Array Structured Memories

STMicro/Intel
UCSD CAD LAB
Weste Text
Memory Arrays

- Random Access Memory
  - Read/Write Memory (RAM) (Volatile)
    - Static RAM (SRAM)
    - Dynamic RAM (DRAM)
  - Read Only Memory (ROM) (Nonvolatile)
    - Mask ROM
    - Programmable ROM (PROM)
    - Erasable Programmable ROM (EPROM)
- Serial Access Memory
  - Serial In Parallel Out (SIPO)
  - Parallel In Serial Out (PISO)
- Content Addressable Memory (CAM)
  - Shift Registers
  - Queues
    - First In First Out (FIFO)
    - Last In First Out (LIFO)
- Electrically Erasable Programmable ROM (EEPROM)
- Flash ROM
# Feature Comparison Between Memory Types

<table>
<thead>
<tr>
<th>Memory Type</th>
<th>SRAM</th>
<th>DRAM</th>
<th>Flash</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>Very fast</td>
<td>Fast</td>
<td>Very slow</td>
</tr>
<tr>
<td>Density</td>
<td>Low</td>
<td>High</td>
<td>Very high</td>
</tr>
<tr>
<td>Endurance</td>
<td>Better</td>
<td>Better</td>
<td>Poor</td>
</tr>
<tr>
<td>Power</td>
<td>Low</td>
<td>High</td>
<td>Very low</td>
</tr>
<tr>
<td>Refresh</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Retention</td>
<td>Volatile</td>
<td>Volatile</td>
<td>Non-volatile</td>
</tr>
<tr>
<td>Scalable</td>
<td>Good</td>
<td>Bad</td>
<td>Good</td>
</tr>
<tr>
<td>Mechanism</td>
<td>Bi-stable latch</td>
<td>Capacitor</td>
<td>FN tunneling, HCI</td>
</tr>
</tbody>
</table>

*STMicro/Intel/UCSD/THNU Memory*
The Memory Hierarchy

Latency

- Disk drives [512 B/4 kB]
- Ethernet [1 – 1500 B]
- Flash/SSD [4 kB]
- L2 cache [32/64 B]
- L1 cache [32/64 B]
- Registers [2 – 8 B]

Speed

- 2 GHz
- 1 GHz
- 500 MHz
- 100 MHz

...simply the bits that matter®
Array Architecture

- $2^n$ words of $2^m$ bits each
- If $n \gg m$, fold by $2^k$ into fewer rows of more columns

- Good regularity – easy to design
- Very high density if good cells are used
Memory - Real Organization

Array of $N \times K$ words

- $\frac{\text{Log}_2 R}{\text{Address Lines}}$
- $\frac{\text{Column Select}}{\text{Row Decoder}}$
- $S_0$
- $S_{R-1}$
- $C$ of $M$ bit words
  - row 0
  - row 1
  - row 2
  - row $N-2$
  - row $N-1$

- $\frac{\text{KxM bits}}{\text{M bit data word}}$

STMicro/Intel/UCSD/THNU

Memory
Hierarchical Memory Architecture

Advantages:
1. Shorter wires within blocks
2. Block address activates only 1 block => power savings
Array Organization Design Issues

- aspect ratio should be relative square
  - Row / Column organisation (matrix)
  - $R = \log_2(N_{\text{rows}})$; $C = \log_2(N_{\text{columns}})$
  - $R + C = N$ ($N_{\text{address\_bits}}$)
- number of rows should be power of 2
  - number of bits in a row need not be...
- sense amplifiers to speed voltage swing
- $1 \rightarrow 2^R$ row decoder
- $1 \rightarrow 2^C$ column decoder
  - $M$ column decoders ($M$ bits, one per bit)
    - $M = \text{output word width}$
Simple 4x4 SRAM Memory

2 bit width: M=2
R = 2 => N_rows = 2^R = 4
C = 1
N_columns = 2^C x M = 4
N = R + C = 3
Array size =
N_rows x N_columns = 16

STMicro/Intel/UCSD/THNU
SRAM Read Timing (typical)

- $t_{AA}$ (access time for address): time for stable output after a change in address.
- $t_{ACS}$ (access time for chip select): time for stable output after CS is asserted.
- $t_{OE}$ (output enable time): time for low impedance when OE and CS are both asserted.
- $t_{OZ}$ (output-disable time): time to high-impedance state when OE or CS are negated.
- $t_{OH}$ (output-hold time): time data remains valid after a change to the address inputs.
SRAM Read Timing (typical)

ADDR

stable

stable

stable

≥ t_{AA}

Max(t_{AA}, t_{ACS})

t_{OH}

CS_L

t_{ACS}

OE_L

t_{AA}
t_{OZ}t_{OE}

DOUT

valid

valid

valid

WE_L = HIGH

STMicro/Intel/UCSD/THNU  Memory
SRAM write cycle timing

~WE controlled

~CS controlled
SRAM Cell Design

- **Memory arrays are large**
  - Need to optimize cell design for area and performance
  - Peripheral circuits can be complex
    - 60-80% area in array, 20-40% in periphery

- **Classical Memory cell design**
  - 6T cell full CMOS
  - 4T cell with high resistance poly load
  - TFT load cell
Anatomy of the SRAM Cell

Write:
• set bit lines to new data value
  • b’ = ~b
• raise word line to “high”
  • sets cell to new state
  • Low impedance bit-lines

Read:
• set bit lines high
• set word line high
• see which bit line goes low
• High impedance bit lines

Terminology:
- bit line: carries data
- word line: used for addressing

STMicro/Intel/UCSD/THNU
SRAM Cell Operating Principle

- Inverter Amplifies
  - Negative gain
  - Slope < –1 in middle
  - Saturates at ends
- Inverter Pair Amplifies
  - Positive gain
  - Slope > 1 in middle
  - Saturates at ends
Bistable Element

Stability
- Require $V_{in} = V_2$
- Stable at endpoints
  recover from perturbation
- Metastable in middle
  Fall out when perturbed

Ball on Ramp Analogy
Cell Static Noise Margin

Cell state may be disturbed by

- **DC**
  - Layout pattern offset
  - Process mismatches
    - non-uniformity of implantation
    - gate pattern size errors

- **AC**
  - Alpha particles
  - Crosstalk
  - Voltage supply ripple
  - Thermal noise

\[ SNM = \text{Maximum Value of } V_n \]

*not* flipping cell state
SNM: Butterfly Curves
SNM for Poly Load Cell

- Q1, Q2, Q3: Saturation Region
- Q4: Linear Region

Results

\[ \text{SNM (V}_{\text{n(MAX)}}) = \frac{\sqrt{\gamma} - 1}{\sqrt{\gamma} + 1} \text{ V}_{\text{th}} + \frac{\gamma + 1 - \sqrt{2\gamma^2 + \gamma + 1}}{\gamma (\sqrt{\gamma} + 1)} (\text{V}_{\text{dd}} - \text{V}_{\text{th}}) \]

where \( \gamma \) (Cell Ratio) = \( \beta_d / \beta_a \)
12T SRAM Cell

- Basic building block: SRAM Cell
  - 1-bit/cell (noise margin again)
- 12-transistor (12T) SRAM cell
  - Latch with TM-gate write
  - Separately buffered read
6T SRAM Cell

- Cell size accounts for most of array size
  - Reduce cell size at cost of complexity/margins
- 6T SRAM Cell
- Read:
  - Precharge bit, bit\_b
  - Raise wordline
- Write:
  - Drive data onto bit, bit\_b
  - Raise wordline
SRAM Design

TI 65nm: 0.46x1.06um²

IBM 65nm: 0.41x1.25um²

Intel 65nm: 0.46x1.24um²

* Figures courtesy A. Chatterjee et al., P. Bai et al., and Z. Luo et al., Int. Electron Device Meeting Tech. Digest, 2004
Vertical 6T Cell Layout

- PMOS Pull Up
- NMOS Pull Down
- Substrate Connection
- SEL MOSFET
- N Well Connection
6T SRAM Array Layout
SRAM Bitcell Design

- Requirements of SRAM bitcell design
  - **Stable read operation**: Do not disturb data when reading
  - **Stable write operation**: Must write data within a specified time
  - **Stable data retention**: Data should not be lost

- Typical transistor sizing
  - Cell ratio \(= \frac{I(PD)}{I(PG)}\) = 1.5 ~ 2.5
  - Pull-up ratio \(= \frac{I(PU)}{I(PG)}\) = 0.5

STMicro/Intel/UCSD/THNU
Bitcell Assembly
Bitcell Array
Detailed SRAM Bitcell Layout

- Vertical: 2 poly pitch
- Horizontal: 5 contact pitch
- Poly-to-contact space > overlay + spacer + strain_layer + CD_control
  \((6.4\,\text{nm}^*) \approx 8\,\text{nm}^{**} \approx 10\,\text{nm}^{**} \approx 2.6\,\text{nm}^*\) = 27\,\text{nm}
- 1 poly pitch = 2 poly_to_contact + poly_width + contact_width
  \approx 54 + 32 + 45^{**} = 131\,\text{nm}

A pitch is a multiple of a drawing grid for fine-grain pattern placement.

\* From ITRS 32nm tech.  
\** From S. Verhaegen et al., SPIE Adv. Litho., 2008
SRAM Read

- Precharge both bitlines high
- Then turn on wordline
- One of the two bitlines will
  - be pulled down by the cell
- Ex: $A = 0$, $A_b = 1$
  - bit discharges, bit_b stays high
  - But A bumps up slightly

- Read stability
  - A must not flip
  - $N1 >> N2$
SRAM Read, 0 is stored in the cell

STMicro/Intel/UCSD/THNU

Memory
SRAM Write

- Drive one bitline high, other low
- Then turn on wordline
- Bitlines overpower cell
- Ex: A = 0, A_b = 1, bit = 1, bit_b = 0
  - Force A_b low, then A rises high
- Writability
  - Must overpower feedback
  - P2 << N4 to force A_b low,
  - N1 turns off, P1 turns on,
  - raise A high as desired
SRAM Sizing

- High bitlines must not overpower inverters during reads
- But low bitlines must write new value into cell
SRAM Column Example

**read**

**write**
Decoders

- $n:2^n$ decoder consists of $2^n$ n-input AND gates
  - One needed for each row of memory
  - Build AND from NAND or NOR gate

choose minimum size to reduce load on the address lines

A1  A0
\[\text{word} \]

\[\text{word0} \]
\[\text{word1} \]
\[\text{word2} \]
\[\text{word3} \]

static

Pseudo-nMOS

\[\frac{1}{2} \]
\[4 \]
\[16 \]

\[\text{word} \]

A0  A1
\[\text{word0} \]
\[\text{word1} \]
\[\text{word2} \]
\[\text{word3} \]

Memory
Single Pass-Gate Mux

bitlines propagate through 1 transistor
Decoder Layout

- Decoders must be pitch-matched to SRAM cell
  - Requires very skinny gates

![Diagram of decoder layout with A3 to A0 lines, VDD and GND connections, NAND gate, and buffer inverter]
Large Decoders

- For $n > 4$, NAND gates become slow
  - Break large gates into multiple smaller gates
Predecoding

- Many of these gates are redundant
  - Factor out common gates into predecoder
  - Saves area
  - Same path effort
FIG 11.14 Lyon-Schediwy decoder
STMicro/Intel/UCSD/THNU
Column Circuitry

- Some circuitry is required for each column
  - Bitline conditioning
  - Sense amplifiers
  - Column multiplexing

- Each column must have write drivers and read sensing circuits
Column Multiplexing

- Recall that array may be folded for good aspect ratio
- Ex: 2k word x 16 folded into 256 rows x 128 columns
  - Must select 16 output bits from the 128 columns
  - Requires 16 8:1 column multiplexers
Typical Column Access
Pass Transistor Based Column Decoder

- Advantage: speed since there is only one extra transistor in the signal path
- Disadvantage: large transistor count

STMicro/Intel/UCSD/THNU
Tree Decoder Mux

- Column MUX can use pass transistors
  - Use nMOS only, precharge outputs
- One design is to use k series transistors for $2^k$:1 mux
  - No external decoder logic needed
Ex: 2-way Muxed SRAM

- More Cells
- word_q1
- write0_q1
- A0
- A0
- write1_q1
- data_v1
- 2-to-1 mux
- two bits from two cells and selected by A0
Bitline Conditioning

- Precharge bitlines high before reads

- Equalize bitlines to minimize voltage difference when using sense amplifiers
Sense Amplifiers

- Bitlines have many cells attached
  - Ex: 32-kbit SRAM has 256 rows x 128 cols
  - 128 cells on each bitline
- \( t_{pd} \propto (C/I) \Delta V \)
  - Even with shared diffusion contacts, 64C of diffusion capacitance (big C)
  - Discharged slowly through small transistors (small I)
- Sense amplifiers are triggered on small voltage swing (reduce \( \Delta V \))
Differential Pair Amp

- Differential pair requires no clock
- But always dissipates static power
Clocked Sense Amp

- Clocked sense amp saves power
- Requires sense_clk after enough bitline swing
- Isolation transistors cut off large bitline capacitance
Sense Amp Waveforms

begin precharging bit lines

STMicro/Intel/UCSD/THNU
Write Driver Circuits
Dual-Ported SRAM

- Simple dual-ported SRAM
  - Two independent single-ended reads
  - Or one differential write

  wordA reads bit_b (complementary)
  wordB reads bit (true)

- Do two reads and one write by time multiplexing
  - Read during ph1
  - Write during ph2

STMicroelectronics/Select/Use/Don't Use/Memory
Multiple Ports

- We have considered single-ported SRAM
  - One read or one write on each cycle
- Multiported SRAM are needed for register files
- Examples:
  - Multicycle MIPS must read two sources or write a result on some cycles
  - Pipelined MIPS must read two sources and write a third result each cycle
  - Superscalar MIPS must read and write many sources and results each cycle
Multi-Ported SRAM

- Adding more access transistors hurts read stability
- Multiported SRAM isolates reads from state node
- Single-ended design minimizes number of bitlines
Logical effort of RAMs

Address

$2^n$ words

n-input AND

$g = \frac{n+2}{3}$

$p = n$

Decoder

SRAM Cell

$g = 2$

$p = 2^{n+1}/3$

FIG 11.25 Critical path for read of small SRAM

STMicro/Intel/UCSD/THNU Memory
Timing is non trivial design problem !!
Twisted Bitlines

- Sense amplifiers also amplify noise
  - Coupling noise is severe in modern processes
  - Try to couple equally onto bit and bit_b
  - Done by twisting bitlines
Alternative SRAM Cells

- Low Voltage/High Leakage/Process Variations crowd the operating margins of conventional SRAM

- Alternative Sense Amplifiers, column and row arrangements, adaptive timing, smaller hierarchy, redundant and spare rows/columns have all been addressed in the literature with some success.

- Some problems come from the cell design itself—modifying the cell can break conflicting demands for optimization
10T

- **Features**
  - BL Leakage reduction

- **Approaches**
  - Separated Read port
  - Stacked effect by M10

- **Performance**
  - 400mV@475kHz, 3.28uW
  - 320mV W/O Read error@27°C
  - 380mV W/O Write error@27°C
  - Vmin=300mV@1% bit errors
  - 256 bits/BL

---

A 256-kb 65-nm Sub-threshold SRAM Design for Ultra-Low-Voltage Operation

B. Calhoun & A. Chandrakasan, JSSC, 2007

STMicro/Intel/UCSD/THNU Memory
10T

- Features
  - BL leakage reduction of data
- Approaches
  - Virtual GND Replica
  - Reverse Short Channel Effect
  - BL Writeback
- Performance
  - 0.2V@100kHz, 2uW
  - 1024 bits/BL
  - 130nm process technology

A High-Density Subthreshold SRAM with Data-Independent Bitline Leakage and Virtual Ground Replica Scheme

Chris Kim, ISSCC, 2007
10T

- **Features**
  - ST cell array can work @160mV
  - 2.1x larger than 6T cell

- **Approaches**
  - Schmitt Trigger based cell
  - Good stability @ LowVDD
  - Good scalability

- **Performance**
  - Read SNM ↑ 1.56x @VDD=0.4V
  - More power saving
  - Leakage power ↓ 18%
  - Dynamic power ↓ 50%
  - Hold SNM @150mV is 2.3x of 6T
  - 130nm process

A 160mV Robust Schmitt Trigger Based Subthreshold SRAM
K. Roy, JSSC, 2007

STMicro/Intel/UCSD/THNU Memory
9T

- Features
  - Modifying from 10T cell
  - 17% more area than 6T cell
  - 16.5% less area than 10T cell

- Approaches
  - More leakage saving than 8T cell
  - Separated read port

- Performance
  - 128 bits/BL @350mV, 100MHz
  - Hold SNM=117mV @300mV
  - Stand-by power: 6uW
  - 65nm process

A 100MHz to 1GHz, 0.35V to 1.5V Supply 256x64 SRAM Block using Symmetrized 9T SRAM cell with controlled Read

STMicro/Intel/UCSD/THNU
9T

- **Features**
  - Read stability enhancement
  - Leakage power reduction

- **Approaches**
  - Separated read port
  - Min. sizing of N3, N4 and negative Vg7, and larger Node3 during stand-by mode for leakage reduction

- **Performance**
  - 2x R-SNM cf. 6T
  - 22.9% leakage power reduction
  - 65 nm PTM

High Read Stability and Low Leakage Cache Memory Cell
Z. Liu and V. Kursun, IEEE Conference, 2007

STMicro/Intel/UCSD/THNU
8T

- **Features**
  - No read disturb
  - About 30% area penalty

- **Approaches**
  - Separate Read & Write WL
  - Separated read port

- **Performance**
  - Larger SNM than 6T
  - Better scalability than 6T

Stable SRM Cell Design for the 32nm Node and Beyond
Leland Chang et. al, Symp. on VLSI 2005

STMicro/Intel/UCSD/THNU
8T

Features
- No read disturb
- Low VDD (350mV)
- Low subthreshold (Sub. Vt) leakage

Approaches
- Separate Read & Write WL
- Separated read port
- Foot-drivers reduce the sub. Vt leakage

Performance
- 65nm process, 128 cells/row
- Operating @ 25KHz
- 2.2uW leakage power

A 256kb 65nm 8T Subthreshold SRAM Employing Sense-Amplifier Redundancy
7T

- Features
  - 23% smaller than Conv. 6T bitcell
  - Low VDD (440mV)
  - Not suit for low speed demand

- Approaches
  - Separate Read & Write WL
  - Separate Read & Write BL
  - Data protection nMOS:N5

- Performance
  - 20ns access time @ 0.5V
  - 90nm process

A Read-Static-Noise-Margin-Free SRAM Cell for Low-VDD and STMicro/Intel/UCSD/THNU High-Speed Applications NEC, ISSCC, 2006
7T
- Features
  - 90% power saving
- Approaches
  - BL swing: VDD/6
- Performance
  - 0.35um process
  - *Leakage* not controlled well

90% Write Power-Saving SRAM Using Sense-Amplifying Memory Cell
T. Sakurai et. al., JSSC, 2004

STMicro/Intel/UCSD/THNU
7T

- Features
  - Low write power
  - SNM is effected by “Read pattern” (Read 0-N2,P2,N4 & Read 1-N1,P1,N3,N5)
  - 17.5% larger than 6T

- Approaches
  - Reducing write power by cut off the (feedback) connection to BL

- Performance
  - 0.18um proceso
  - 49% write power saving

Novel 7T SRAM Cell For Low Power Cache Design
R. Aly, M. Faisal and A. Bayoumi
IEEE SoC Conf., 2005
6T

- Features
  - Single-ended
  - Low VDD
- Approaches
  - Adjustable header/footer (virVDD, virGND)
- Performance
  - VDD range: 1.2V~193mV
  - Vmin=170mV with 2% redundancy

A Sub-200mV 6T SRAM in 0.13 μm CMOS

STMicro/Intel/UCSD/THNU

ISSCC, 2007

Memory
5T

- Features
  - Single-ended
  - Single BL, Single WL
  - Area 23% smaller than 6T
- Approaches
  - BL precharge to Vpc=600mV
  - Asymmetric cell sizing
  - Differential SA is used for Read
- Performance
  - 75% BL leakage reduction cf. 6T
  - SNM is 50% lower than the 6T’s
  - 0.18um process

A High Density, Low Leakage, 5T SRAM for Embedded Caches

STMicro/Intel/UCSD/THNU

I. Carlson et.al., ESSCIRC, 2004
Example Electrical Design: UCSD

32nm prototype

- Butterfly (read stability)
- N-curves (read and write stability)
- $I_{\text{read}}$ (read stability and access time)
- $V_{\text{DD}}^{\text{HOLD}}$ (data retention)
- $I_{\text{leakage}}$ (power and data retention)

- SPICE Model:
  - 32nm HKMG (high-K/metal-gate) from PTM
- Reference Design
  - Scaled bitcell from TSMC 90nm bitcell

<table>
<thead>
<tr>
<th></th>
<th>TSMC 90nm</th>
<th>32nm scaled from TSMC 90nm (REFERENCE)</th>
<th>32nm proposed (for 30x12, 25x12)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L (nm)</td>
<td>W (nm)</td>
<td>L (nm)</td>
</tr>
<tr>
<td>Pull-up</td>
<td>100</td>
<td>100</td>
<td>32</td>
</tr>
<tr>
<td>Pull-down</td>
<td>100</td>
<td>175</td>
<td>32</td>
</tr>
<tr>
<td>Pass-Gate</td>
<td>115</td>
<td>120</td>
<td>37</td>
</tr>
</tbody>
</table>
Butterfly and N-Curves

- Measure method
  - Increase VR and measure VL
  - Increase VL and measure VR
  - Make voltage transfer curve in VR and VL axes → Butterfly
  - Measure $I_{in}$ → N-curve
\( I_{\text{read}}, I_{\text{leakage}} \text{ and } VDD_{\text{HOLD}} \)

- \( I_{\text{read}} \)
  - Measure bitline current when WL switches to high

- \( I_{\text{LEAKAGE}} \)
  - Measure VDD (or VSS) current when WL=0

- \( VDD_{\text{HOLD}} \)
  - Decreasing VDD voltage, while WL=0
  - Measure minimum VDD voltage when \( |V(\text{nl}) - V(\text{nr})| = \) ‘sensing margin’
    
    \( (100\text{mV is assumed}) \)

<table>
<thead>
<tr>
<th>REFERENCE</th>
<th>REFERENCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>( I_{\text{read}} )</td>
<td>41.2 \text{uA}</td>
</tr>
<tr>
<td>( I_{\text{leakage}} )</td>
<td>85.4 \text{nA}</td>
</tr>
<tr>
<td>( VDD_{\text{HOLD}} )</td>
<td>110 \text{mV}</td>
</tr>
</tbody>
</table>

STMicro/Intel/UCSD/THNU Memory
Corner Simulation: Butterfly and N-Curve

(SS, 125degC, 1.0V) (NN, 25degC, 1.0V) (FF, -40degC, 1.0V)

STMicro/Intel/UCSD/THNU Memory
Corner Simulation: $I_{\text{read}}, I_{\text{leakage}}$ and $VDD_{\text{HOLD}}$
# Processor Memory Bandwidth/Pins

<table>
<thead>
<tr>
<th>Year</th>
<th>Brand name</th>
<th># cores</th>
<th>Pin count</th>
<th>FSB or QPI, GB/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>Pentium-III</td>
<td>1</td>
<td>370</td>
<td>0.5</td>
</tr>
<tr>
<td>2001</td>
<td>Celeron</td>
<td>1</td>
<td>479</td>
<td>1.1</td>
</tr>
<tr>
<td>2002</td>
<td>Xeon 1.6</td>
<td>1</td>
<td>603</td>
<td>3.2</td>
</tr>
<tr>
<td>2003</td>
<td>Xeon 3.2</td>
<td>1</td>
<td>603</td>
<td>4.3</td>
</tr>
<tr>
<td>2004</td>
<td>Xeon 3.6</td>
<td>1</td>
<td>604</td>
<td>6.4</td>
</tr>
<tr>
<td>2005</td>
<td>Xeon 7030</td>
<td>2</td>
<td>604</td>
<td>6.4</td>
</tr>
<tr>
<td>2006</td>
<td>Xeon X5470</td>
<td>4</td>
<td>771</td>
<td>10.7</td>
</tr>
<tr>
<td>2007</td>
<td>Xeon X7350</td>
<td>4</td>
<td>604</td>
<td>8.5</td>
</tr>
<tr>
<td>2008</td>
<td>Core i7-920</td>
<td>4</td>
<td>1366</td>
<td>19.2</td>
</tr>
<tr>
<td>2009</td>
<td>Core i7-950</td>
<td>4</td>
<td>1366</td>
<td>19.2</td>
</tr>
<tr>
<td>2010</td>
<td>Core i7-970</td>
<td>6</td>
<td>1366</td>
<td>25.6</td>
</tr>
<tr>
<td>2011</td>
<td>Core i7-3960X</td>
<td>6</td>
<td>2011</td>
<td>25.6</td>
</tr>
</tbody>
</table>
GPU Bandwidth/Core (local and PCIe)
Why 3D DRAM?

![Diagram of 3D DRAM structure with TSV connections and layers.](image)

- **Projected improvement to DRAM of 3-D interconnections**
  - Bandwidth: 800%
  - Power Consumption: 50%
  - Size Reduction: 35%

STMicro/Intel/UCSD/THNU