



Ilmenau Technical University
Faculty of Computer Science and Automation
Department of System and Control Theory

Software Performance Estimation for a Mission Level Design Flow

Date: 28.09.2004

Author: Thomas Lohfelder

Contents

- Introduction
- Design Flow in MLDesigner
- Software Performance Estimation
- Example
- Outlook

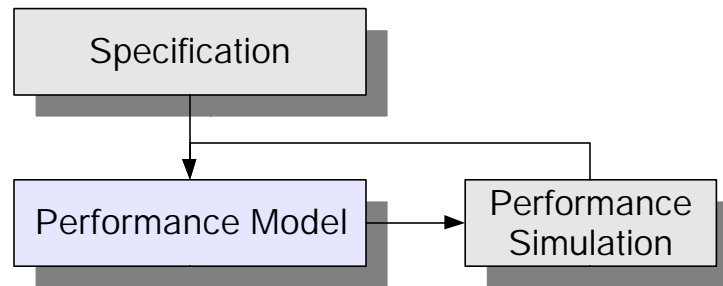
Introduction

- Electronic mixed hardware/software systems are raising in complexity at an exponential rate
- Time-to-market should be decreased while the quality should stay the same or should become better
- Therefore, several computer-aided design methodologies for detailed modeling of such systems (VHDL, C Code) were introduced in the last years

Introduction

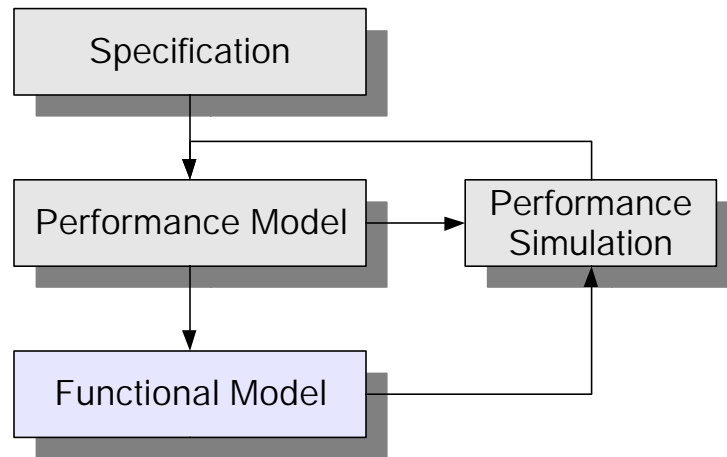
- Nowadays the reached complexity makes detailed modeling impossible
- Major problems rising with this complexity
 - Immense modeling effort
 - Simulation speed
- Solution:
 - Raise level of abstraction without loosing too much of the accuracy

Design Flow in MLDesigner



- Performance Model:
 - Describes the architecture
 - Software is modeled as simple delays

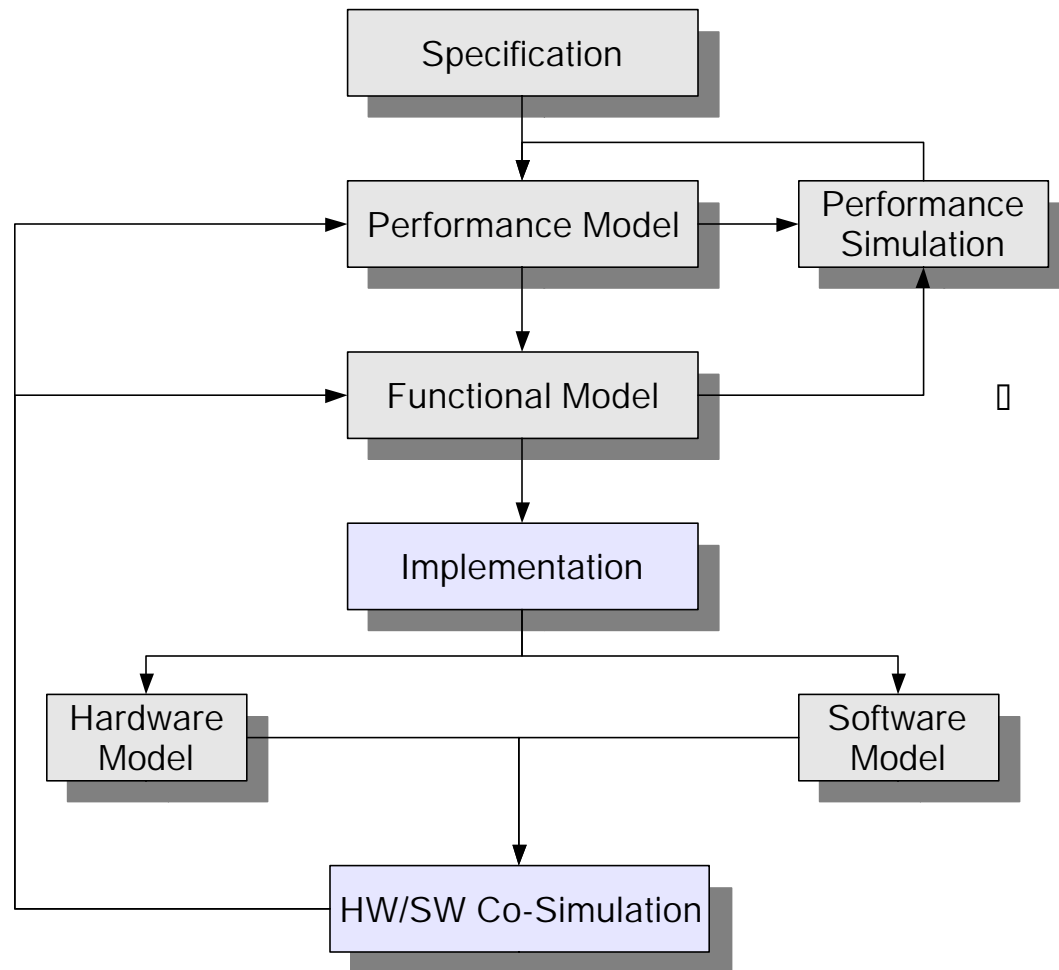
Design Flow in MLDesigner



▫ Functional Model:

- Describes the software
- Independent of level of abstraction

Design Flow in MLDesigner



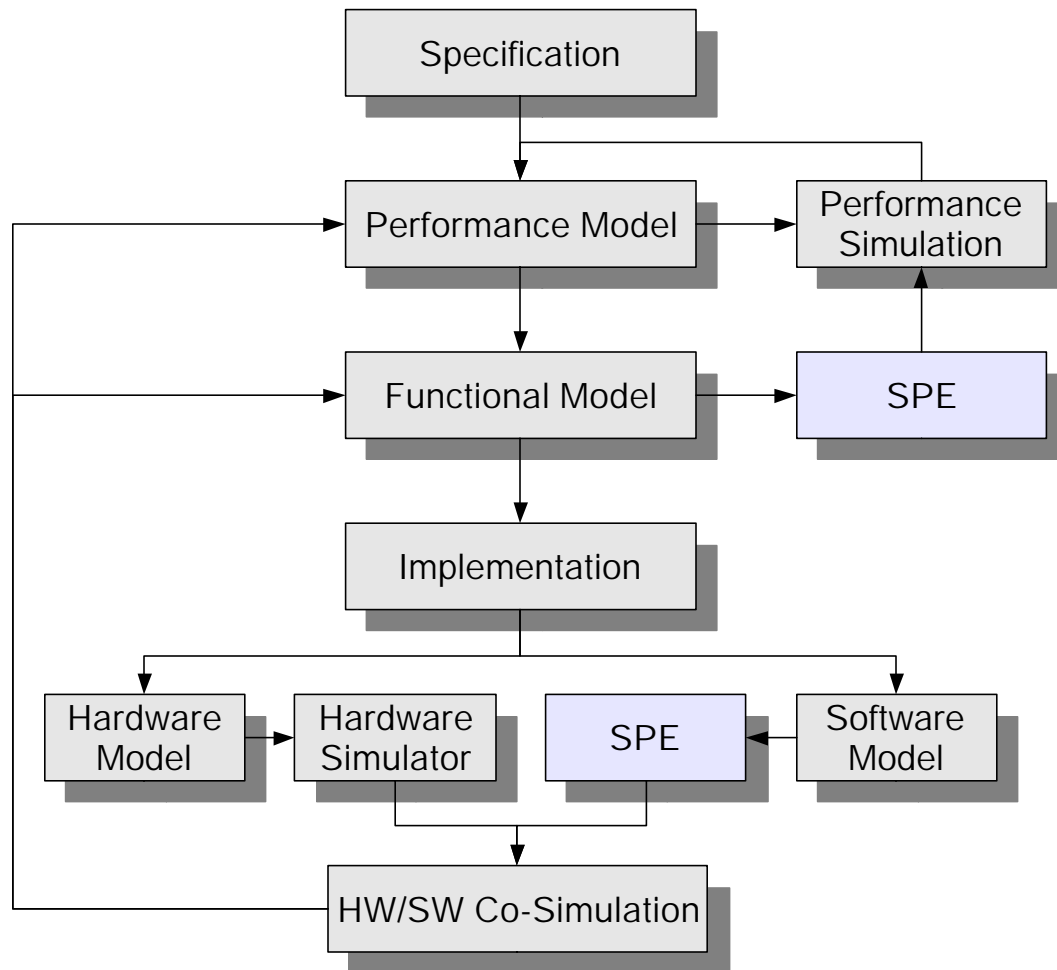
Implementation:

- HW Model: HW Description Language
- SW Model: Source Code

HW/SW Co-Simulation:

- Concurrent Simulation of HW and SW

MLDesigner Design Flow



- Software Performance Estimation:
 - Takes Source Code
 - Estimates Cycles Counts

Software Performance Estimation

- Definition
- Approaches
- General Work Flow
- Work Flow in MLDesigner

Definition

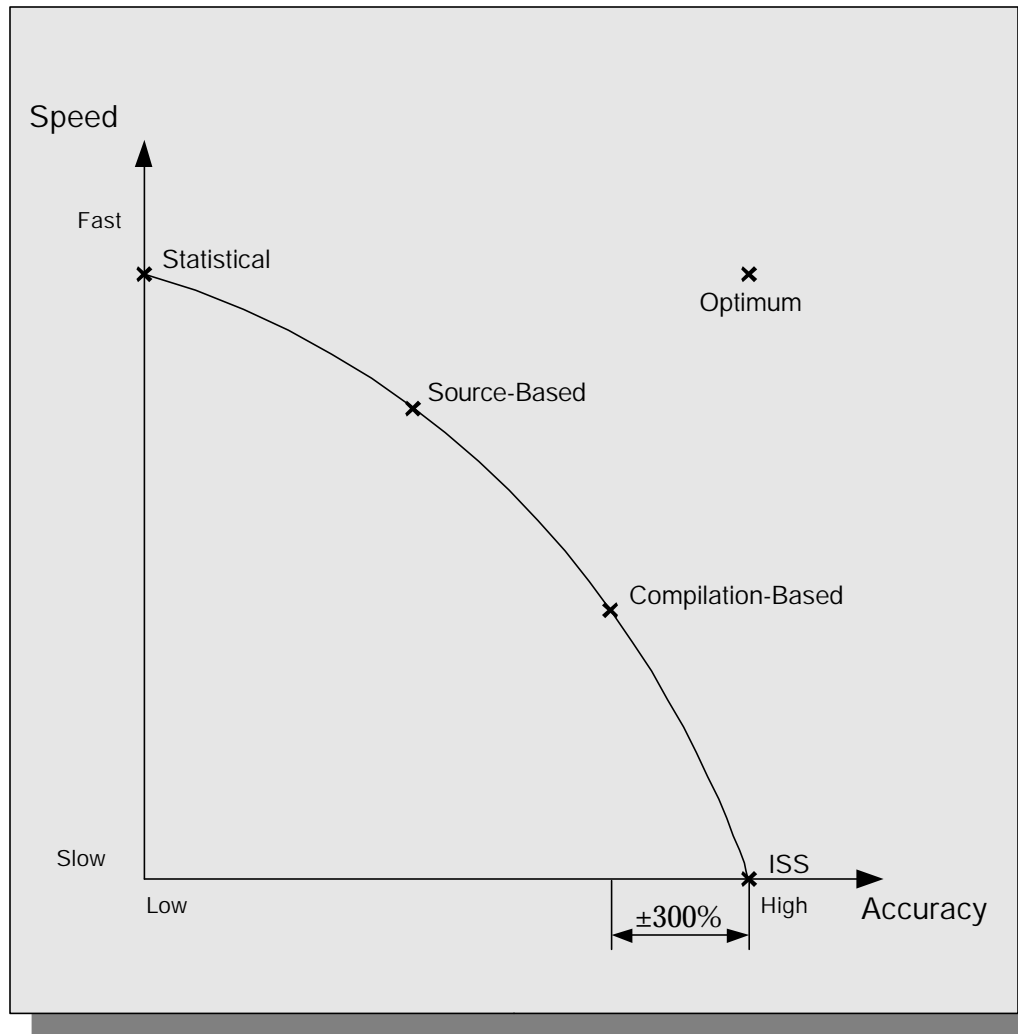
Software Performance Estimation

- Is the estimation of the performance of a piece of software running on a specific target processor
- Requirements: Source code must exist
- Result: Cycle count

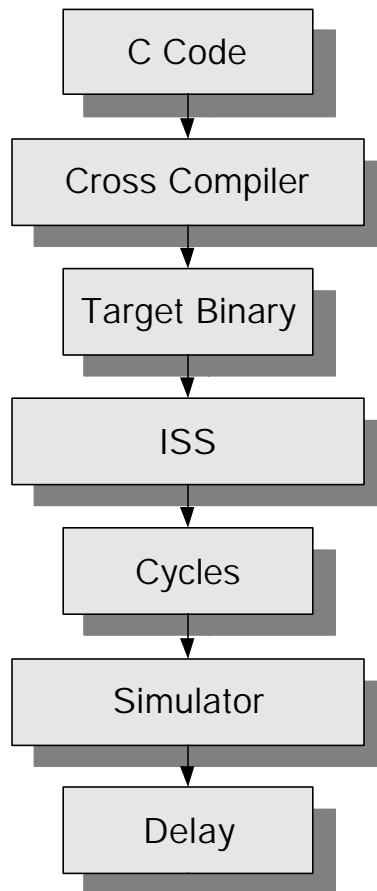
Approaches

- Statistical Estimation
- Source-Based Estimation
- Compilation-Based Estimation
- Estimation using Instruction Set Simulators

Approaches

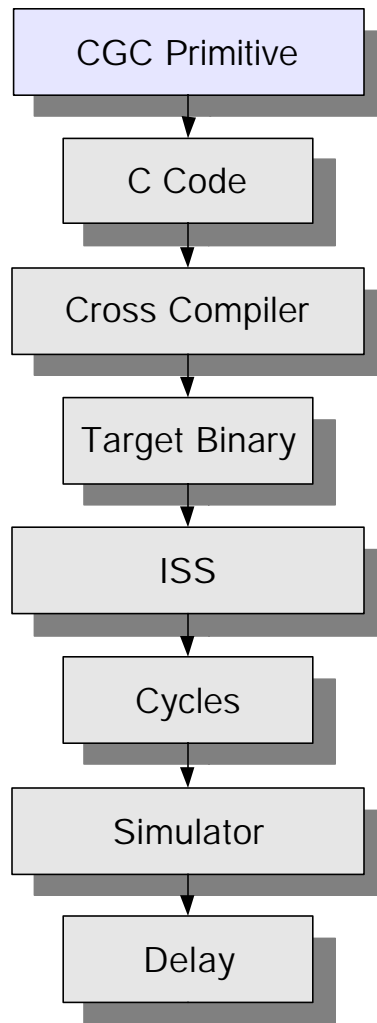


General Work Flow



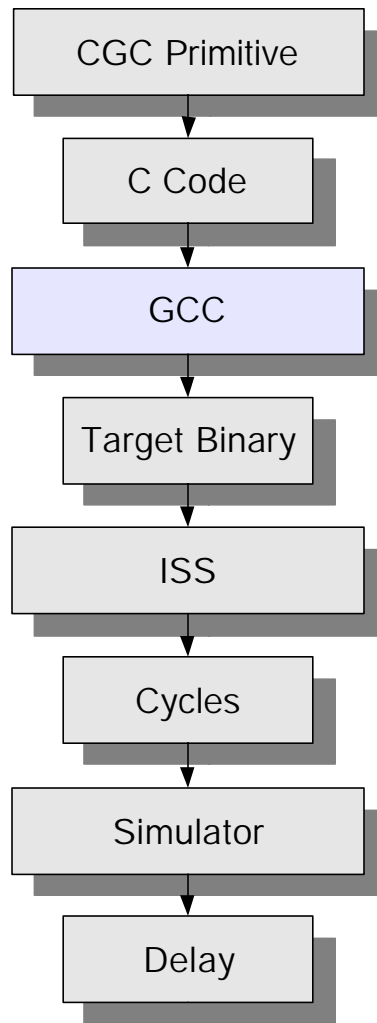
- Cross compiler generates target binary on host machine
- Instruction set simulator executes binary and calculates necessary cycles
- Simulator calculates delay caused by the code execution from cycles

Work Flow in MLDesigner



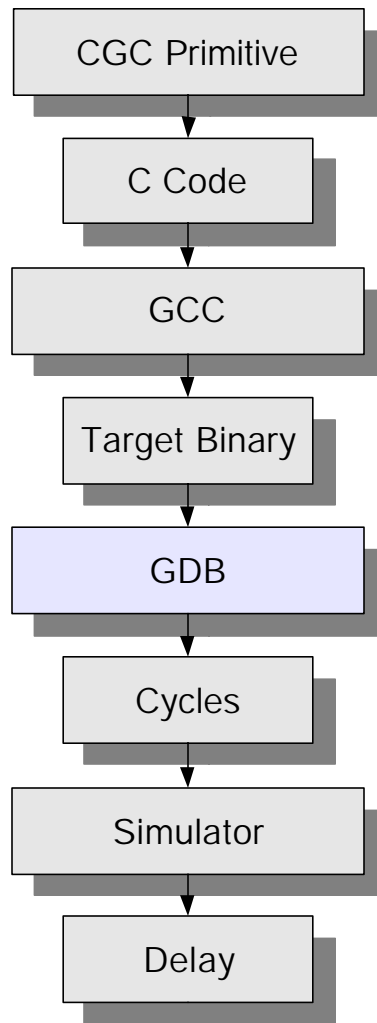
- CGC Domain is a code generation domain of MLDesigner for generation of C Source Code
- CGC Co-Design target introduced

Work Flow in MLDesigner



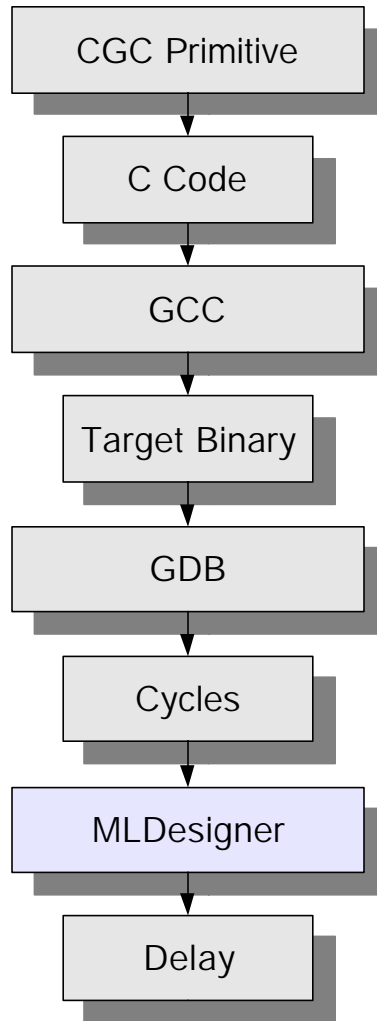
- GNU Compiler Collection
- C front-end of GCC is used for cross compiling
- GCC contains cross-compilers for processors like ARM, Motorola 68k, PowerPC and MIPS

Work Flow in MLDesigner



- GNU Project Debugger
- GDB is a collection of instruction set simulators
- GDB contains fast instruction set simulators like the ARMuLator for the ARM processor

Work Flow in MLDesigner



- Performes DE Simulation with delays calculated from cycles and processor speed

Example

- Problem Definition
- Model Generation
- Simulation and Results

Problem Definition

- Selection of an algorithm that has a runtime depending on the incoming data
-> Fibonacci Algorithm
- 1. Task: How many Fibonacci numbers can be calculated by an ARM processor with 2.5 Mhz in one second
- 2. Task: How do compiler optimizations influence the result

Model Generation

C Function

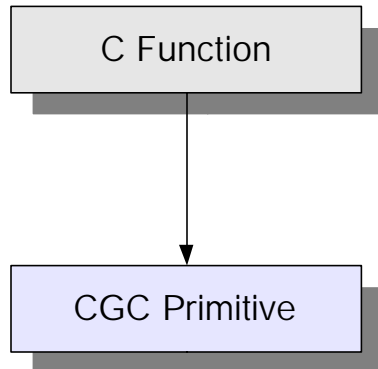
```
void fibonacci(int n)
{
    int data[n];

    int i, j, temp, c;

    i = 1;
    j = 0;

    for (c = 0; c < n; c++)
    {
        data[c] = i;
        temp = i + j;
        j = i;
        i = temp;
    }
}
```

Model Generation



```
defprimitive
{
  name {Fibonacci}
  domain {CGC}
  ...
  codeblock(fibonacci)
  {
    int data[$ref(Input)];

    int i, j, temp, n;

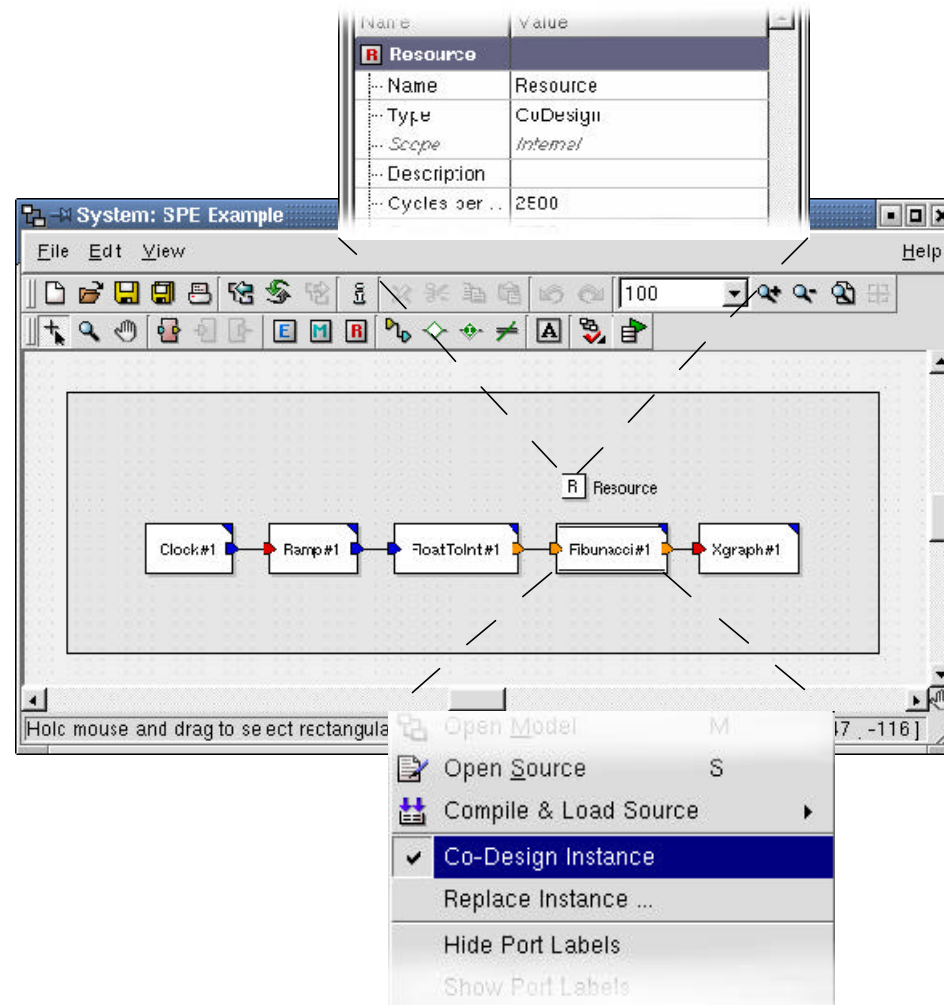
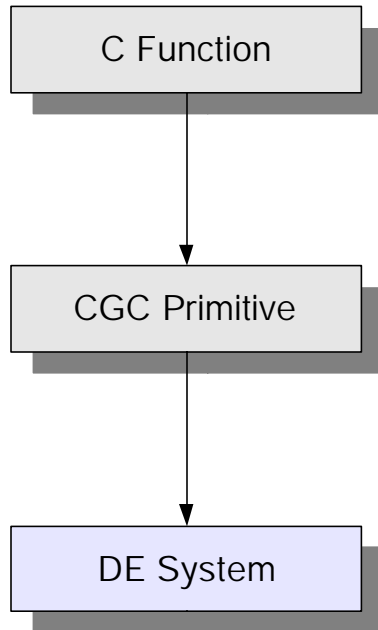
    i = 1;
    j = 0;

    for (n = 0; n < $ref(Input); n++)
    {
      data[n] = i;
      temp = i + j;
      j = i;
      i = temp;
    }

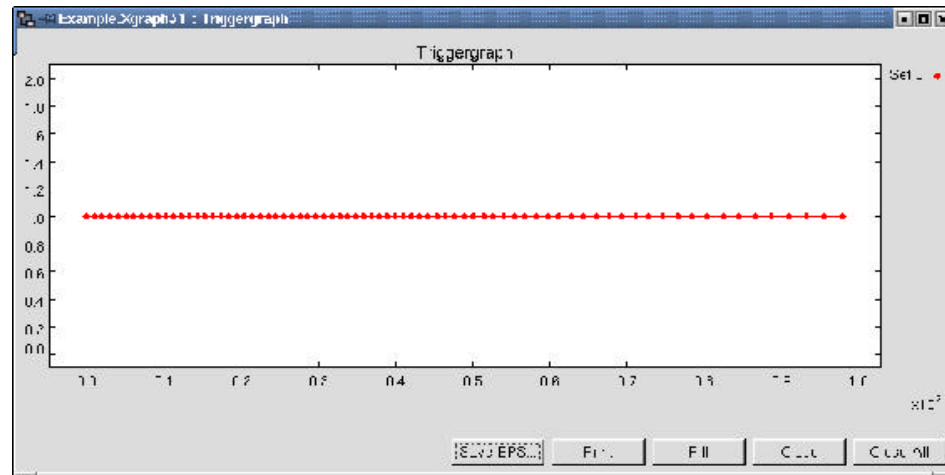
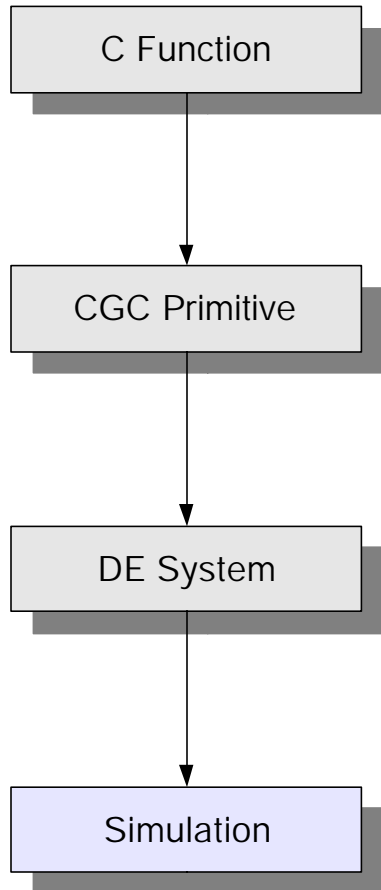
    $ref(Output) = 1;
  }

  go
  {
    addCode(fibonacci);
  }
}
```

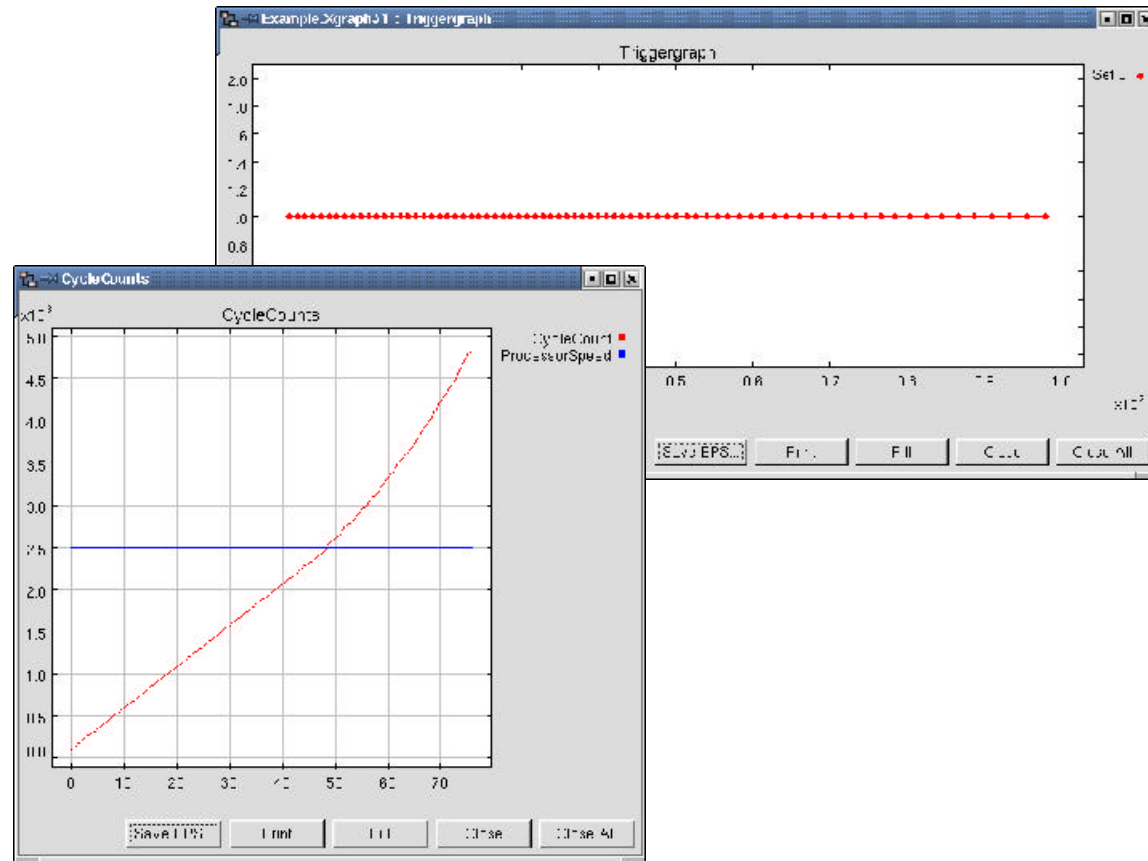
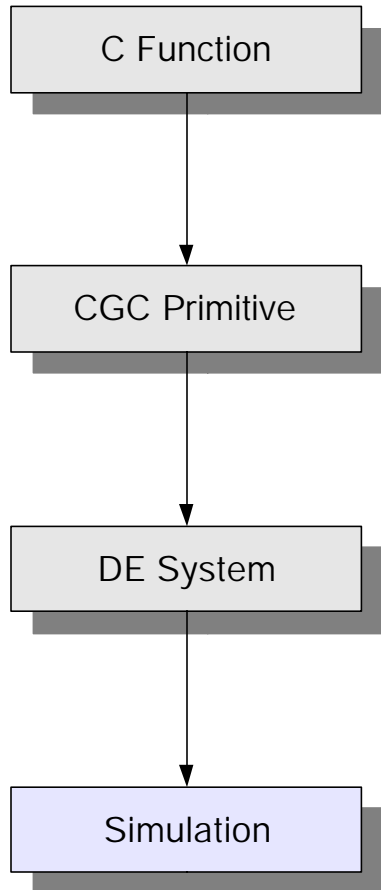
Model Generation



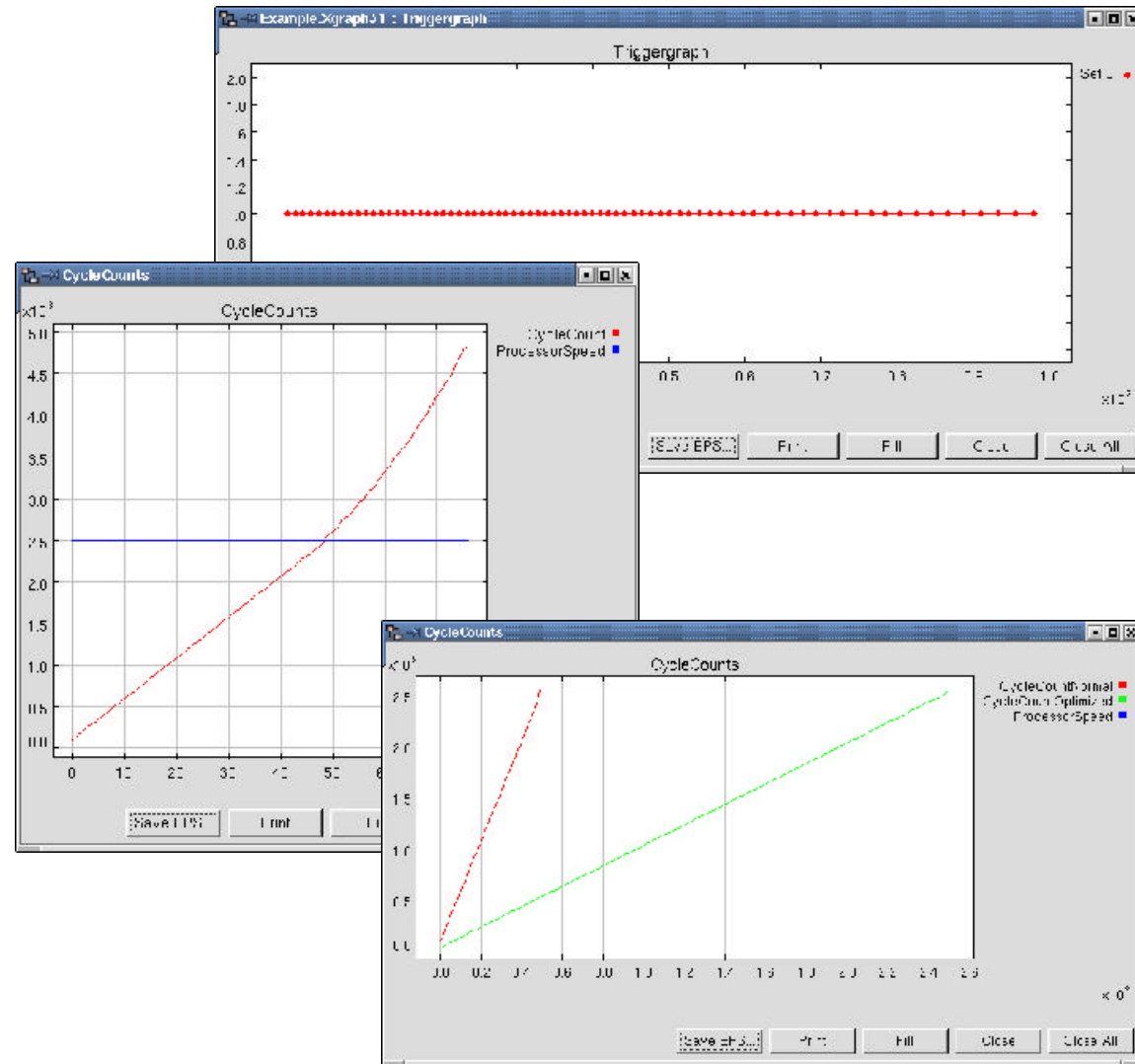
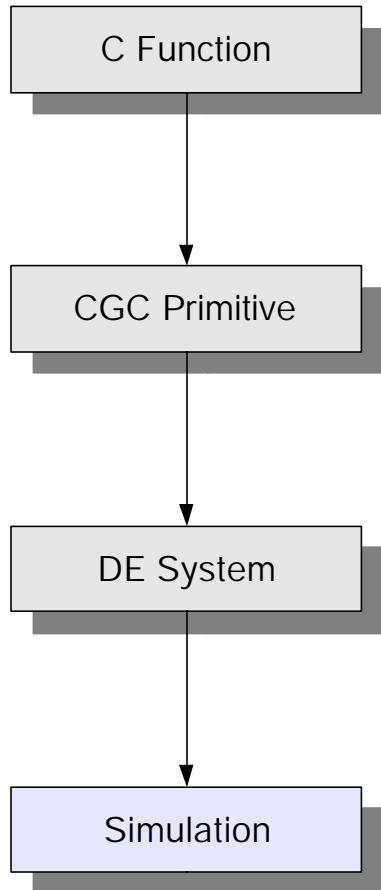
Simulation and Results



Simulation and Results



Simulation and Results



Outlook

- Abstract modeling of hardware and software including the according synthesis algorithms
 - FSM
 - UML
 - SystemC
- Development of modeling capabilities at implementation level

Thank you for your attention!