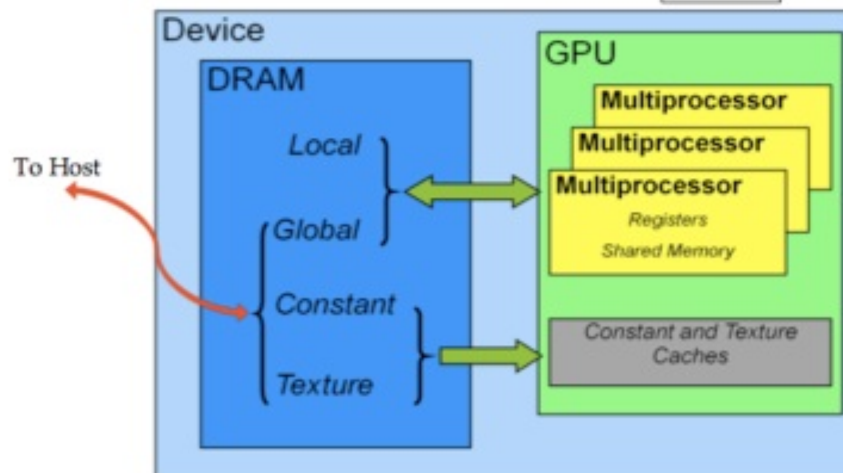
Known constants. Precalculated and ***stored in memory***

track parameters (output)

track coordinates (input hit information)

Varying GPU Memory Lookup





From nVidia CUDA C Best Practices Guide (v 4.0)

- Algorithm accesses pre-defined constants for track fitting
  - $p_i = \vec{f_i} \cdot \vec{x} + q_i$
- Location in memory affects latency
- Significant dependence of latency on handling of ***memory lookup***
  - Differences ~ 10 μs between register and global memory
- Good management → optimized performance

Wes Ketchum's slides from TIPP 2011

Summary of Lecture III

- Trigger in HEP: the view into the future
  - L1 Tracking Trigger will be crucial to LHC physics program at much higher luminosity
  - *HUGE challenges* in implementing such tracking trigger capability
  - Mentioned some R&D activities at Fermilab. There are many other ideas/projects out there at other places. No time to cover in this lecture (you can learn more from recent TIPP 2011 conference).
  - 3D Technology could be useful, *once it becomes mature enough*
  - Associate Memory is just one approach for hardware pattern recognition… and could use some new/crazy ideas here…
  - GPU seems promising in helping higher level trigger. A few experiments/groups are working on this (e.g. NA 62, see TIPP2011)
  - ATCA technology is very attractive (see backup slides for a recent case study… no time to cover. Patrick will introduce ATCA in his talk)

Summary: We should pay close attention to what industry is developing and take full advantage of that

Backup slides

- The next two slides should be included in Lecture 1
  - Cross Section vs energy
  - The two slides should be after Slide 26 in Lecture 1
  - Will update Lecture 1.

Either increase the luminosity, or the cross section, often both:

Rate = σ L

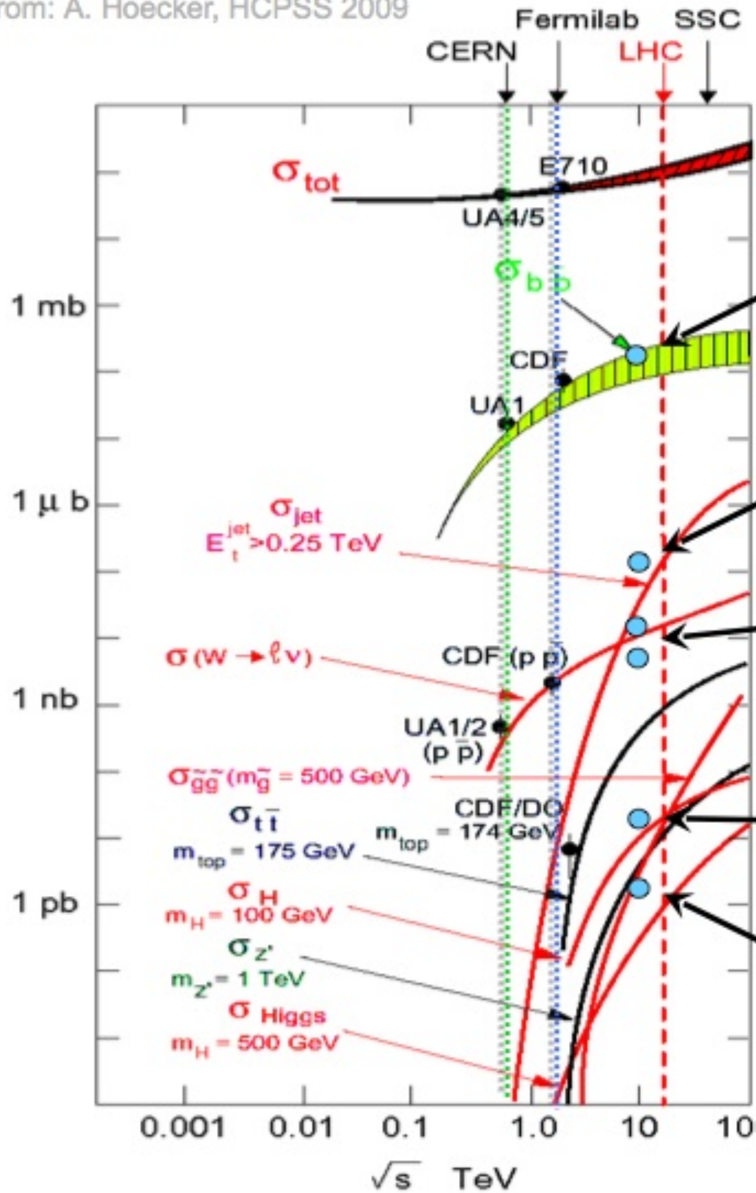
- Increase the cross section
  - Go to higher energy
    - From Tevatron to LHC
  - Running on resonance:
    - e.g. KLOE/BES/CLEO/Babar/Belle/LEP/SLC/…
- Increase the luminosity
  - Higher luminosity/intensity
    - From CLEO/CESR to B factories to SuperBs
  - Larger detector
    - CTA, LHASSO etc

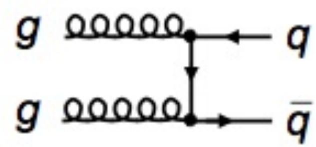
Slide 26 from Lecture 1

ALL lead to challenges in Triggering

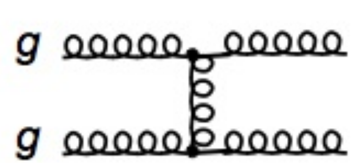
Cross section vs energy at hadron collider

Slide from: A. Hoecker, HCPSS 2009

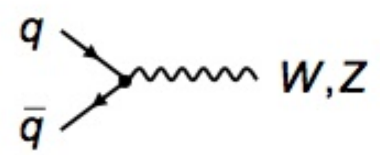




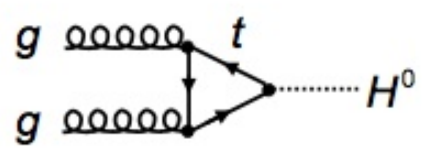
Quark-flavour production



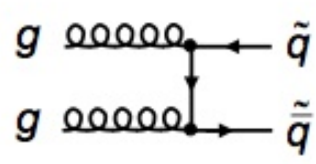
High-p_T QCD jets



W, *Z* production

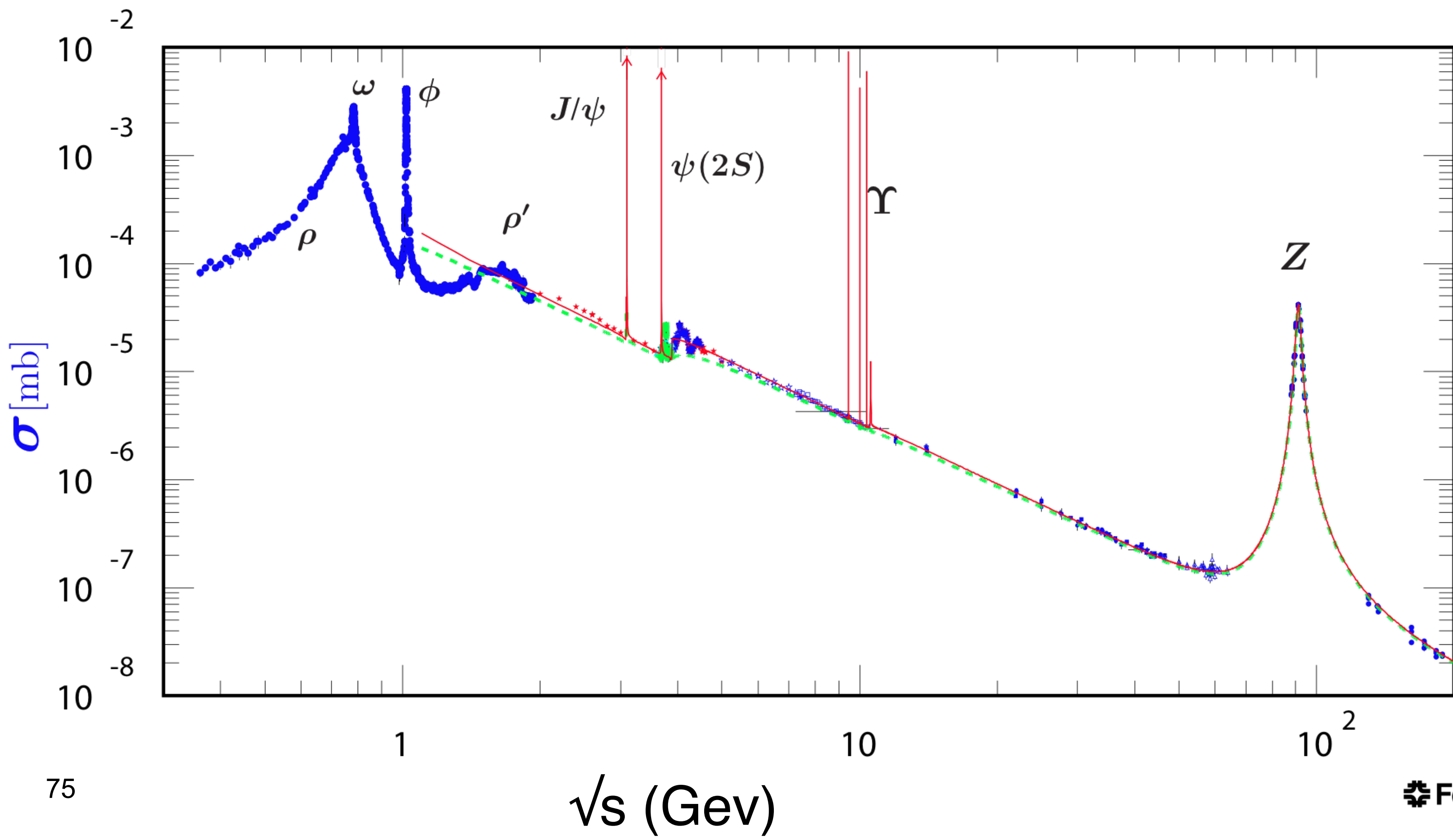


gluon-to-Higgs fusion



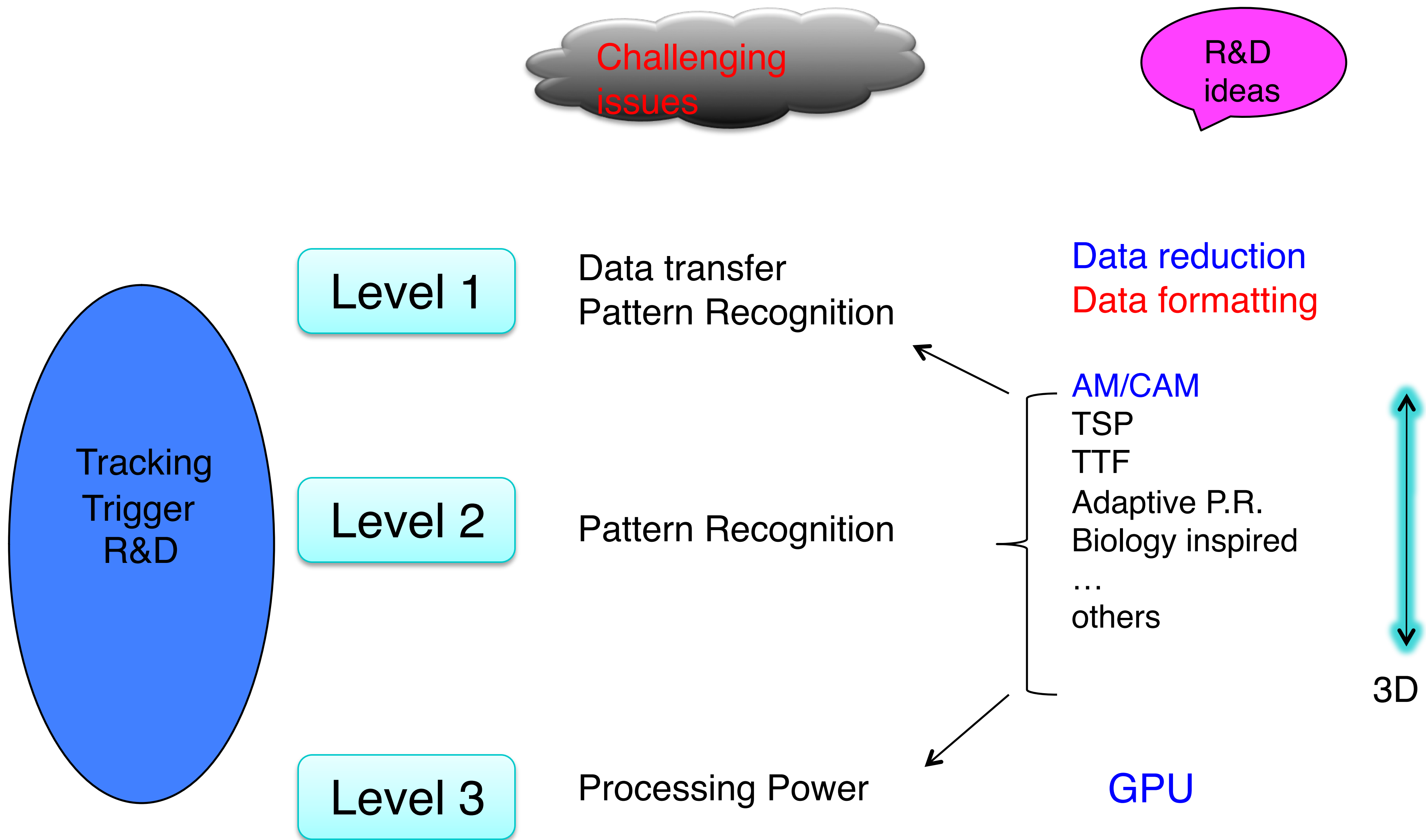
squarks, gluinos
(m ~ 1 TeV)

Total cross section in e^+e^- Collisions



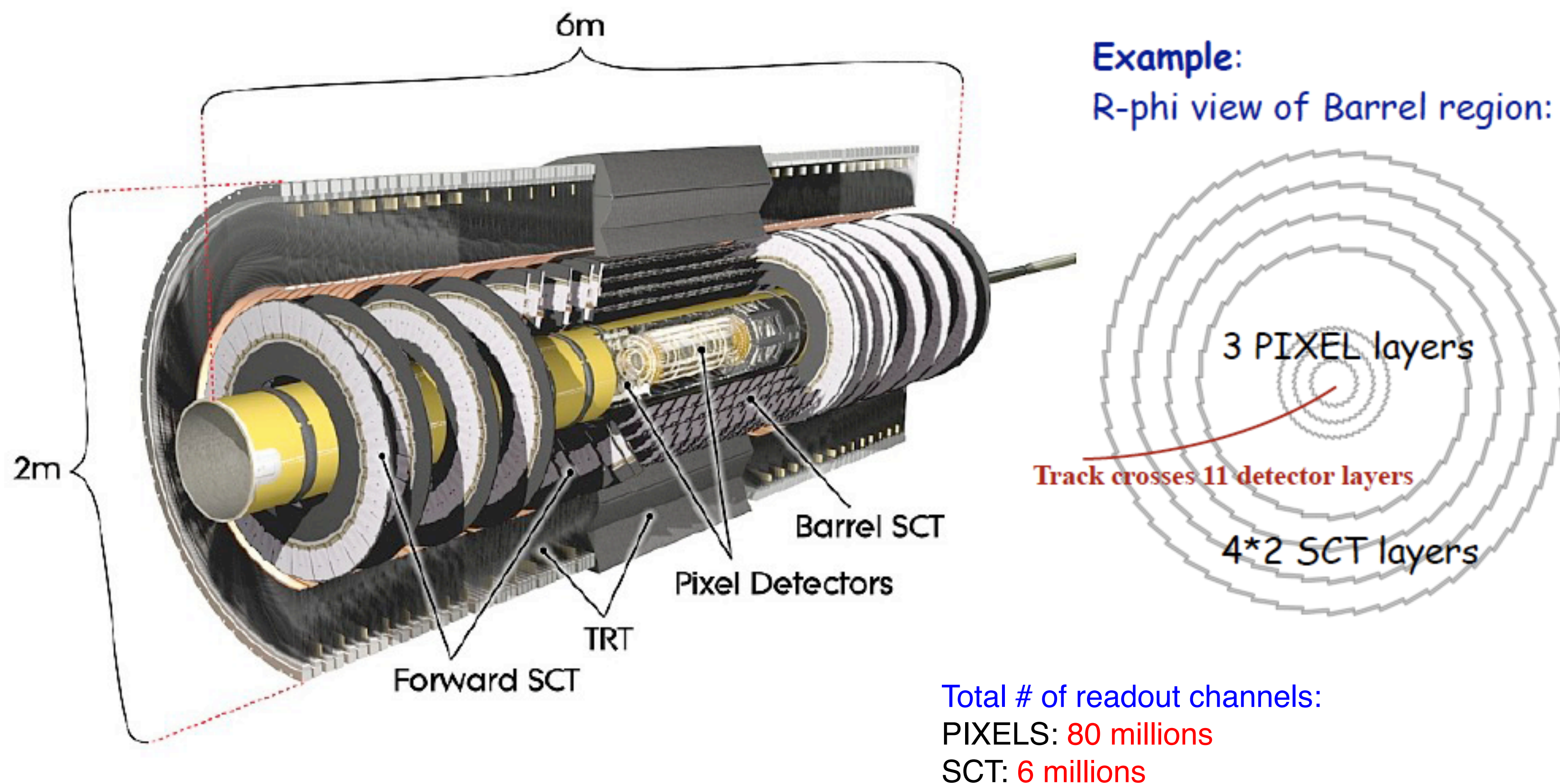
One more case study

- Data Formatter issues for FTK
- PRELIMINARY RESULTS (work in progress)
  - Just to give an very recent example, for educational purpose, to show how to study some of the design issues in tracking trigger
  - And the advantage of ATCA over VME



The ultimate physics reach of LHC will critically depend on the ultimate tracking trigger capabilities of its experiments

Fast tracking with pixel and SCT det.



Total # of readout channels:
PIXELS: 80 millions
SCT: 6 millions

Silicon Tracking Systems: ATLAS Barrel

TRT: R = 1082 mm, R = 554 mm

SCT: R = 514 mm, R = 443 mm, R = 371 mm, R = 299 mm

Pixels: R = 122.5 mm, R = 88.5 mm, R = 50.5 mm

R = 0 mm

B = 2T

TRT

SCT

Pixels

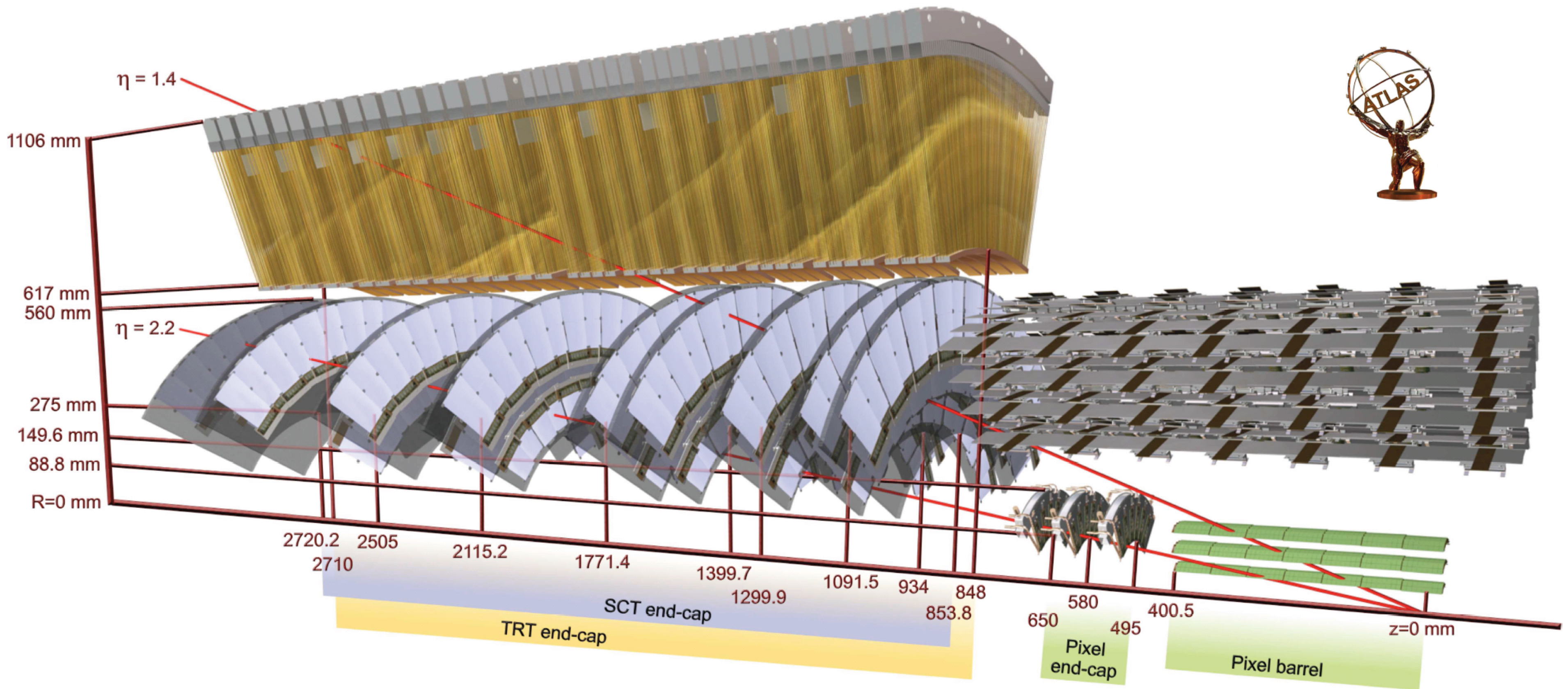
SCT:

- 6.3M channels
- 4 double barrel layers
  - 80mrad stereo angle
  - strip pitch 80 μm
  - binary readout

Performance: $(\eta = 0)$

- $\sigma(p_T)/p_T =$ 0.038% $\times\, p_T$ (GeV)
- σ_b = 11 μm @ p_T = 1 TeV

Silicon Tracking Systems: ATLAS



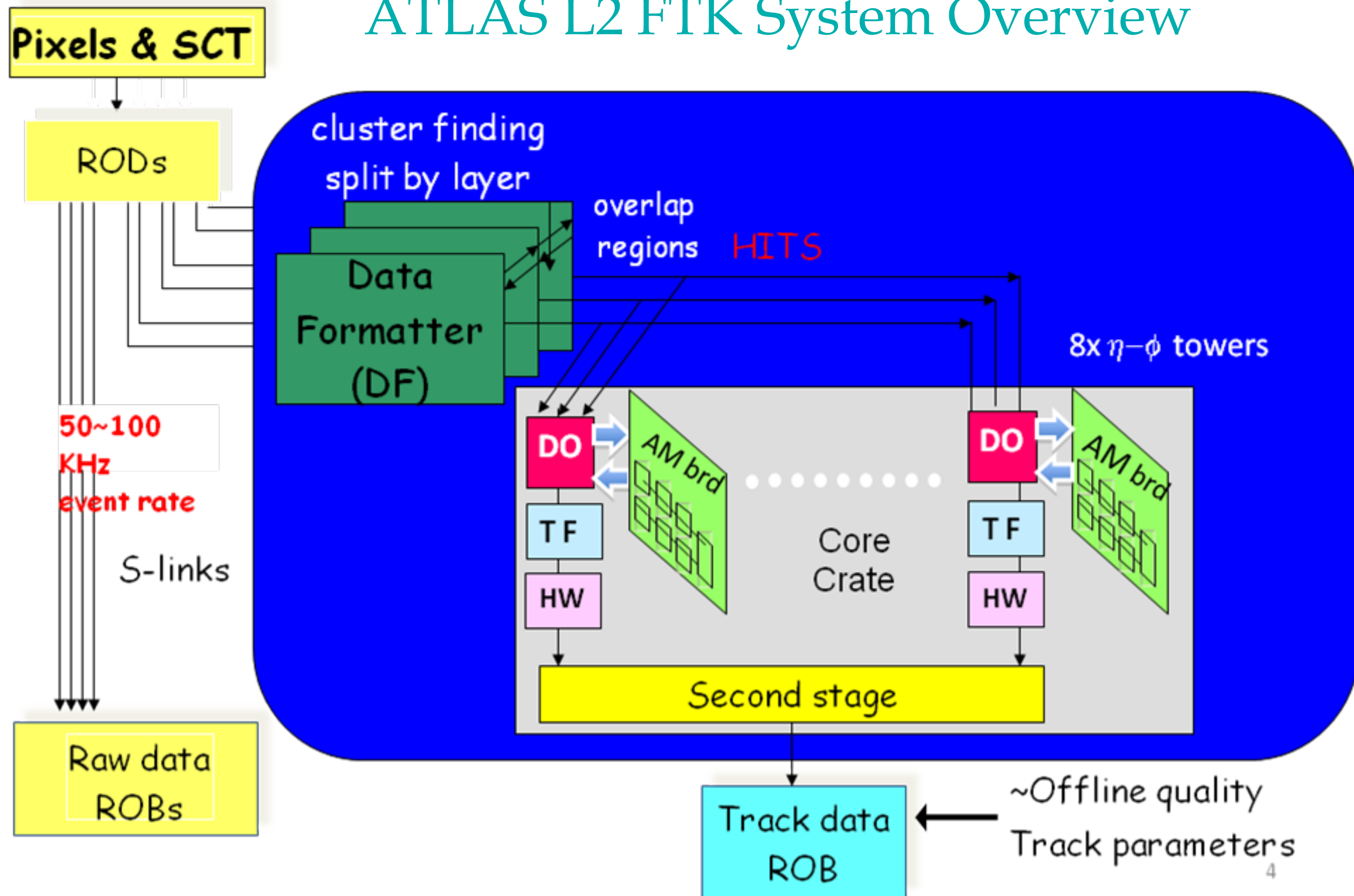
SCT: 9 double sided-disks (radial+40mrad)

- $1.5 < |\eta| < 2.5$

Performance: $(\eta = 2.5)$

- $\sigma(p_T)/p_T = 0.11\% \times p_T$ (GeV)
- $\sigma_b = 11\ \mu m$ @ $p_T = 1$ TeV

Fermilab



- Highly parallel data flow: 64 η-ϕ towers in 8 core crates and 4-fold parallelism within each tower (for 3×10^{34})

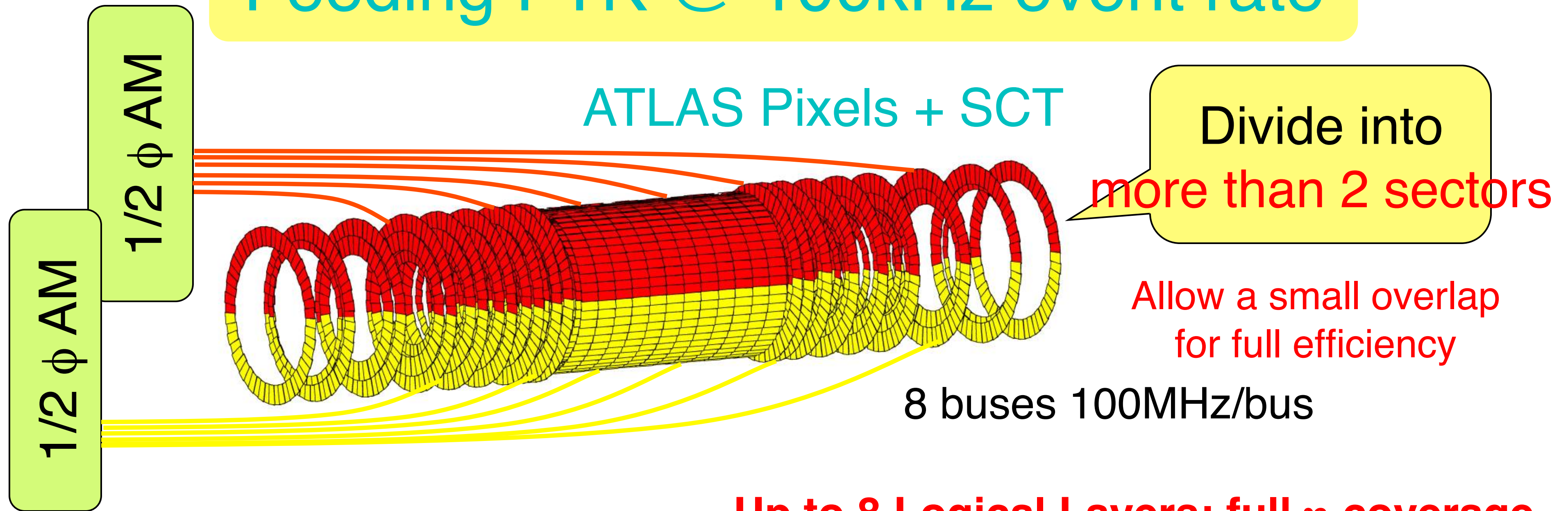
Fermilab

The technical difficulties

- \# of hits in the tracking chamber per beam crossing: 200k
  Must transfer to FTK each 10 μs (100 kHz level-1 trigger rate)
  ⇒ ~ 20 gigawords per second transfer
- This much data makes both stages in tracking very challenging: pattern recognition and track fitting
- There are several hundred good tracks per beam crossing.
  - 10 μs / event ⇒ < 100 ns/track for pattern recognition plus track fitting

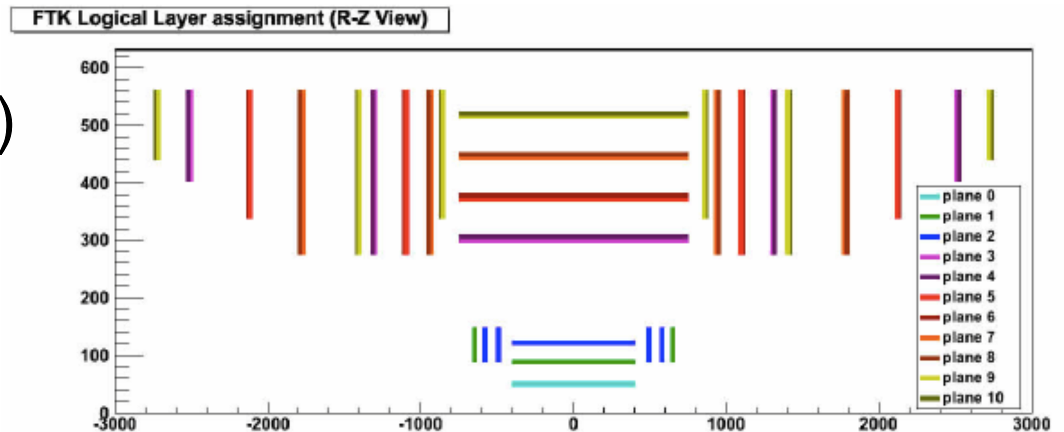
Will comment on data formatting issue next
(A recent real-life example on track trigger design studies)

Feeding FTK @ 100kHz event rate



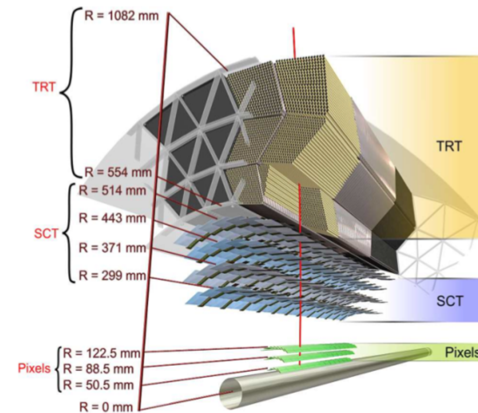
Up to 8 Logical Layers: full η coverage

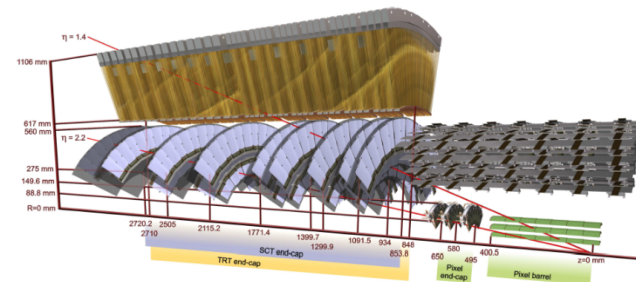
- 8 ϕ regions each with
- 8 sub-regions ($\eta - \phi$ towers)
  - $\delta\phi \sim 22.5^o$, $\delta\eta \sim 1.25$
  - bandwidth for up to $3*10E34\ cm^{-2}s^{-1}$



Data Formatter Inputs

- Detector modules are mapped to RODs
  - 6 to 48 modules per ROD/ROL
  - Developed tools to extract this mapping from ATLAS Conditions DB (COOL)
- 222 ROLs to DF
  - 1.25Gbps SLINK fiber with LC connector

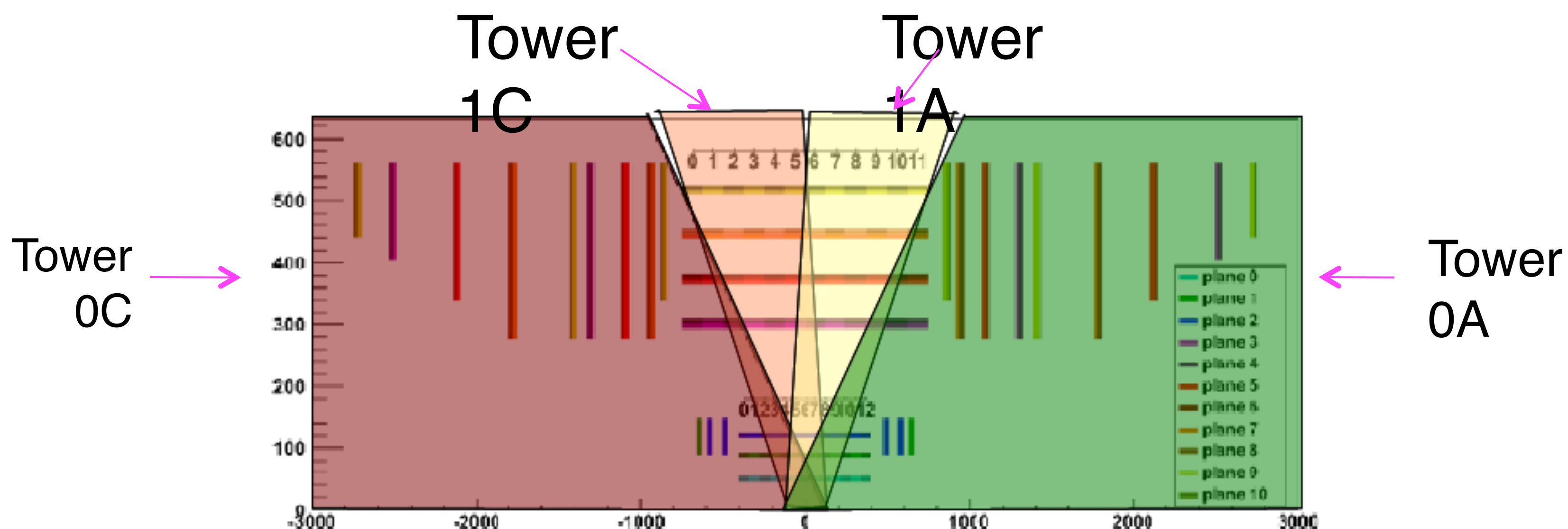




Fermilab

FTK Towers

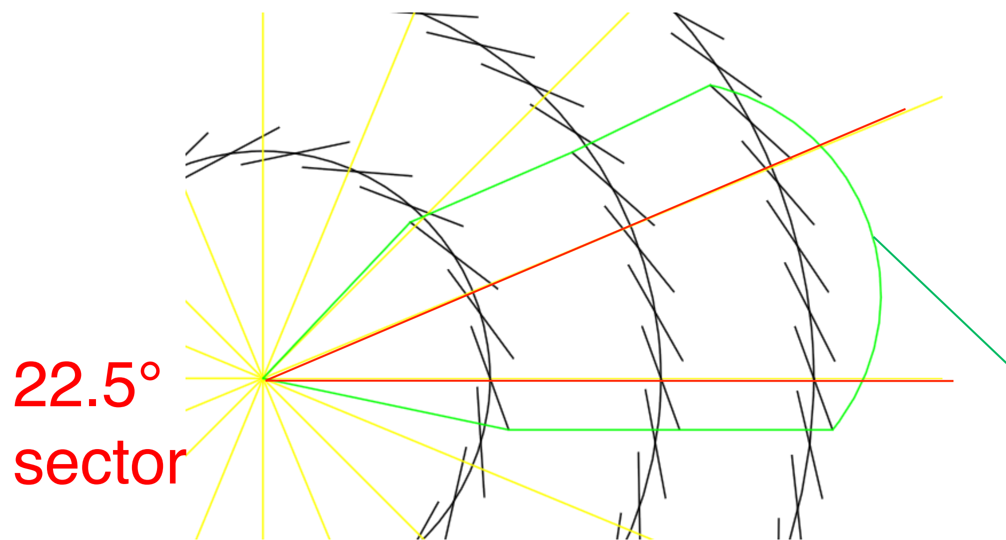
- 16 ϕ regions: 22.5° + ~10° overlap
- η sharing in four regions:



- 64 η-ϕ towers
- DF sends to 64 FTK processors downstream

Study on Data Sharing Needs: Some Basic Assumptions

- Combine **η towers 0 and 1** on a single DF board
- Total 32 DF boards: 16 A side + 16 C side
- Tower boundaries are **module defined**



Include even the modules which touch the tower boundary. This is more than 10° phi overlap called out in the FTK TP

ROLMAP Simulator

- Reads in two files:
  - ROL-module map (intense effort extract from ATLAS DB)
  - DF-module map
- Calculates the intersection between DF boards and ROLs
  - Module defined boundaries for DF boards
- Assigns ROLs to DF boards
  - Minimize sharing between DF boards
  - Balance the number of DF inputs
- Reports how many modules need to shared between DF boards
- Recently re-written in Python
  - C-- effort abandoned, too many damn pointers, painful sorting

Work done by Fermilab engineer Jamieson Olsen

Simulation Results

- 6-7 input ROLs per DF board
  - Does not include IBL yet
- Data sharing between DF boards is:
  - Highly dependent on ROL-module mapping
  - Not a regular pattern
  - More extensive than originally thought

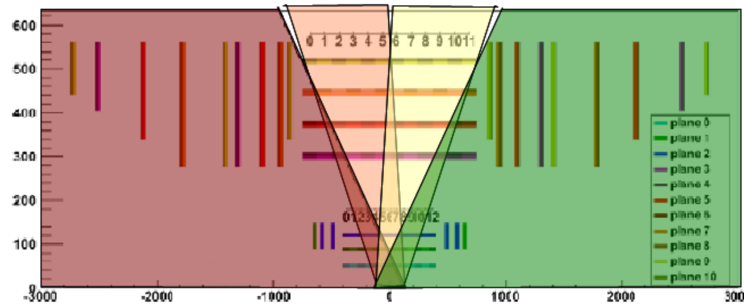
DF Data Sharing Matrix

	A01	A02	A03	A04	A05	A06	A07	A08	A09	A10	A11	A12	A13	A14	A15	A16	C01	C02	C03	C04	C05	C06	C07	C08	C09	C10	C11	C12	C13	C14	C15	C16
A01	X	31	3													37	39	19	7													15
A02	58	X	98	46	9											3	4	24	11													
A03	14	28	X	27		3	3									2	9	47	42	4		7	7									
A04			44	X	67	6													14	30	15	2										
A05	11	15	16	30	X	62	2									2			6	11	32	42	6									
A06				7	57	X	81	47	9												3	21	8									
A07				2	5	30	X	41														11	30	11								
A08					10	14	61	X	48			8	8										11	35	12			8	8			
A09								18	X	43	16	3												2	26	26	15					
A10	4	5	4	3		1		6	85	X	57	59	11	1		1									9	16						
A11	3	4	4	3	3	4	4	5	20	44	X	67	6	4	4	4									2	10	26	15				
A12								2	2	9	26	X	29	6										6	6	13	28	30	19			
A13								6			3	50	X	49										2			7	42	47	11		
A14	11												57	X	72	51													3	15	5	
A15	2			1	4	4	4	3					13	53	X	38														9	24	7
A16	34	14	14	10	2							2	5	5	56	X	4														17	35
C01	38	25	6													8	X	45	16	10												30
C02	4	14	6														81	X	93	42	1			1	3	4	3	3				16
C03	9	16	32	11													14	21	X	52												2
C04			5	34	14														19	X	32	6										
C05				18	42	8														50	X	56	14	10								
C06					1	10	4													16	60	X	82	44								
C07		7	7			14	22	3									4	7	6	5	8	25	X	17	3	4	3	4	4	4	3	5
C08						3	24	36	11													7	82	X	43							
C09								9	36	14														23	X	48	14	12				
C10									9	21	13													16	91	X	59	48				
C11										54	64	4												2	5	63	X	27				
C12											19	69	6													4	37	X	12	6		
C13												4	44	23														26	X	79	14	10
C14													1	13	4			1	1		3	5	4	4	1	1	1	6	63	X	76	40
C15														19	35	9												2	5	46	X	17
C16	4														14	30	34	6												4	66	X

Read across the row. e.g, DF Board A01 exports 31 modules to DF A02, 3 modules to DF A03, 37 modules to DF A16, etc.

Partitioning Into Crates

Tower 0C + 1C on One DF board



Tower 0A + 1A on One DF board

- Data sharing in η direction is significant in barrel regions
  - Towers 0 and 1 on one side combined on a DF board
- To minimize η sharing it makes sense to have DF boards A*n* and C*n* in the same crate
- It works out well to group 4 A boards and 4 C boards in a crate
  - Let's return to the DF data sharing matrix with this partition

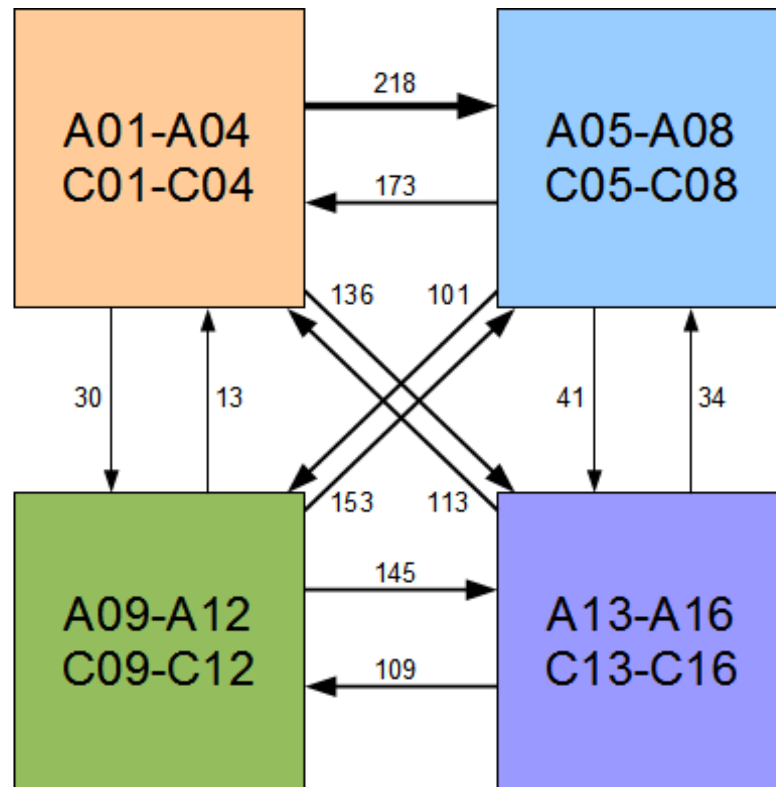
Fermilab

Crate Boundaries

	A01	A02	A03	A04	C01	C02	C03	C04	A05	A06	A07	A08	C05	C06	C07	C08	A09	A10	A11	A12	C09	C10	C11	C12	A13	A14	A15	A16	C13	C14	C15	C16
A01	0	31	3		39	19	7																					37				15
A02	58	0	98	46	4	24	11		9																			3				
A03	14	28	0	27	9	47	42	4		3	3			7	7													2				
A04			44	0			14	30	67	6			15	2																		
C01	38	25	6		0	45	16	10																				8				30
C02	4	14	6		81	0	93	42					1			1					3	4	3	3								16
C03	9	16	32	11	14	21	0	52																								2
C04			5	34			19	0	14				32	6																		
A05	11	15	16	30			6	11	0	62	2		32	42	6													2				
A06				7					57	0	81	47	3	21	8		9															
A07				2					5	30	0	41		11	30	11																
A08									10	14	61	0			11	35	48			8	12			8	8				8			
C05				18				50	42	8			0	56	14	10																
C06								16	1	10	4		60	0	82	44																
C07		7	7		4	7	6	5		14	22	3	8	25	0	17					3	4	3	4					4	4	3	5
C08										3	24	36		7	82	0	11				43											
A09												18				2	0	43	16	3	26	26	15									
A10	4	5	4	3						1		6					85	0	57	59	9	16			11	1		1				
A11	3	4	4	3					3	4	4	5					20	44	0	67	2	10	26	15	6	4	4	4				
A12												2				6	2	9	26	0	6	13	28	30	29	6			19			
C09												9				23	36	14			0	48	14	12								
C10																16	9	21	13		91	0	59	48								
C11																2		54	64	4	5	63	0	27								
C12																			19	69		4	37	0	6				12	6		
A13												6				2			3	50			7	42	0	49			47	11		
A14	11																								57	0	72	51	3	15	5	
A15	2			1					4	4	4	3													13	53	0	38		9	24	7
A16	34	14	14	10	4				2											2					5	5	56	0			17	35
C13																				4				26	44	23			0	79	14	10
C14						1	1						3	5	4	4					1	1	1	6	1	13	4		63	0	76	40
C15																								2		19	35	9	5	46	0	17
C16	4				34	6																					14	30		4	66	0

Inter-Crate Data Sharing

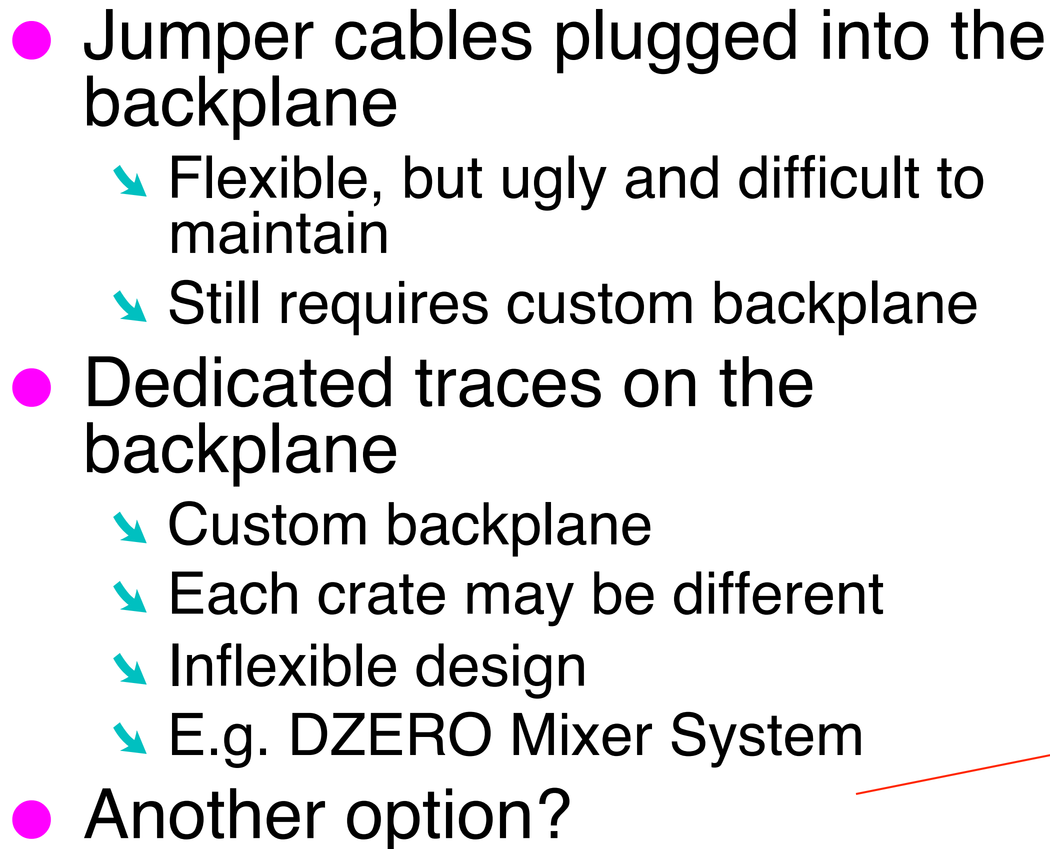
- ROLMAP calculates how many modules need to be shared across crate boundaries
- Data sharing between crates can be implemented with high speed links



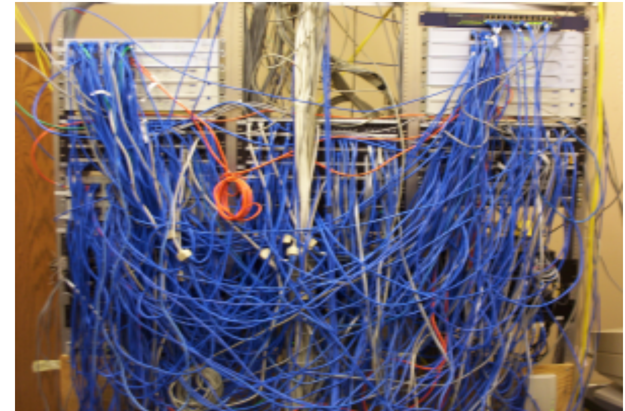
Data Formatter study Conclusions

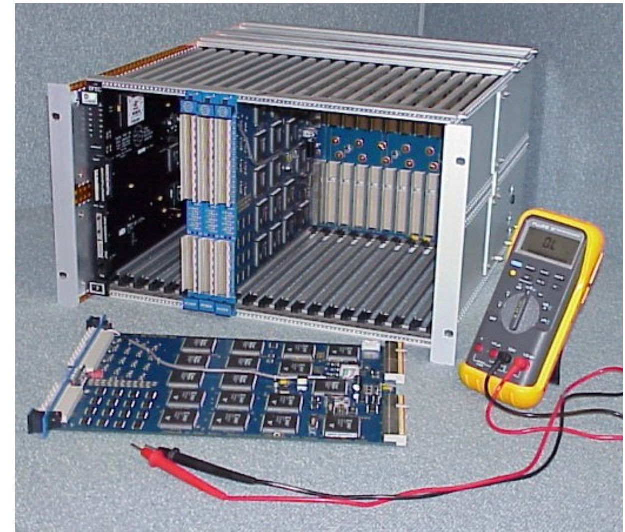
- A bottom-up approach to the DF design
- Considerable work went into determining ROL-module and DF-module mapping
- Developed a simulation tool to visualize the data sharing between boards
- We considered new and existing technologies to implement the DF data sharing
  - Considering future expansion, flexibility
  - Let the DF data sharing requirements drive hardware selection, not the other way around
- VME-type backplane solutions are not a good fit
- ATCA full mesh is a **natural fit** for the DF design

Intra-Crate Sharing Options



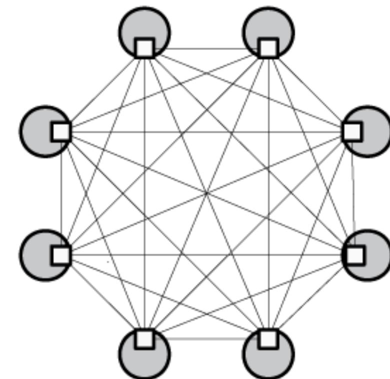
- Jumper cables plugged into the backplane
  - Flexible, but ugly and difficult to maintain
  - Still requires custom backplane
- Dedicated traces on the backplane
  - Custom backplane
  - Each crate may be different
  - Inflexible design
  - E.g. DZERO Mixer System
- Another option?



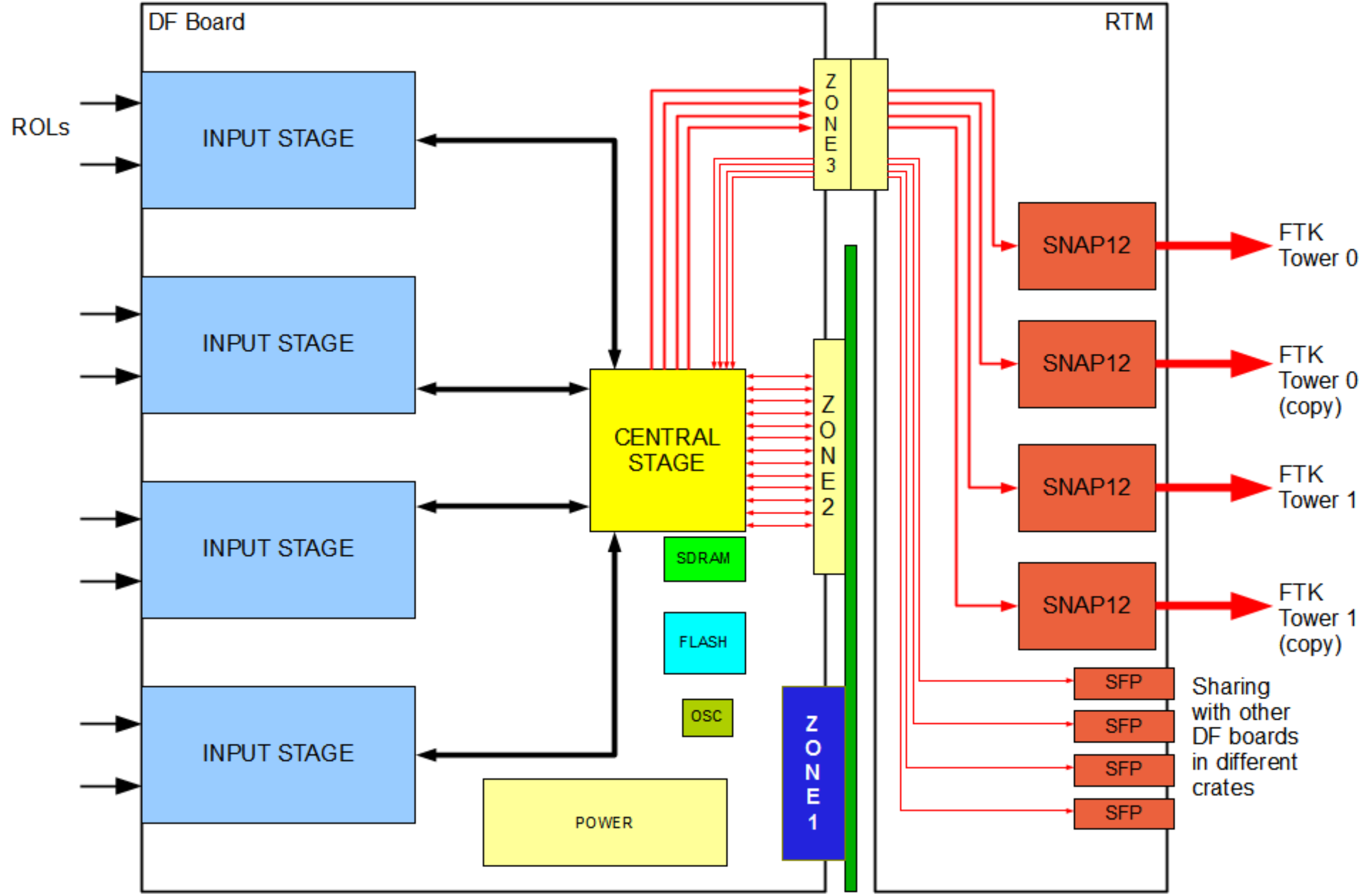


Advanced TCA

- ATCA 14 slot crate
  - 8U boards with transition modules
- Full mesh backplane
  - Each slot has 4 connections to every other slot
  - Rated for up to 40Gbps
  - “Protocol agnostic”
- Wide adoption in telecom over past 10 years
- Extremely robust and designed for high availability
  - 48VDC power, integrated cooling, hot swap, hardware management



ATCA DF Board Possibilities



Hao Xu's slide
at TIPP 2011