Using FPGAs in Supercomputing – Reconfigurable Supercomputing

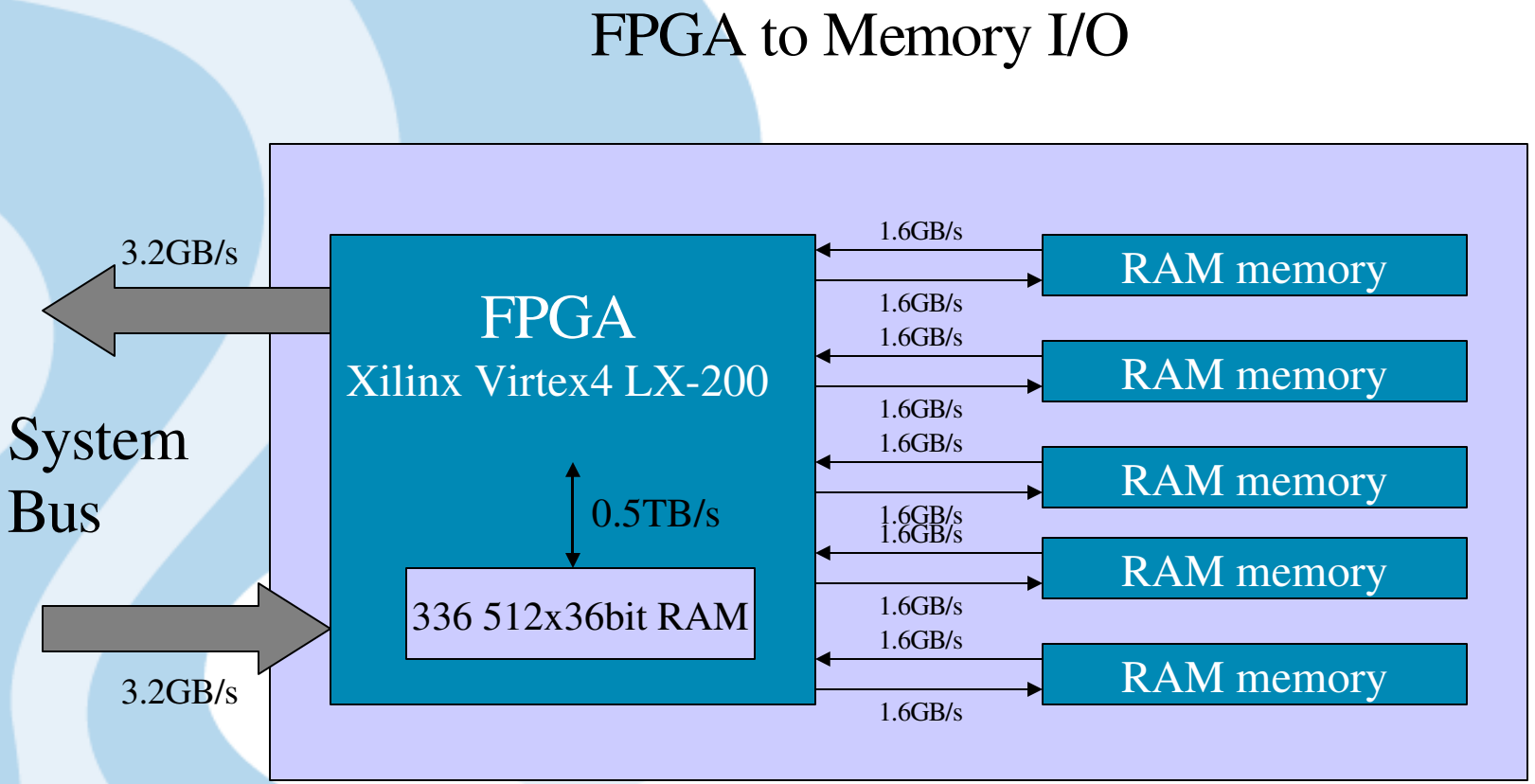
mitrion

Why FPGAs?

- FPGAs are 10-100x faster than a modern Itanium or Opteron
  - Performance gap is likely to grow further in the future
- Several major vendors now have FPGA modules
  - Cray XD1 in 2004, soon on the XT3/XT4
  - Silicon Graphics launched RASC in 2005
  - Linux Networx announced in June, launching in a few months
  - Others in progress…

mitrion

A typical FPGA Module



FPGAs

- Empty re-configurable silicon surface
- Compared to fixed silicon at the same process technology (90nm):
  - ~10 times slower clock frequency
  - ~100 times larger area used per gate
- Compared to CPUs
  - 10-100 times faster

mitrion

Programming a Field Programmable Gate Array

- Without a circuit design, an FPGA is just an empty silicon surface
  - What is meant by “Programmable” in the acronym “FPGA” is “a circuit design can be loaded”
- Designing a circuit is not “programming” from a software developer’s point of view
  - Not suitable for supercomputing application developers (biologists, astronomers, etc)

mitrion

mitrion

Reconfigurable Supercomputing: Three new Directions

Three new directions: Libraries

- Hand crafted full-FPGA circuits for specific tasks
  - Called as library routines - don't touch legacy source code
- Bandwidth limitations
  - Sending and receiving arguments
  - Reconfiguration of the FPGA
  - Very hard to overlap transfers automatically
- Circuit design is difficult
  - No libraries yet

mitrion

Three new directions: Bottom Up

- Raise the abstraction level of classic circuit design methods
  - Behavioral Synthesis
- The Holy Grail of circuit design – has been researched for over 20 years
  - Several circuit design tools have come a long way
  - Hard to get good performance

mitrion

Three new directions: Top Down

- Create a circuit design that can run software - Processors
  - Micro-blaze and Nios won't cut it
  - Configurable processors
- FPGAs require a new Processor Architecture
  - Must be automatically adaptable for the program it will run
  - Must be massively parallel

mitrion

mitrion

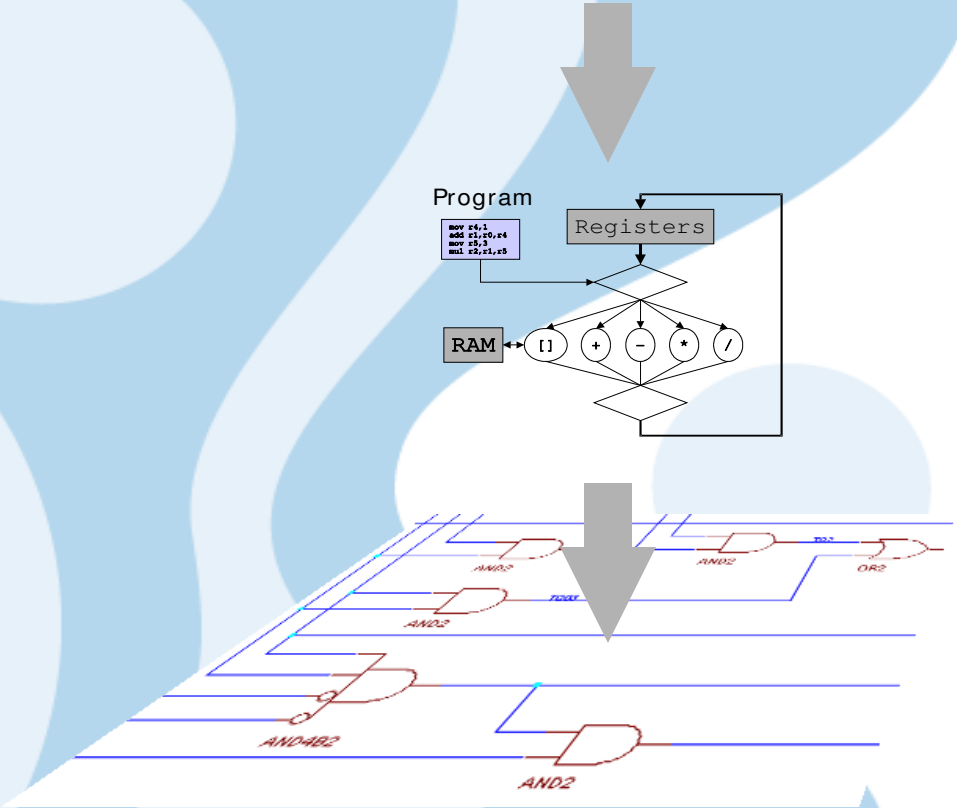
The Mitrion Processor Architecture

The von Neumann processor architecture

```
int:48<30> main()
{
  int:48 prev  = 1;
  int:48 fib   = 1;

  int:48<30> fibonnacci = for(i in <1..30>)
  {
    fib  = fib+prev;
    prev = fib;
  } <>fib;

} fibonnacci;
```



The von Neumann architecture is based on a State Machine. It is characterized by operating on one instruction at a time from an instruction stream.

+ Easily programmable
+ Executes programs of any size
– Sequential; works on an instruction stream
– Low silicon utilization
– I/O intensive
➢ Needs very high clock frequency

mitrion

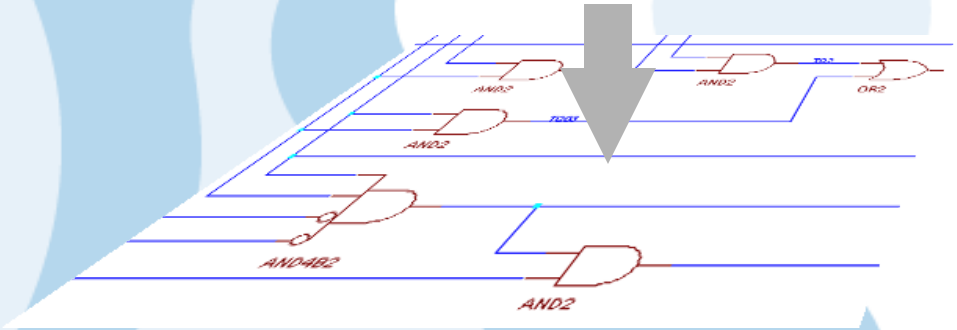
The Purpose Of A Processor Architecture

```
int:48<30> main()
{
  int:48 prev  = 1;
  int:48 fib   = 1;

  int:48<30> fibonnacci = for(i in <1..30>)
  {
    fib  = fib+prev;
    prev = fib;
  } <>fib;

} fibonnacci;
```

?



- A processor is an abstraction layer
- A machine, built in hardware that performs your program, written in software
- Completely separates hardware from software
  - ISA: branches, arithmetic operations, indirect indexing, status registers, etc, etc
  - Circuits: gates, flip-flops, clocks, PLLs, UCFs, etc, etc

mitrion

Processor Architecture: A Cluster-On-A-Chip

- Traditional Clusters:
  - A set of nodes connected in a fixed network
  - Network latency on the order of many thousands of clock-cycles
    - ➢ Each node needs to run a block of code to mask network latency

mitrion

Processor Architecture: A Cluster-On-A-Chip

- Network topology specific for algorithm
  - Ideal properties
- Very Fine-Grain Parallelism
  - A single instruction on each node
  - Each node adapted to run its instruction

- Fully non-von Neumann architecture
- The architecture replaces sequential instruction scheduling with parallel packet switching

mitrion

Mitrion Tools Do Not Allow Circuit Design

- The Mitrion tools can only program and configure the Mitrion Virtual Processor architecture
  - The tools can not be used to create circuit designs
- Since circuits can not be designed, no errors regarding circuit design can be made
- The Mitrion Processor lets you run software in your FPGA

mitrion

Mitrion Offers Portability And Scalability

- You program the processor, you do not design a circuit for a specific FPGA platform

Just configure a processor for the new platform from your old source-code.

- Easy upgrades to the next generation of performance available.
- Exchange code with colleagues on other platforms

mitrion

The Mitrion Platform

1) The Mitrion Virtual Processor
   - A configurable processor design for a fine-grain massively parallel, soft-core processor
   - 10-30 times faster than traditional CPUs

2) The Mitrion-C programming language
   - An intrinsically parallel C-family language

3) The Mitrion Software Development Kit
   - Compiler
   - Debugger/Simulator
   - Processor configurator

mitrion

mitrion

Running Software in FPGAs

Mitrion-C

- The Mitrion Processor needs a fine-grain, fully parallel programming language
  - Parallel at the level of individual instructions
- A C-family language – but not ANSI-C!
  - Steeper learning curve, but better in the long run
- The automatically parallelising C or Fortran compiler doesn't exist anyway
  - The Holy Grail of parallel computing for 20 years

mitrion

Mitrion-C

- Does not describe Order-of-Execution. Instead it describes Data-Dependencies
- Syntactic support to make design-decisions regarding parallelism salient
  - Designed to *preserve* parallelism in the algorithm
  - Designed to *reveal* parallelism in the algorithm
- Allows a complete data-dependency graph to be created of the algorithm

mitrion

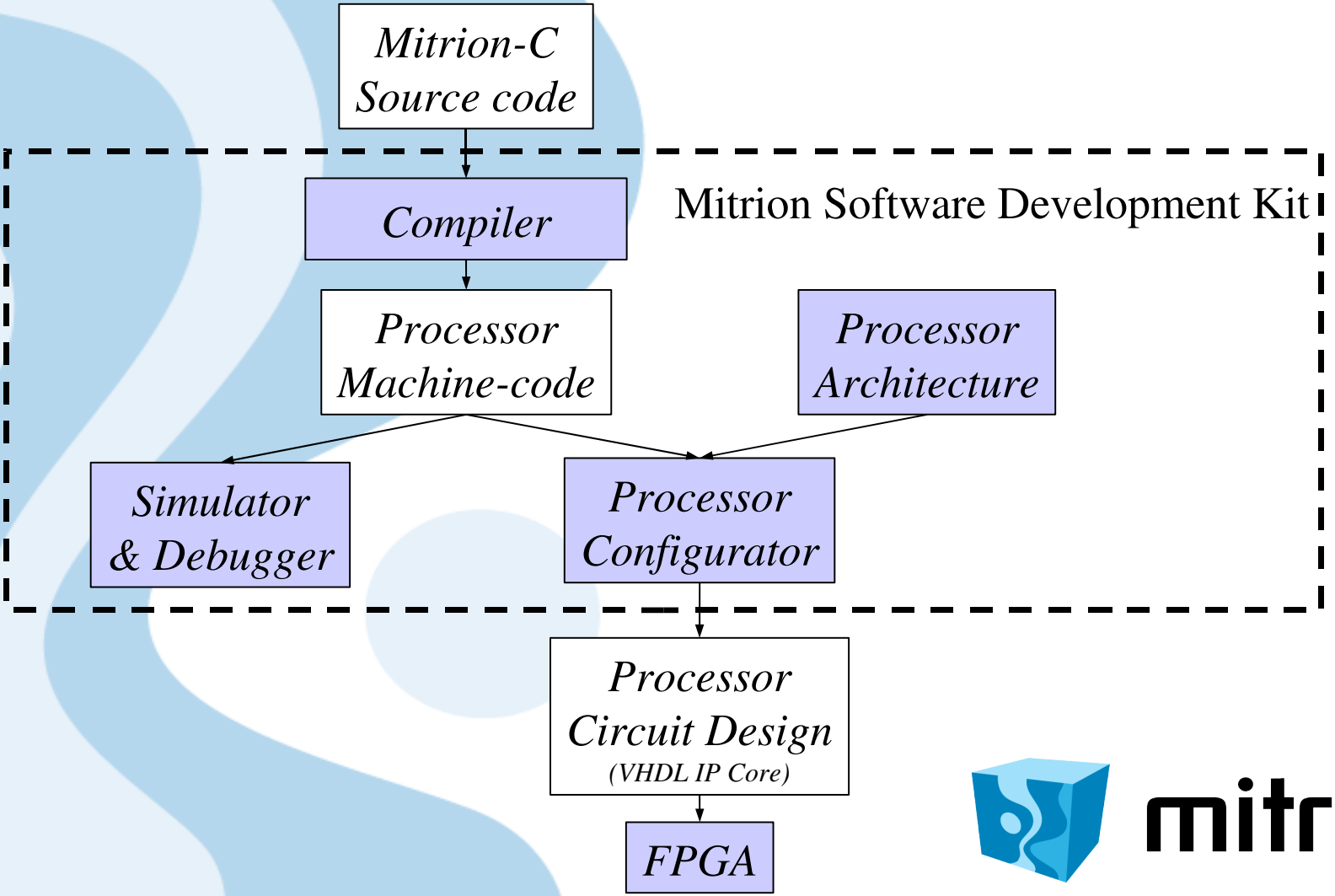
Applying the Mitrion Platform

1) Identify the time-critical portion of the program
2) Write it as code in Mitrion-C
3) Perform a function call to run your Mitrion-program

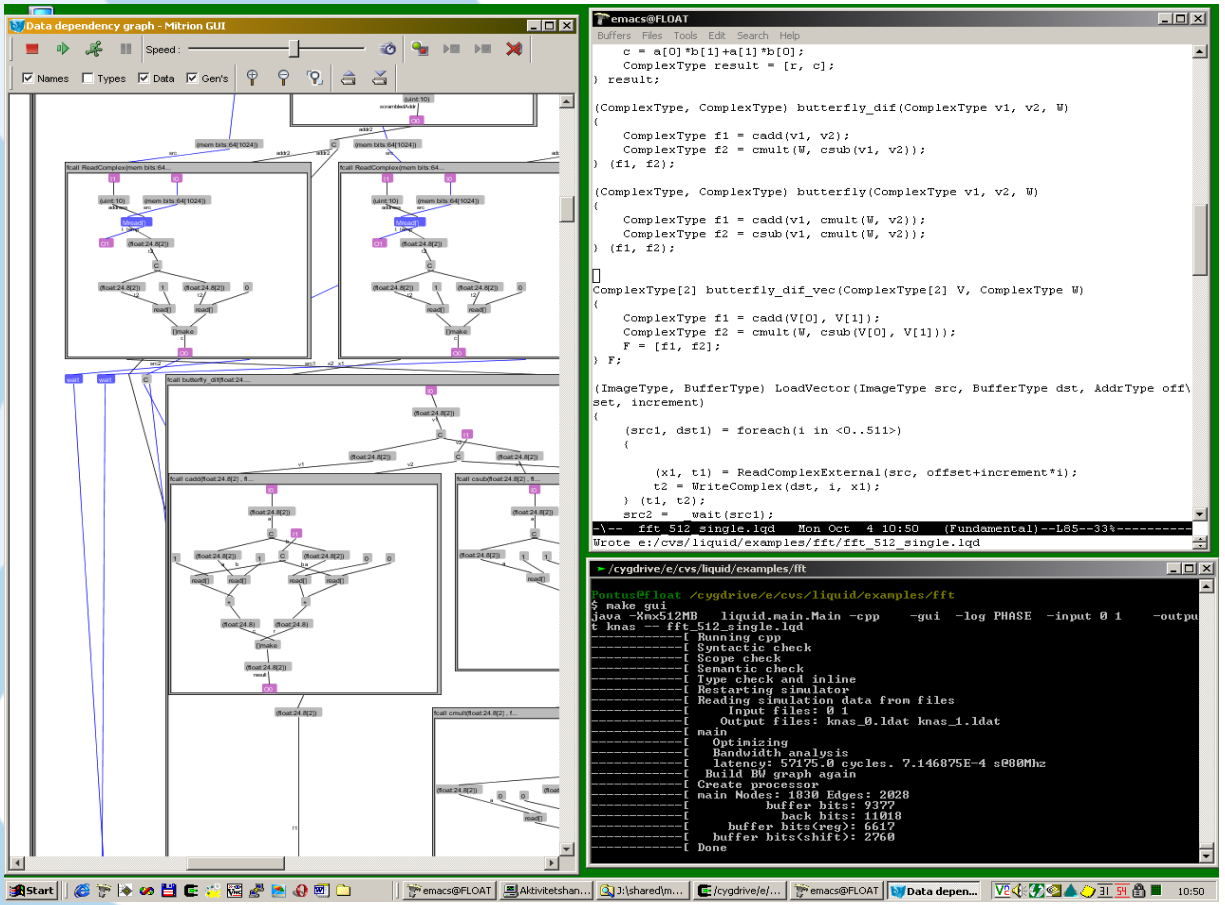
➢ Most of your code is unchanged

mitrion

Compiling A Mitrion Program



Compiler, Simulator And Debugger



Contact Information

Mitrionics AB
Ideon Science Park
SE-223 70 Lund
www.mitrionics.com

stefan@mitrionics.com

mitrion