MARAM

Novel Memory Architecture for System Level Integration

David Sherman
VP Engineering
Alpine Microsystems, Inc
dsherman@alpinemicro.com
Presentation Agenda

• Background: Introduction to System Level Integration
  – Overview of base technology
  – Distinction from previous modular technology
  – Physical & Electrical requirements & characteristics

• Justification for MARA™ concept & DieCore™

• Architecture of the MARA SM & the MARA DM

• System Uses for MARA Memories

• Conclusion, Status of SLI Initiatives
System Level Integration

• **What is it?**
  – Up to 8 (or more) high value die on a single high quality Silicon modular assembly. Complements and extends SOC capability.

• **What it is not:**
  – “Multi-chip package/System in Package”. MCP/SiP is usually 2-3 smaller standard ICs with perhaps a few passives, on a lower quality organic laminate
  
  – Traditional “MCM-D multi-chip modules”. These suffer from compound yield problems, difficult test, fabrication & repair = $$$$
Alpine MicroBoard™ Architecture

MicroBoard: Low Profile Multi-Die Silicon BGA
MicroBoard Integration Advantage

• MicroPallet™ carrier yields like a packaged part
  – Use package part test vectors,
  – Perform burn-in before assembly of complete MicroBoard
  – Break up and distribute complex routing problems
  – No compound yield loss (final flip-chip assembly is >>99% yield)

• Mix & match different process technologies,
  – DRAM, logic, SiGe, Flash, GaAs, ........
  – Eliminate “Frankenstein process” which integrates these on single die
  – Each IC (or DieCore) is separately processed, tested and yielded
    (Contrast with SoC: Process cost & yield hit of complex process is
     shared by every core for SoC)
DieCore Memory Attributes

- **Flexibility**
  - Small granularity similar to embedded cores
  - multi-porting & multi-banking for many data consumers

- **High bandwidth & Low latency**
  - Wide words to achieve bandwidth similar to embedded memories. Latency should be comparable to embedded.

- **Low power**
  In both embedded & co-designed cases proximity leads to lower capacitance & therefore lower power.
MARA Static Memory (SM) Architecture

- **DQA[35:0]**, **DQB[35:0]**, **DQC[35:0]**, **DQD[35:0]**, **DQE[35:0]**, **DQF[35:0]**
- **PortA[35:0]**, **PortB[35:0]**, **PortC[35:0]**, **PortD[35:0]**, **PortE[35:0]**, **PortF[35:0]**
- **Serial 2 Parallel, Parallel 2 Serial Bank0**
- **Serial 2 Parallel, Parallel 2 Serial Bank1**
- **Bank0 Din**, **Bank0 Dout**, **Bank1 Din**, **Bank1 Dout**
- **Bank0_Addr**, **Bank1_Addr**
- **16k x 288 BANK0**
- **16k x 288 BANK1**
- **Address & command scheduling and arbitration**

**Port**
- PortA[35:0]
- PortB[35:0]
- PortC[35:0]
- PortD[35:0]
- PortE[35:0]
- PortF[35:0]
MARA Static Memory Key Features

• Flexible port configuration
  – configure I/O’s - 6x36 or 3x72 or 1x144 + 2x36 or ....

• High Bandwidth
  – Slow corner 500MHz Clk, 1Gbit data DDR
  – Typical corner 650Mhz Clk, 1.3 GBit DDR

• Low Latency:
  – 2 Clk latency to return of data burst
  – Port speed flexibility - “Bandwidth Impedance matching”
    Run each data port @ 2x, 1x, 1/2x, 1/3x Clk independently
    Burst length 1,2,4,8 per command, per port

• Backward compatibility modes:
  – Configure as Quad port “QDR-like” with snoop,
    72 wide triple port DDR, 72 wide triple port ZBT, etc
MARA Static Memory Key Features (2)

- No Bus & Bank contention
  - Variable late write collision-less QUAD port operation. Any address can control any DQ port
- Low Power
  - ~1.3W (estimate) max total power @ 500MHz (1Gb), .5W in core for typical .18u process - fully utilized banks
     216 I/O’s w predrivers & load consumes another .8W peak
- Low Noise I/O:
  - Small swing, HSTL-like. No static termination power, CD/CQ source synchronous strobes
- High testability:
  - JTAG 1149.1 Boundary Scan w MemBIST for in-situ self testing for “always on” networking cards
- Pass-thru feature:
  - Snooping traffic, immediate shared data between processes
MARA Dynamic Memory (DM) Architecture
High Speed DDR Compatible DRAM

Note: This is a block diagram, drawing is for architectural clarity. The bus & Xbar logic actually imposes only a small die size penalty due to Alpine bump density.
MARA Dynamic Memory Key Features

• 64 Mbit
  – Small die to match embedded DRAM granularity
• Flexible I/O
  – QUAD 16 or 32 bit ports, dual 64, one 128 bit port
• Flexible Core
  – 16 banks, in 8 groups of 2.
  – Each port gets entire memory. Or, Each 32 bit port gets it’s own quad bank 16Mbit memory.
• Quad independent address/control port
  – Each data port gets it’s own control bus for maximum design utility
• High Bandwidth
  – 128 bits wide to match embedded width - 12.8 GByte/Sec/part
  – typical bin is 800 Mb DDR data, i.e 8x Rambus Direct per part
• Low Latency: 21 nS to random data, <30nS random cycle
Dual NPU w “Super-QDR”

DDR Ingress Data from switch fabric or previous NPU @ ? clk

D_A3[31:0]

D_A2[31:0]

Address3(to SRAM 2&3)

DDR Egress Data to switch fabric or next NPU @ ? clk

Q_B2[31:0]

Q_B3[31:0]
Save 8 SRAMs from your design

NPU PIPELINE that operates on basically identical data in 3 (or more) ZBT SRAM banks, as an example

4 Mara SM DieCores in ZBT compatible mode

12 die -> 4 die, Module saves many sq inch PCB

SAME NPU PIPELINE
Quad NPU module

DDR A → DDR B

MaraDM DieCore3 → MaraDM DieCore4

MaraDM DieCore5 → MaraDM DieCore6

MaraDM DieCore7

DDR C

DDR D

DieCore1

DieCore2

DieCore3

DieCore4

DieCore5

DieCore6

DieCore7

NPU 1

NPU 2

NPU 3

NPU 4

Mips

Mips

Mips

Mips

Shared L2

Shared L2

72

72

80

16
Quad NPU module (2)

It is now possible to economically design a 2 NPU 1 Die SOC w 4 MBit shared L2 cache. (Someone has)

But that also means,

It is now possible to economically design a 4 NPU 7 Die SLI w 320 MBit shared L3 cache. (Someone will, is it you?)

This illustrates a fundamental point.....

SLI technology gives a permanent 1-2 generation advantage to those who use it
**Generic SLI Standard Parts**

CPU Die is co-designed with Mara SM & Mara DM to optimize the cache sizes on die. For example larger L1 on die, Mara SM is L2, Mara DM is L3 & working memory.

Programmable Die stays completely flexible, no guessing at correct memory core size. Not tied to particular CPU architecture.
I know you have been wondering………

**Multiported**  Flexibility at system level

**Arrayed**  Small, distributed multi-ported macros

**Random**  Non-blocking access from each port

**Access**  Wide & low latency

**Memory**  1st DieCores are memories because most generally useful.

**9M MARA Static Memory** is scheduled for tapeout Sept 2001. Samples in Q4.

18Mbit version samples in Q1.

**64M MARA Dynamic Memory** design with major DRAM partner, Samples Q1
Conclusion

DieCore system partition is

- much cheaper,
- more functional,
- superior time to market when compared to “pure” SOC option.

System Level Integration is a strategy used by

- successful companies to bring cutting edge products to market
- on time, in an era of tight R&D budgets.