Sixth ACM-IEEE International Conference on Formal Methods and Models for Codesign (MEMOCODE'2008)

June 5-7, Anaheim, CA, USA

MEMOCODE '08

Design Contest - Submitted by

Nanyang Technological University, NTU

Singapore

Aung. Y.L, Chuong. L.M, Chandrasekaran. S, Jin. C, Linh. D.H, Prakash. A, Rajarathinam. M.A, Yupeng. C

NANYANG TECHNOLOGICAL UNIVERSITY

Resources:

- Performance Normalization Factor : 1.0
- Xilinx XUP Virtex™-II Pro Development System
- 100MHz System Clock
- 300MHz PPC Clock
- EDK 9.1
- ISE 9.1, ChipScope Pro 9.1i, ModelSIM
- + 2 weeks of Design/System Exploration
- + 2 weeks of Implementation

NANYANG TECHNOLOGICAL UNIVERSITY

Result:

df - HyperTerminal

File Edit View Call Transfer Help

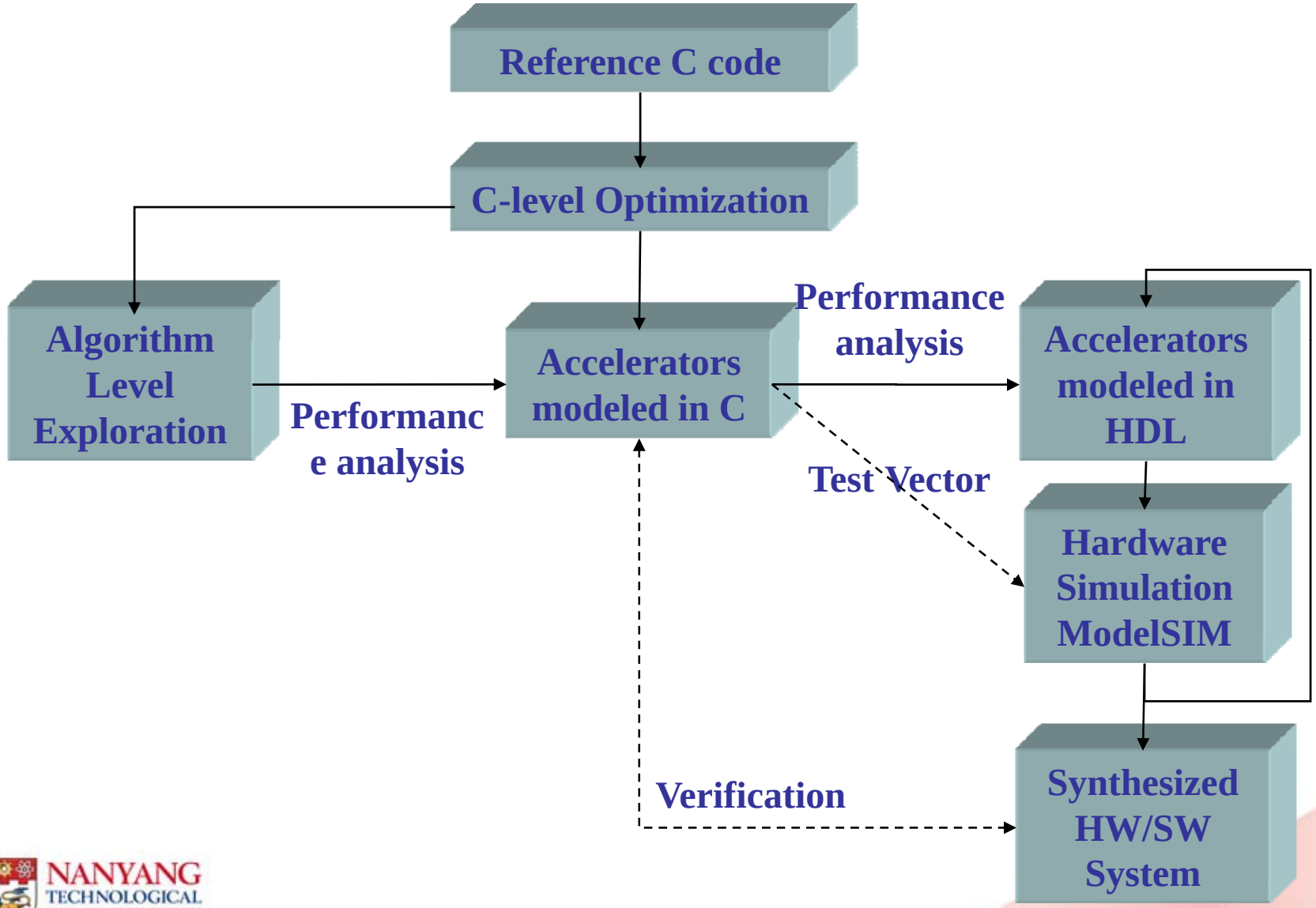
```
Sorting starts
Case (rand, pwr=  6) : Elapsed       193, speed up 27.145079,  correct
Case ( rot, pwr=  6) : Elapsed       201, speed up 28.427860,  correct
Case (rand, pwr= 10) : Elapsed      4520, speed up 39.267700,  correct
Case ( rot, pwr= 10) : Elapsed      4377, speed up 40.223896,  correct
Case (rand, pwr= 14) : Elapsed     94330, speed up 46.767433,  correct
Case ( rot, pwr= 14) : Elapsed     70858, speed up 60.684029,  correct
Case (rand, pwr= 18) : Elapsed   1960018, speed up 49.889465,  correct
Case ( rot, pwr= 18) : Elapsed   2078183, speed up 47.133228,  correct
Relative Geometric Mean: 41.094964
Sorting completed
_
```

MAX SPEED UP: 60X, AVERAGE: 41X

Connected 0:46:50 | Auto detect | 9600 8-N-1 | SCROLL | CAPS | NUM | Capture | Print echo

NANYANG TECHNOLOGICAL UNIVERSITY

Approach:



NANYANG TECHNOLOGICAL UNIVERSITY

Concept:

QUIKOM SORT (Combine Quick and Comb Sort)

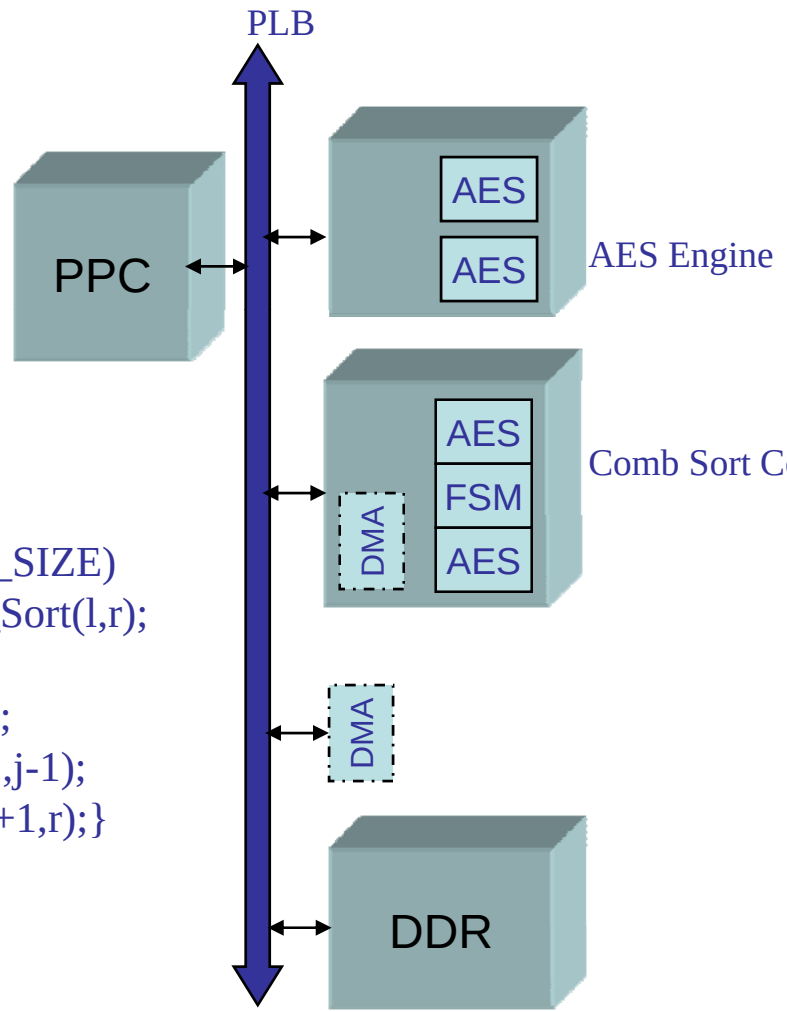
- Increase System's Parallelism by Performing Quick Sort first.
- Hardware Comb Sorters sort the partitioned intervals independently
  - Quikom Sort (Pseudo Code): On Power PC

```
Quikom_sort(l,r)
{
If (l<r)
{
If ((r-l)< COMB_SIZE) Do_HW_Comb_Sort(l,r);
  else {
  j = Partition(l,r);
  Quikom_sort (l,,j-1);
  Quikom_sort (j+1,r);}
}}
```

- Software Quick Sort is accelerated by Hardware AES Core
- Do_HW_Comb_Sort function feeds the data to HW sorters and returns

NANYANG TECHNOLOGICAL UNIVERSITY

System Architecture:



PPC:

```
Quikom_sort(l,r)
{
If (l<r)
{
If ((r-l)< COMB_SIZE)
Do_HW_Comb_Sort(l,r);
  else {
  j = Partition(l,r);
  Quikom_sort (l,,j-1);
  Quikom_sort (j+1,r);}
}}
```

AES Engine:

- Average 320 ns per AES compared to 2400 ns in SW
- Speed up 8x
- AES Core taken from www.opencores.org

Comb Sort Co-processor:

- 4K records local memory
- Sort 4k of records in ~ 17ms
- 2 Sorters in final version
- **Attached PLB-DMA Controller is unstable**

NANYANG TECHNOLOGICAL UNIVERSITY

Performance:

FPGA Resources Used:

Comb sort with HW AES

- 50% SLICES
- 10%BRAM
- **Speedup - 6X**

Quick sort with HW AES:

- 50% SLICES
- 10%BRAM
- **Speedup - 10X**

1 comb sorter:

- 80% of SLICES
- 30% of BRAM
- **Speedup - 35X**

2 comb sorters:

- 100% of SLICES
- 52% of BRAM
- **Speedup - 41X**

NANYANG TECHNOLOGICAL UNIVERSITY

- **Execution Time Distribution:**
  - Quick Sort's Partitioning: 40%
  - Data Transfer: 30% (10% for during Quick Sort and 20% during Comb Sort)
    - EDK's DMA controller: Max 160MB/sec
    - When data > 8kB → transfer fails :-/
  - Comb Sort: 30%
    - Bigger local memory of Comb-Sorter will reduce Partitioning time
    - Comb Sort Time is almost constant for each MAXRECORD
- **Investigations and Bottlenecks:**
    - EDK's PLB-DMA set-up did not help much
    - More Comb Sort Co-Processors would help
    - Better Scheduler would help in performance increase

NANYANG TECHNOLOGICAL UNIVERSITY

THANK YOU!

NANYANG TECHNOLOGICAL UNIVERSITY