Behavioral Simulator of Analog-to-Digital Converters

Grzegorz Zareba
Olgierd. A. Palusinski

University of Arizona

Outline

- Introduction and Motivation
- Behavioral Simulator of Analog-to-Digital Converters
- Basic Building Modules of Analog-to-Digital Converters
  - Example of sample-and-hold module
- Simulation of Analog-to-Digital Converters
  - Example of 8-bit multistage A/D converter
  - Example of 8-bit pipelined A/D converter
- Summary
- Future work

Introduction
Simulation Levels

- **Behavioral level simulation**
  - **circuit is described by structural and behavioral blocks**

Simulink, Verilog, VHDL

- **Register level simulation**
  - **circuit is defined by combinational and sequential components**
  - **sequence of register transfers and arithmetic operations is used to describe circuit operation**
- **Switch level simulation**
  - **CMOS transistors are simplified and seen as gate-controlled switches**
- **Gate level simulation**
  - **transistors are grouped into logic gates**
- **Electrical level simulation**
  - **delivers the greatest amount of details about the circuit**
  - **requires solving a system of nonlinear ordinary differential equations**

PSpice, MicroCap

Introduction
Available Simulation Tools

Two options are available for behavioral simulation of A/D converters:

- **Commercial Simulation Tools (Matlab/Simulink, HDL-based simulators)**
- **Dedicated simulators (capable to simulate only one particular A/D converter)**

Disadvantages of Commercial Simulation Tools:

- **expensive in terms of computer time**
- **translation of simulation language is needed**
- **limited by simulation language capability**

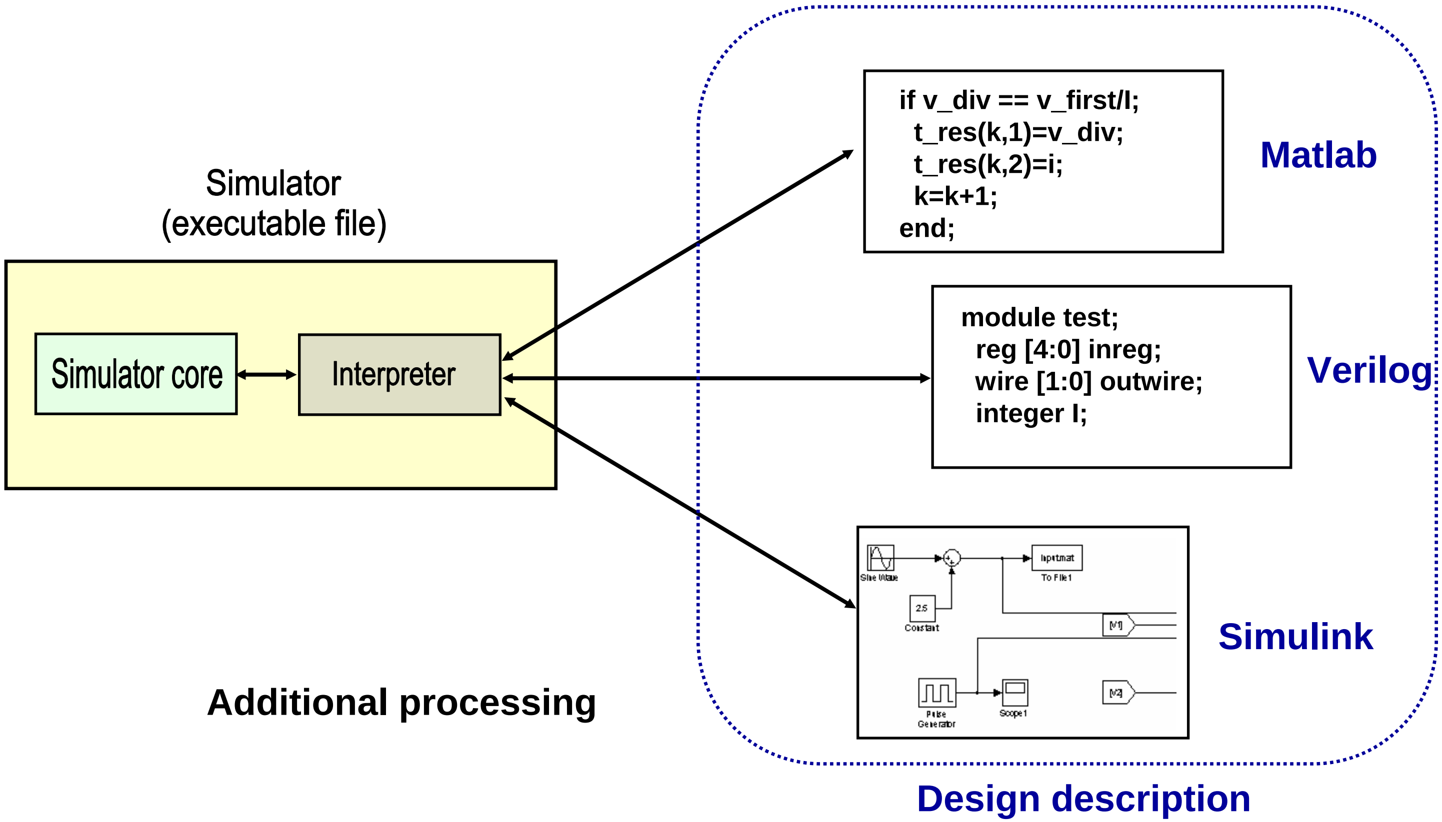
Disadvantages of Dedicated simulators:

- **excessive programming effort needed for implementation of converter model**
- **allows for simulation only one dedicated A/D converter**

Introduction
Simulations with Commercial Simulation Tools

Simulation languages: VHDL, VHDL-A, Verilog, etc

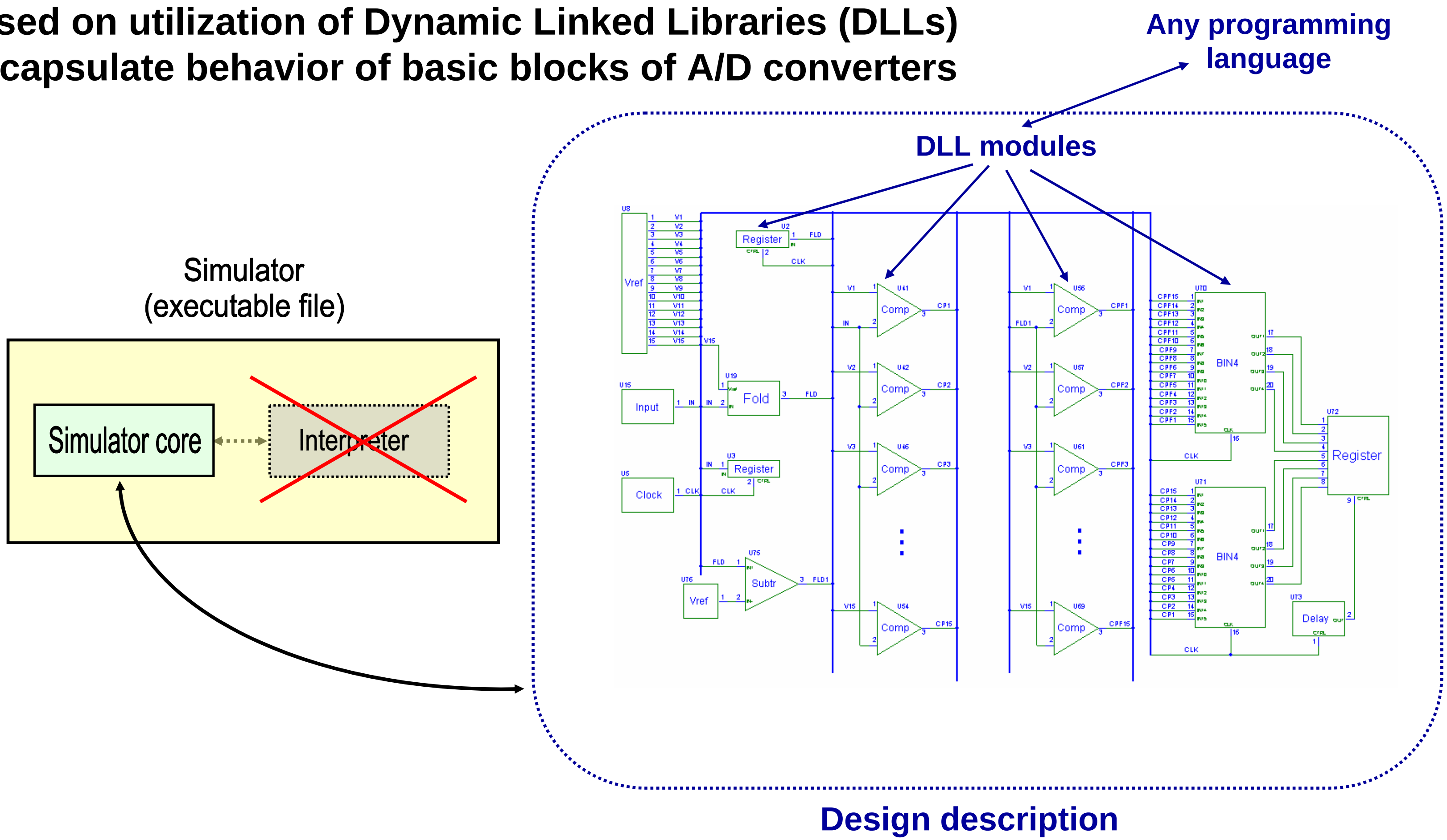
Graphical languages: Simulink, LabView, VEE, etc



Introduction

New Approach in Behavioral Modeling of A/D Converters

A new approach in behavioral modeling of A/D converters is based on utilization of Dynamic Linked Libraries (DLLs) to encapsulate behavior of basic blocks of A/D converters



Introduction

New Approach in Behavioral Modeling of A/D Converters

What is a DLL module?

A library of executable functions or data that can be used by a Windows application

Advantages:

- **Any programming environment can be used to create a DLL module**
- **DLL module can be modified without having to update the simulator**
- **Executable module**

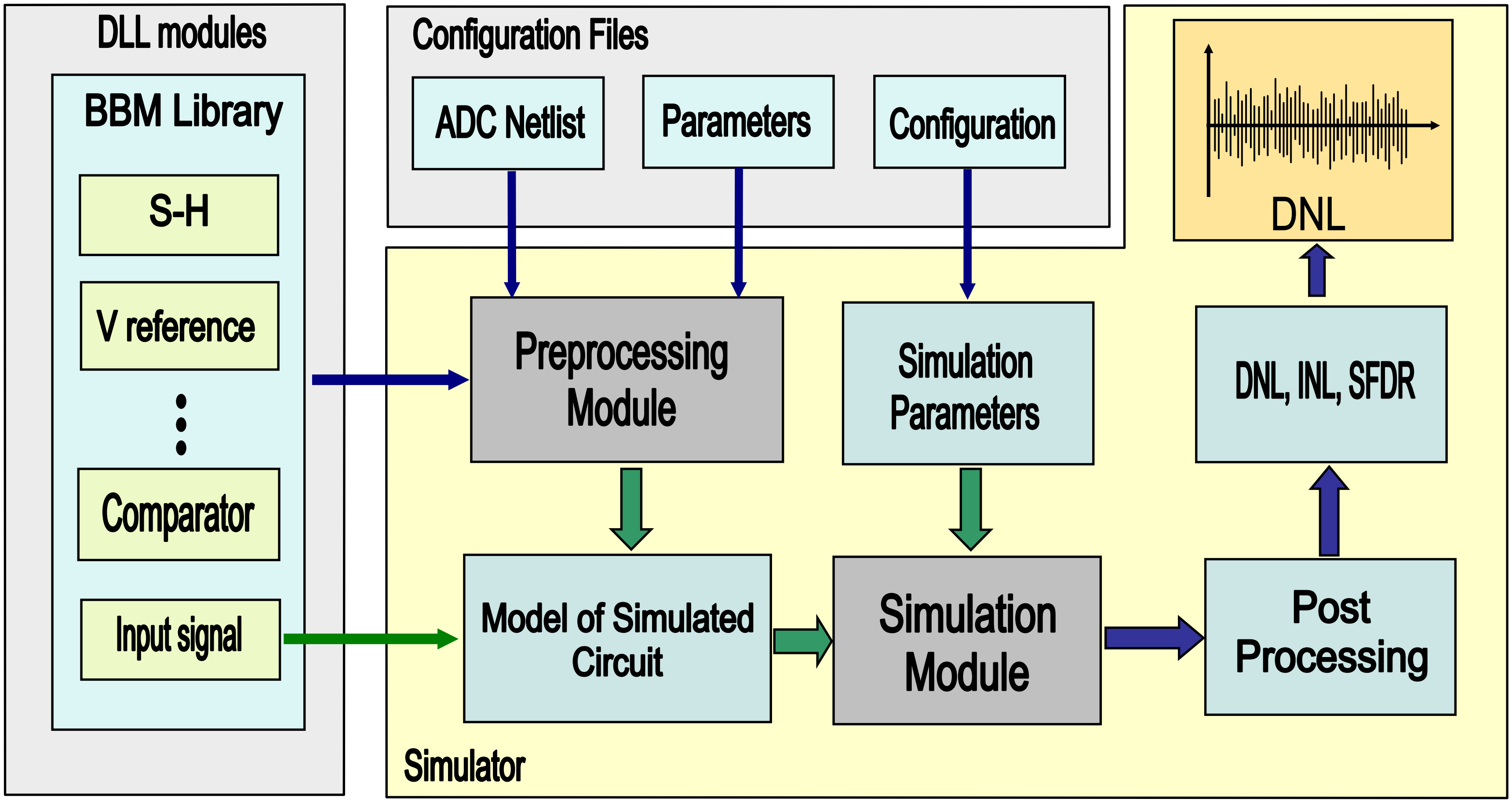
Disadvantage?

- **It seems that creation of a DLL module requires a proficiency in programming**

*.exe		*.dll		
Simulator	+	**BBMs**	=	**Executable model of A/D converter**

Behavioral Simulator of A/D Converters

Structure of the simulator



Behavioral Simulator of A/D Converters

Representation of A/D converters

Configuration Files

- ADC Netlist
- Parameters
- Configuration

Simulation parameters:
- Simulation time
- Simulation mode

Parameters of BBMs:
- input offset voltage
- droop rate
- slew rate
- hysteresis
- delay

Connectivity of BBMs

[BEGIN]
name="Vref N"
id=27
type=Block_VREFN

<out>
out[1]=31:in[2]
out[1]=23:in[1]
out[1]=24:in[1]
out[1]=25:in[1]
out[1]=26:in[1]

[END]

[BEGIN]
name="SubADC"
id=23
type=Block_SUBADC

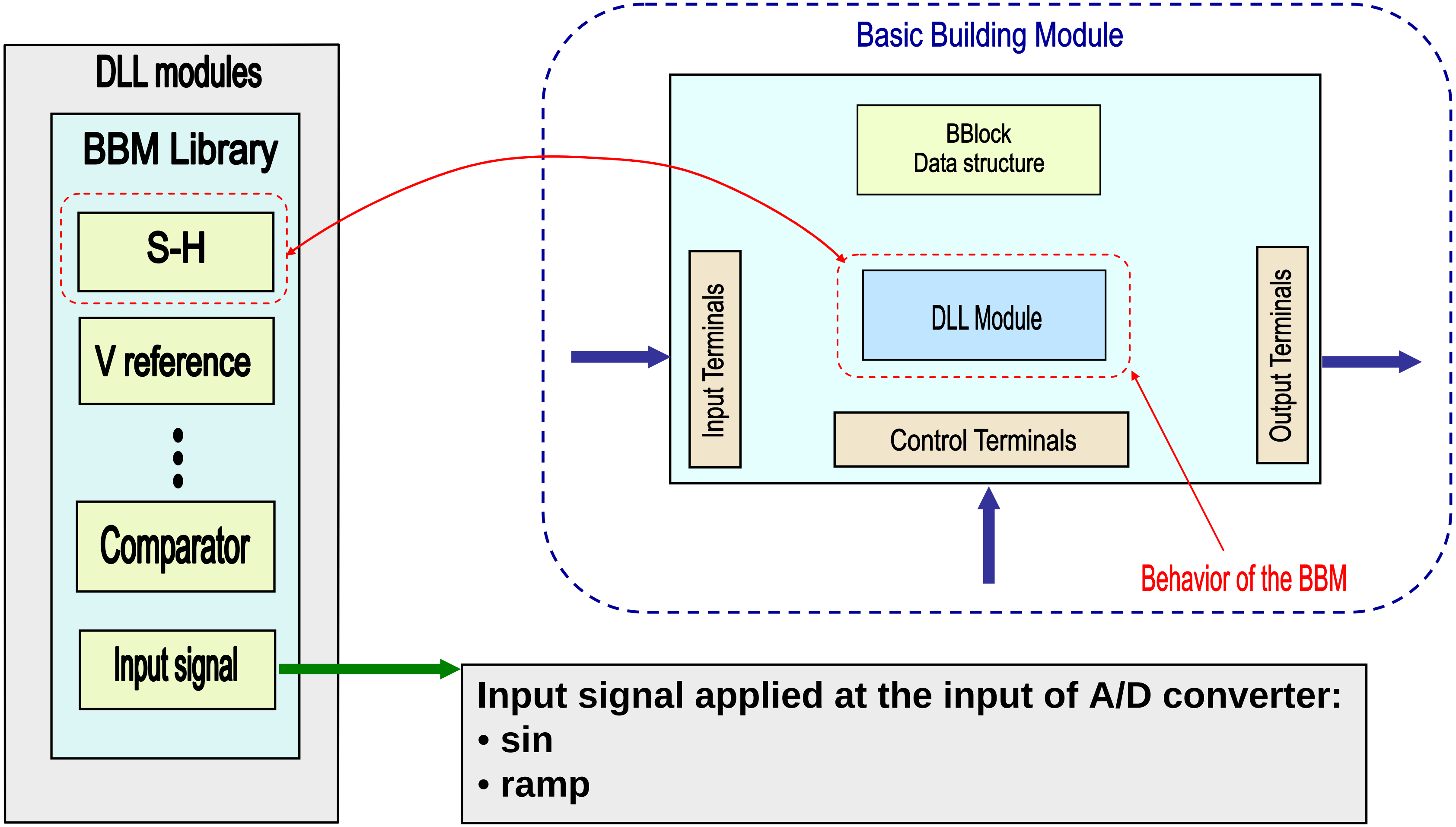
<in>
in[1]=27:out[1]
in[2]=15:out[1]

<out>
out[1]=14:in[1]
out[2]=14:in[2]

<ctrl>
ctrl[1]=3:out[1]
[END]

Behavioral Simulator of A/D Converters

Basic Building Modules of A/D converters



Input signal applied at the input of A/D converter:

- **sin**
- **ramp**

Behavioral Simulator of A/D Converters

BBMs – Example of BBM written in C++

DLL Module ← Executable behavior of the BBM

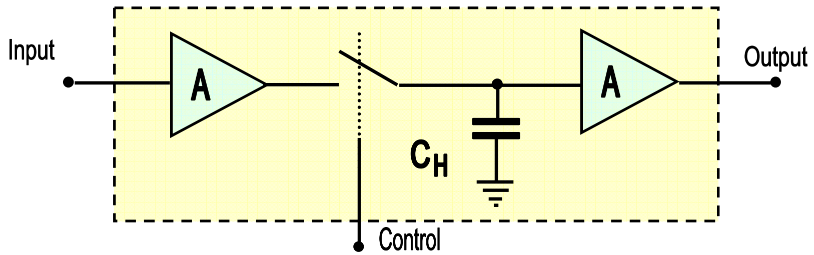
```
if( bCtr )
  { // Block activated by the control line
      if( bSample )
        {
          dOutput = dInput;
          bSample = false;
        }
      else
        {
          dOutput = dInput;
          bSample = true;
        }
  }
else  // Block activated by the output line
  {
     if( bSample )
        dOutput = dInput;
  }
```

More flexible than existing simulation languages

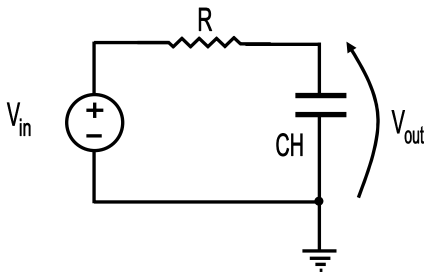
Basic Building Modules

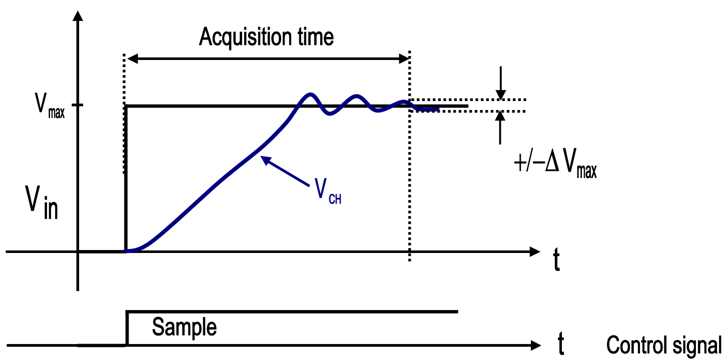
Sample-and-Hold Module

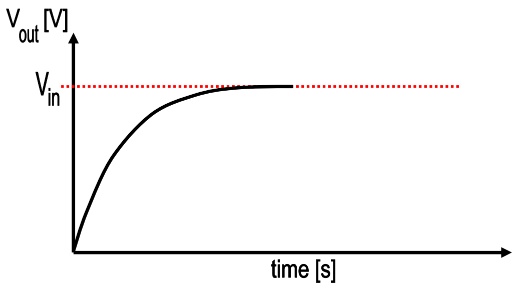
Typical Sample-and-Hold circuit



Approximation with RC circuit







$$\tau = -\frac{t_{acq}}{\ln(0.001)}$$

Basic Building Modules

Sample-and-Hold Module – Behavioral model

Sampling mode: charging capacitor C_H

$$V_{CH}(t) = V_{CH}(t-\Delta t) + \left(V_{in}(t) + V_{off}(t)\right)\left(1 - e^{\frac{\Delta t}{t_{acq}}\cdot\ln(0.001)}\right)$$

Sampling mode: discharging capacitor C_H

$$V_{CH}(t) = \left(V_{CH}(t-\Delta t) - V_{in}(t)\right)\cdot e^{\frac{\Delta t}{t_{acq}}\ln(0.001)} + V_{CH}(t-\Delta t)$$

Holding mode: discharging capacitor C_H

$$V_{CH}(t) = V_{CH}(t-\Delta t) - D_r \cdot \Delta t$$

t_{acq} - acquisition time,

$V_{in}(t)$ - input voltage,

Δt - time step of the simulator,

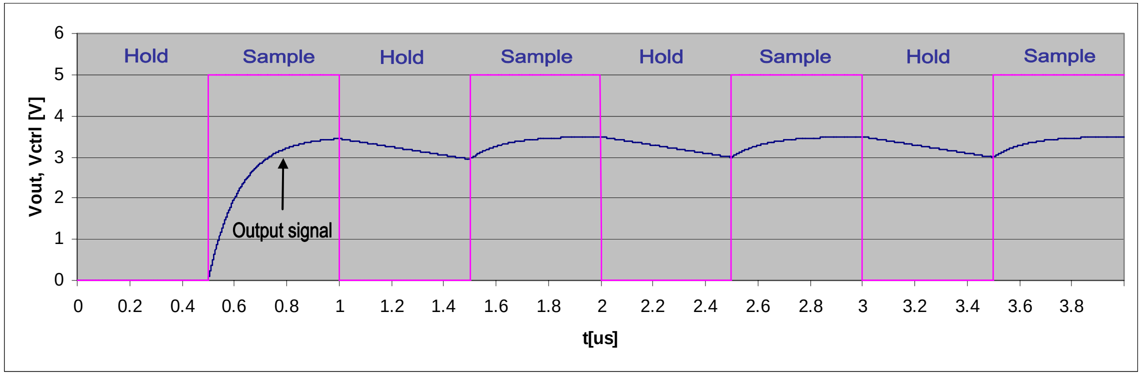
$V_{off}(t)$ - input offset voltage,

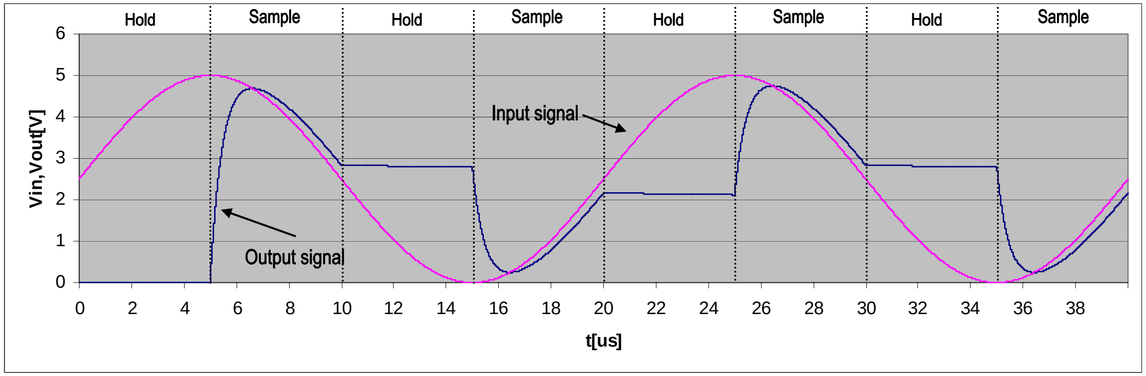
D_r - droop rate

Basic Building Modules

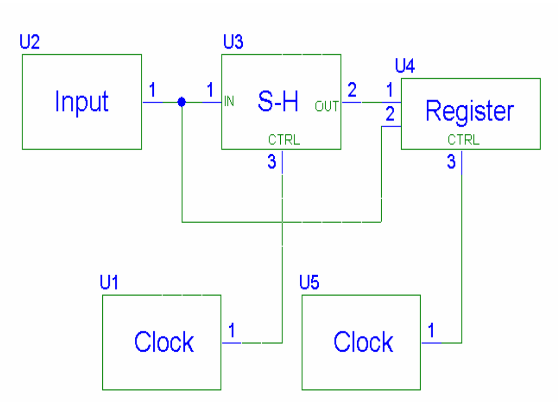
Sample and Hold Module – Simulation results

Simulation results





Test circuit



Basic Building Modules

Other Modules

Analog BBMs:

- Comparator
- Sample-and-Hold
- Analog Switch
- Voltage Reference
- Folding circuit
- Summation
- Subtraction

Digital BBMs:

- Digital Register
- Shift Register

Mixed-Signal BBMs:

- Sub-ADC
- Sub-DAC
- Binary Encoder

Control BBMs:

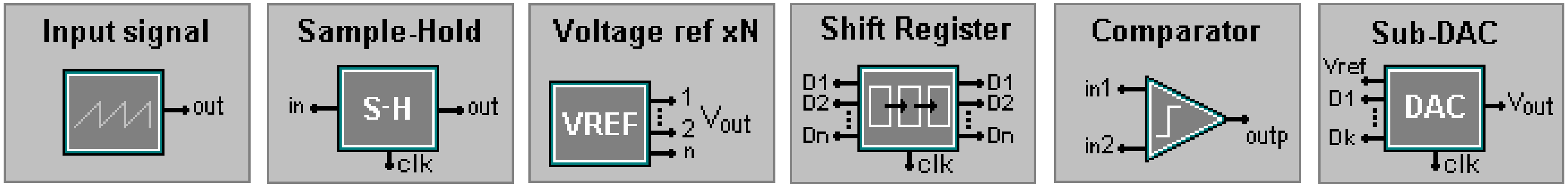
- Input Signal
- Register
- Clock
- Clock Delay
- Noise Generator

Flash, multi-stage, pipelined, and folding A/D converters

Basic Building Modules

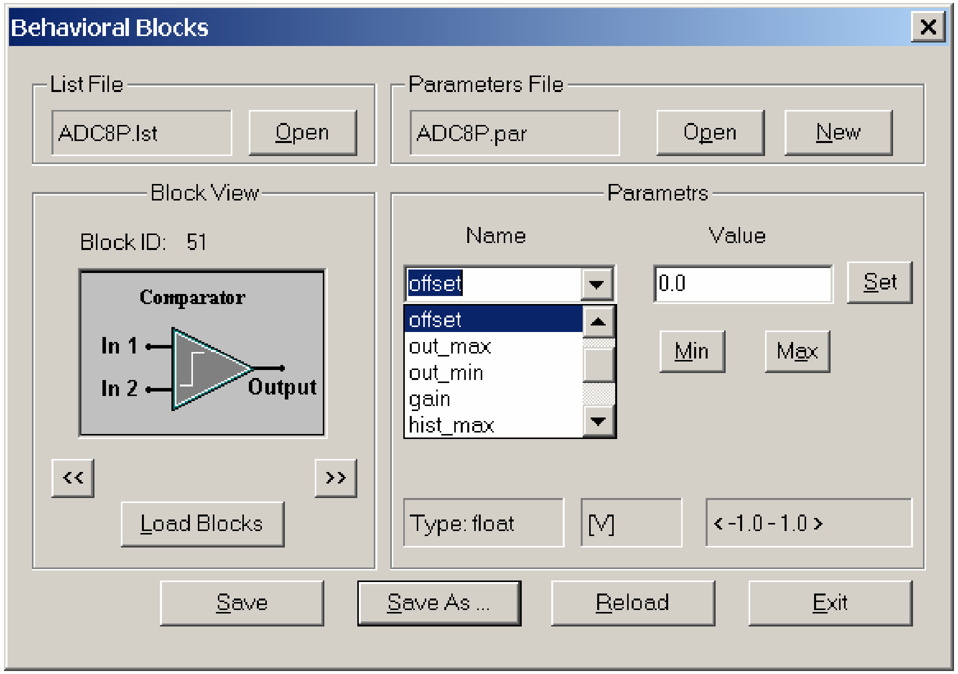
Graphical representation of BBMs

BBM's (Behavior encapsulated in Dynamic Link Library):



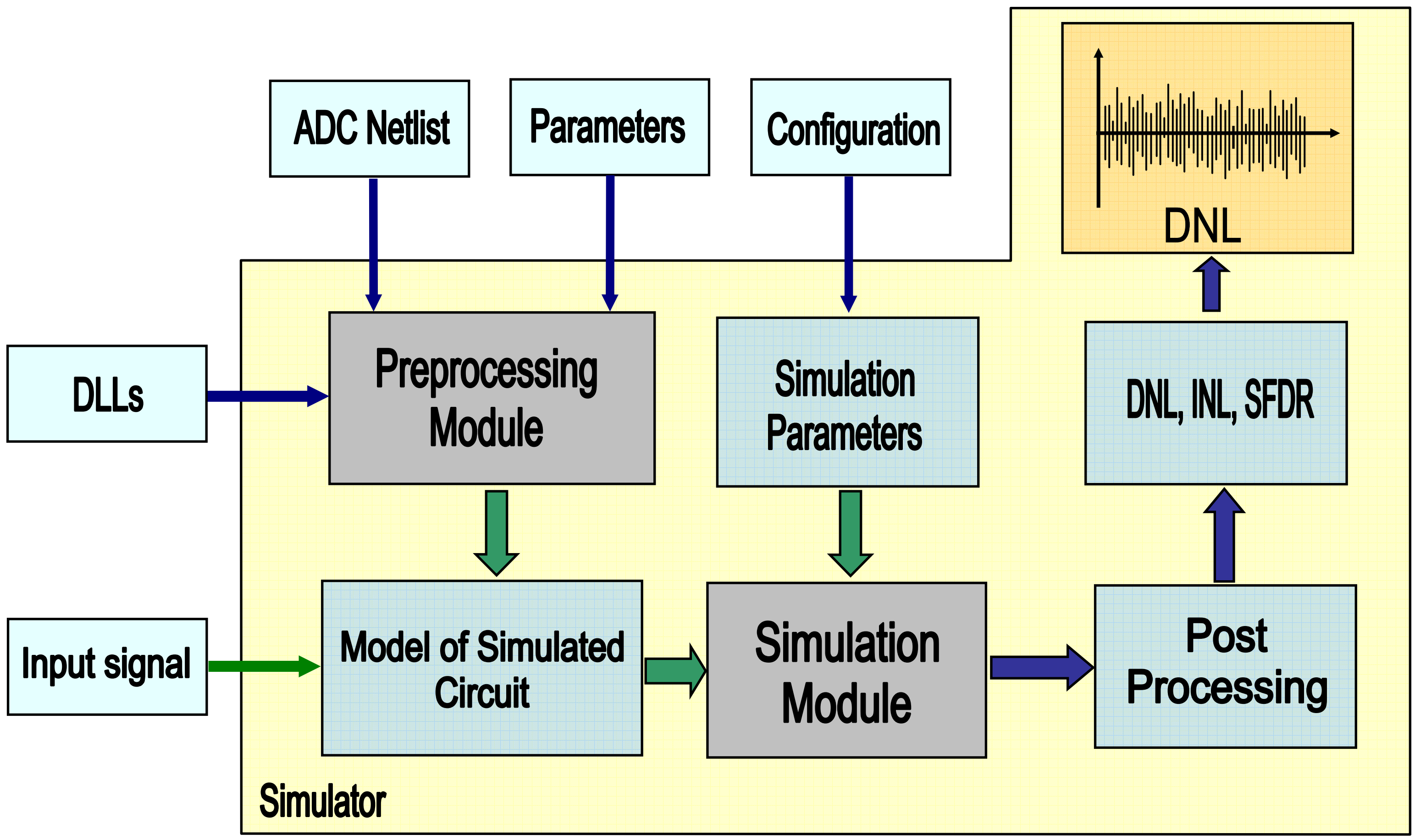
Setting Parameters for Comparator Module:

- **Gain**
- **Input Offset Voltage**
- **Slew rate**
- **Min Output Amplitude**
- **Max Output Amplitude**
- **Min Hysteresis**
- **Max Hysteresis**



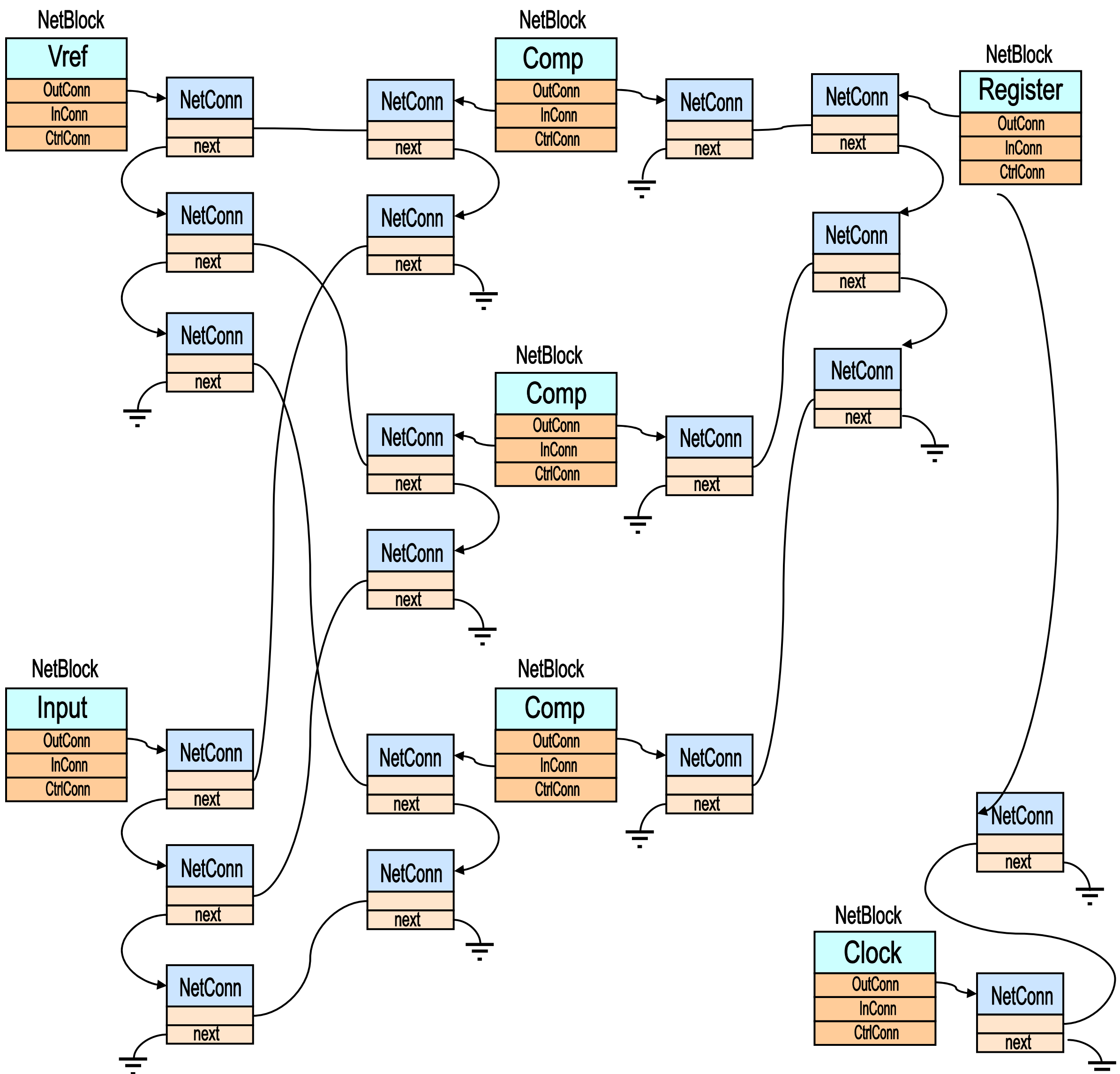
Behavioral Simulator of A/D Converters

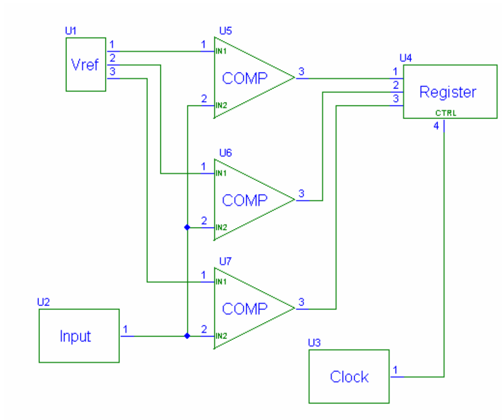
Simulator core



Behavioral Simulator of A/D Converters

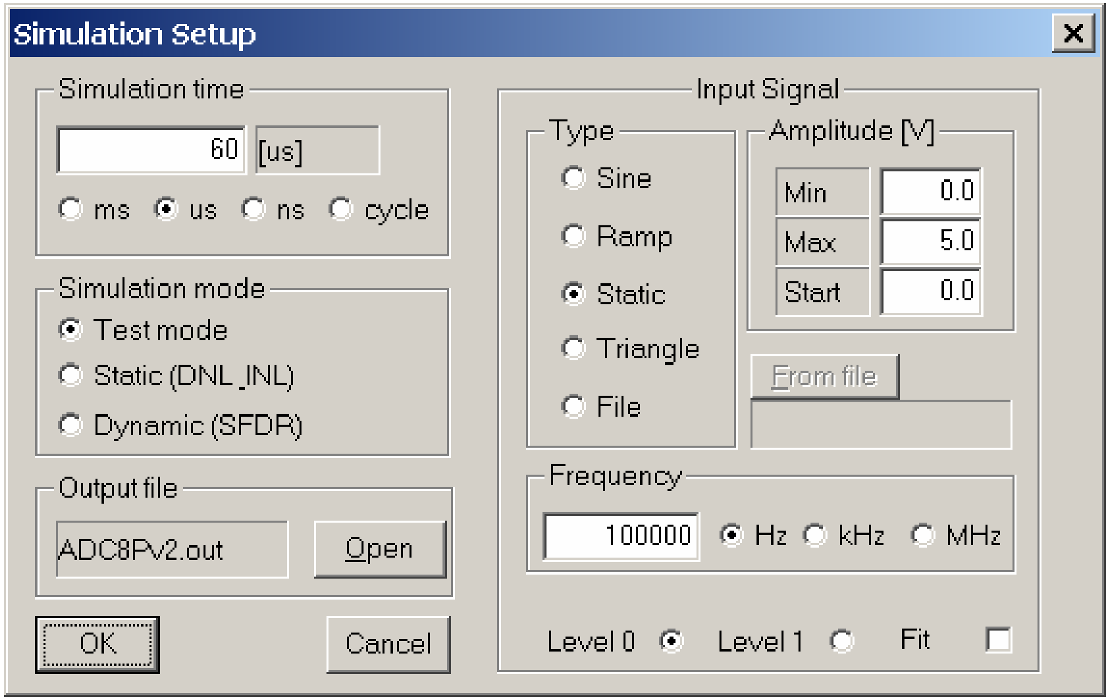
Simulation Module – Multilevel dynamic list





PSpice Schematic

Behavioral Simulator of A/D Converters
Simulation Module – Simulation setup



Simulation Setup:

- **Simulation Time**
- **Simulation Mode**
- **Input Signal type**
- **Clock Frequency**
- **Output File**

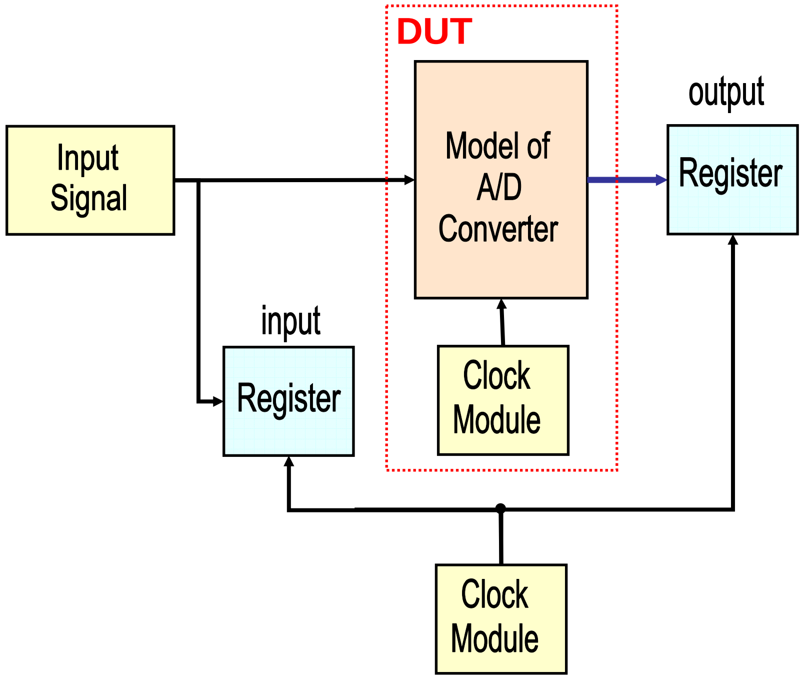
Behavioral Simulator of A/D Converters

Post-Processing Module

Post-processing:

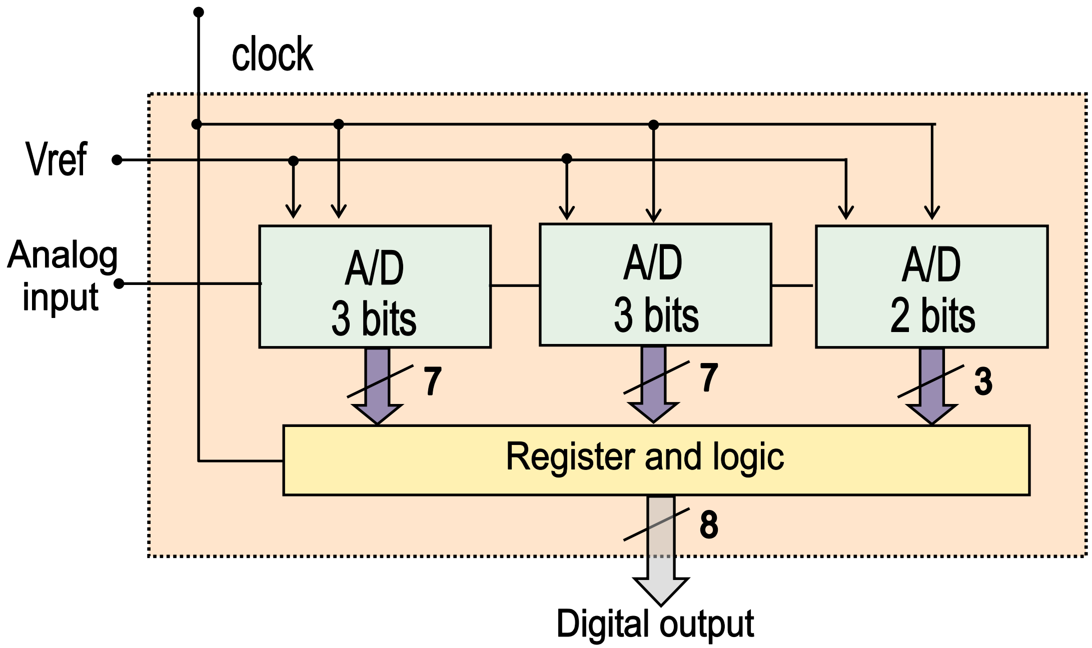
- **Localization of code transition points**
- **Calculation of DNL and INL**
- **Determination of offset and gain error**
- **Calculation of SFDR**

Required circuit configuration:



Simulation of A/D Converters

8-bit Multistage A/D Converter



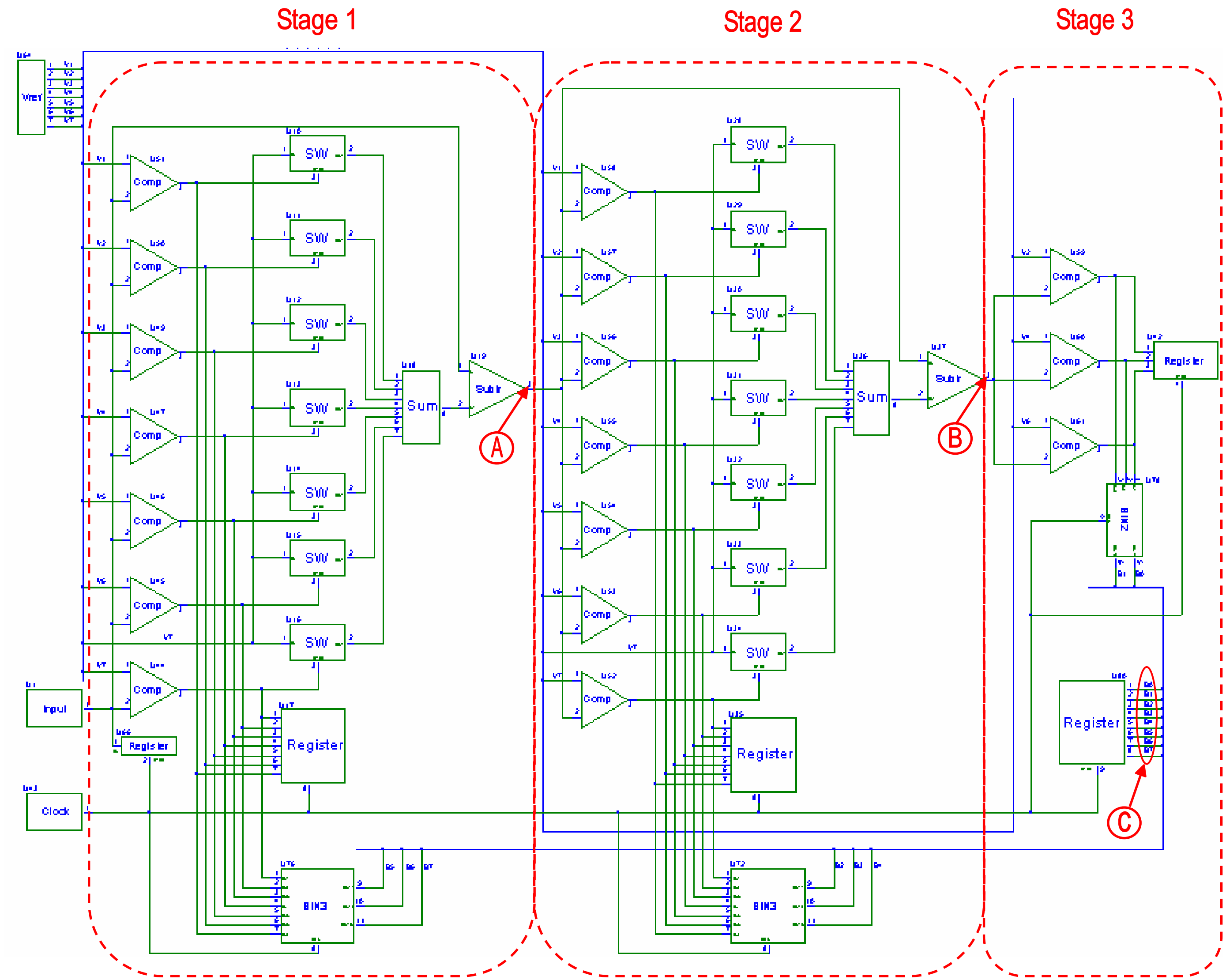
17 Comparators

17 Analog Switches

1 Reference Voltage

Simulation of A/D Converters

8-bit Multistage A/D Converter



Simulation of A/D Converters

8-bit Multistage A/D Converter – Simulation results

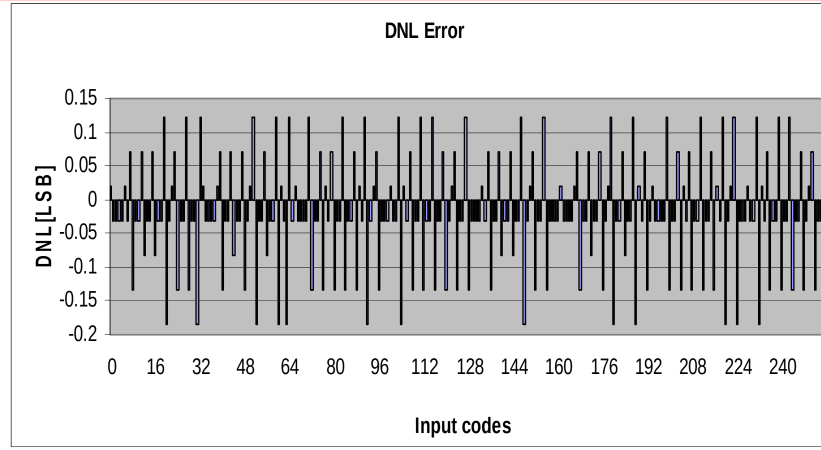
Simulation completed

Simulation report

Simulation mode	TESTMODE
Simulation time [s]	9.784
Last Clock	6000500
Iterations	12000

OK

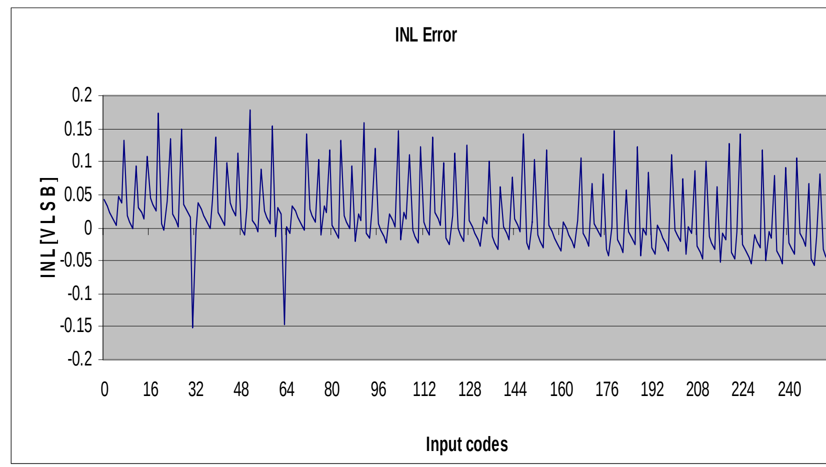
P III, 733 MHz, 256 MB RAM



Simulation results

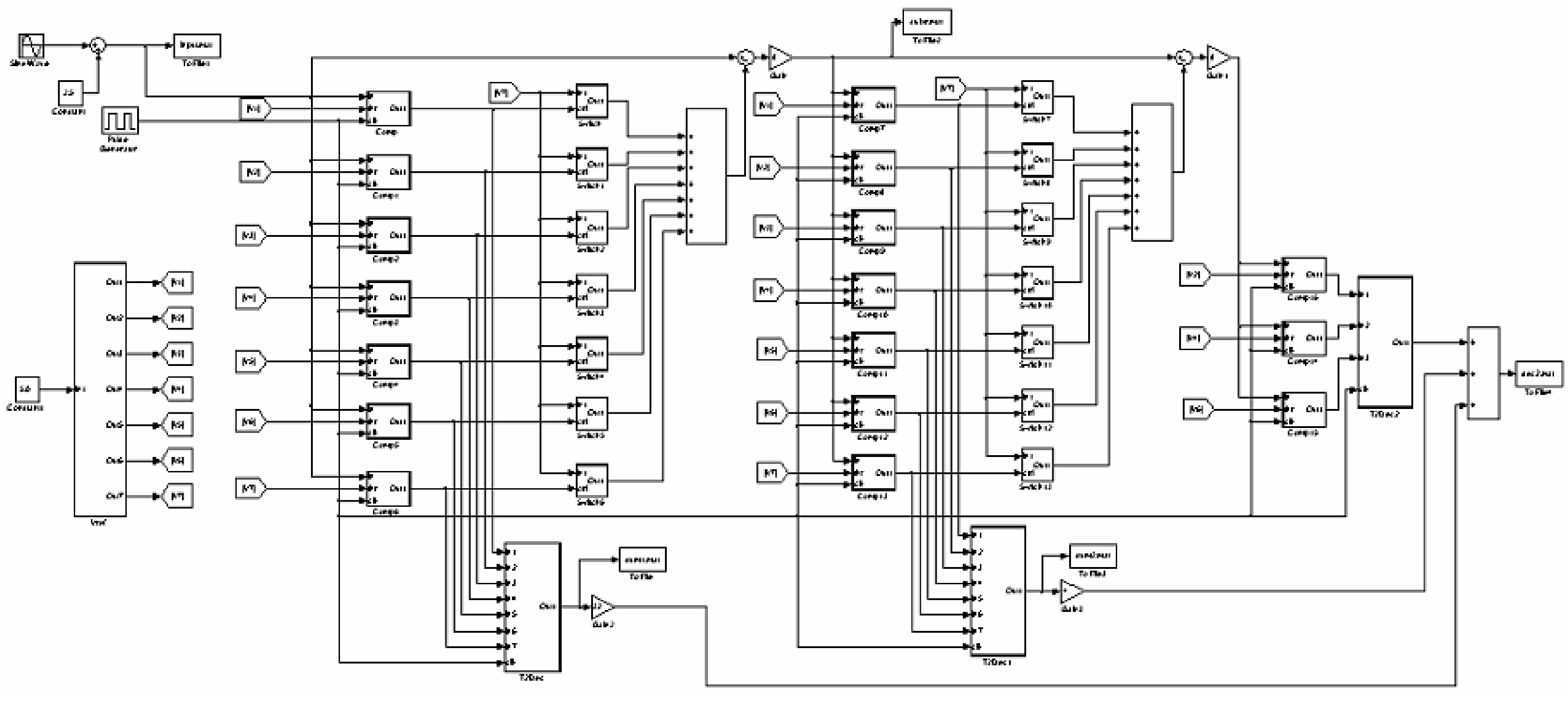
Architecture	PIPELINED		Resolution	8	bit	
Missing Codes	NO	VLSB	[V]	0.019608		
Gain Error	0.022077	DNL MAX	[LSB]	0.122000	0.146770	
Last Code	255	DNL MIN	[LSB]	-0.184000	-0.165985	
Offset Error	0.000000	INL MAX	[LSB]	5.559000	0.151064	
Offset Code	0	INL MIN	[LSB]	-0.020000	-0.178038	
					Gain corr	

OK



Simulation of A/D Converters

8-bit Multistage A/D Converter – Simulink

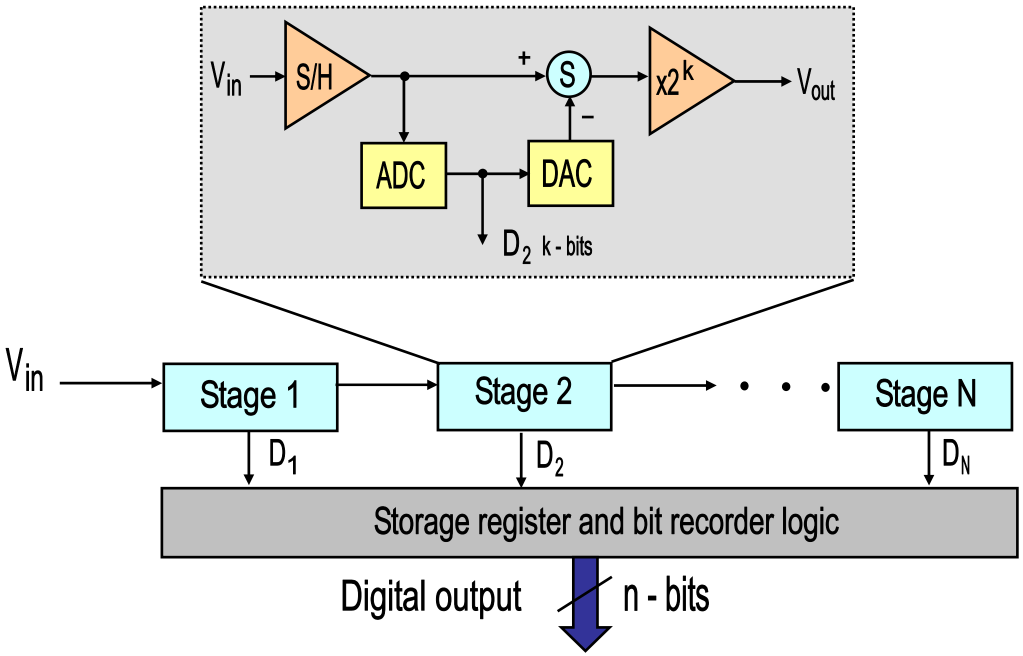


Simulink (ideal model) – 6 min.

Behavioral Simulator – 10 sec.

Simulation of A/D Converters

Pipelined A/D Converters



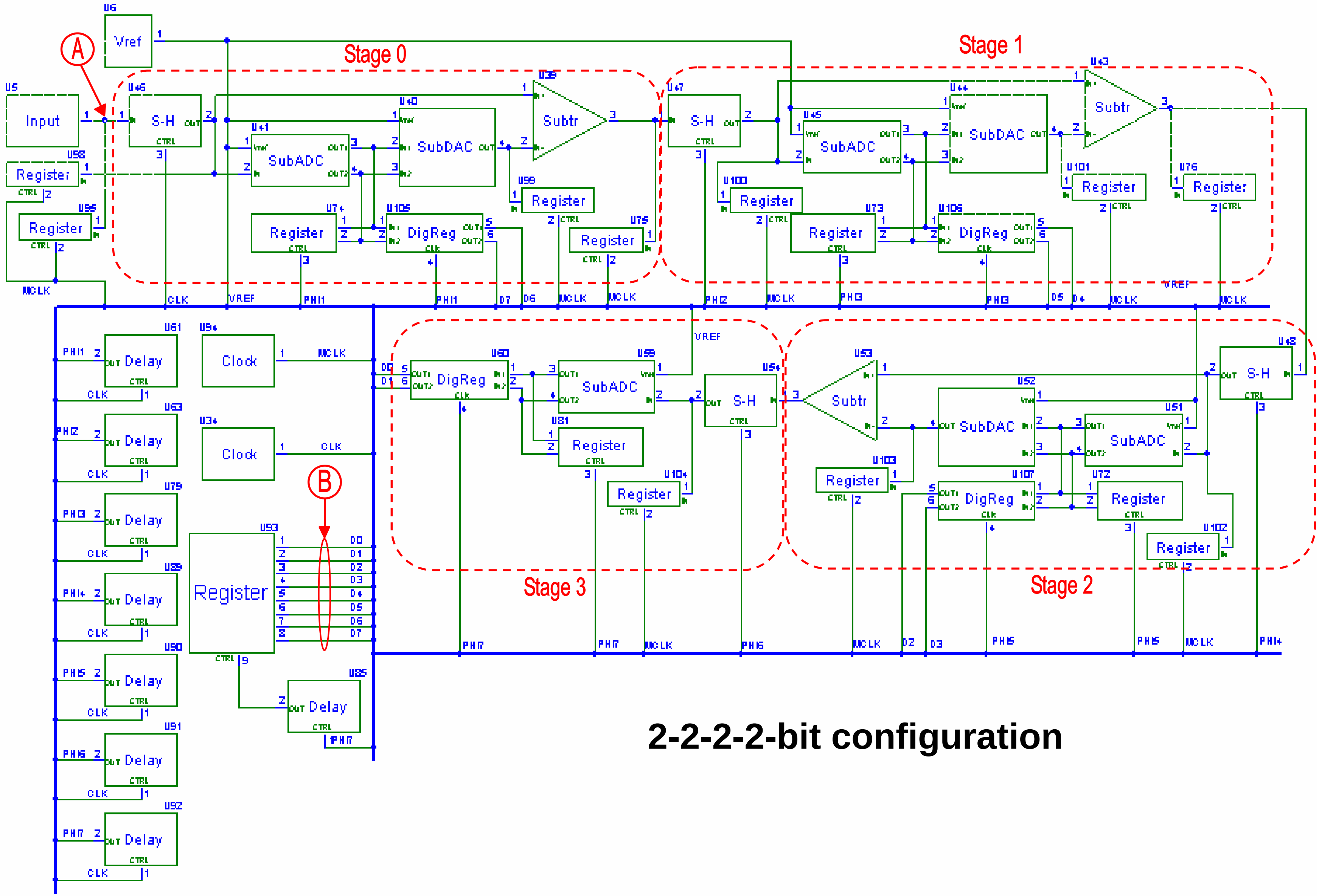
Basic elements:

- **Sample-and-hold**
- **Sub-ADC**
- **Sub-DAC**
- **Summation**
- **Amplifier**
- **Shift register**
- **Digital correction**

$$V_{res} = V_{in} - D_k\left(V_{in}\right) \cdot \frac{V_{FS}}{2^k - 1} \; [V]$$

Simulation of A/D Converters

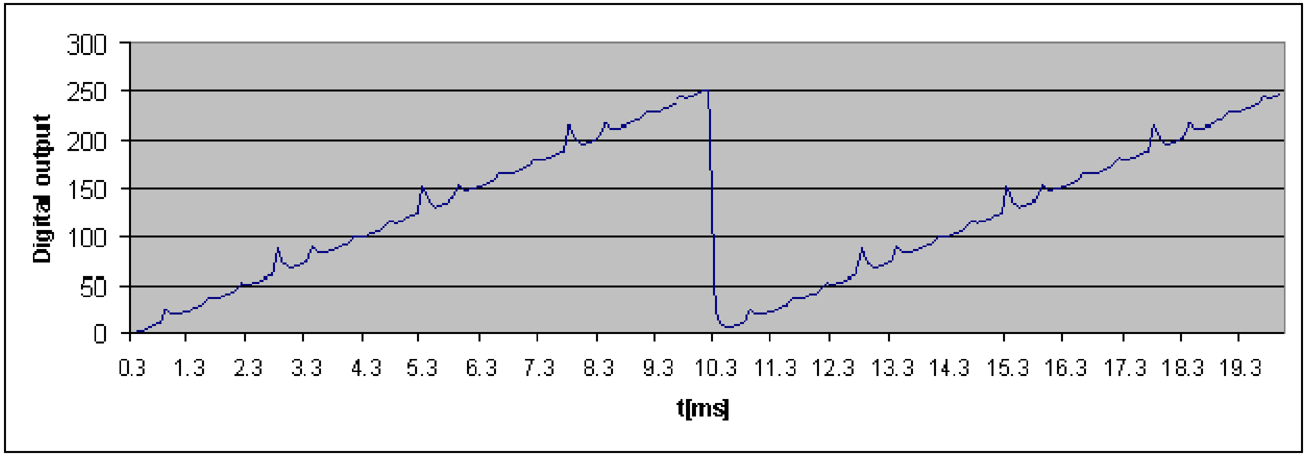
8-bit Pipelined A/D Converter - Schematic



2-2-2-2-bit configuration

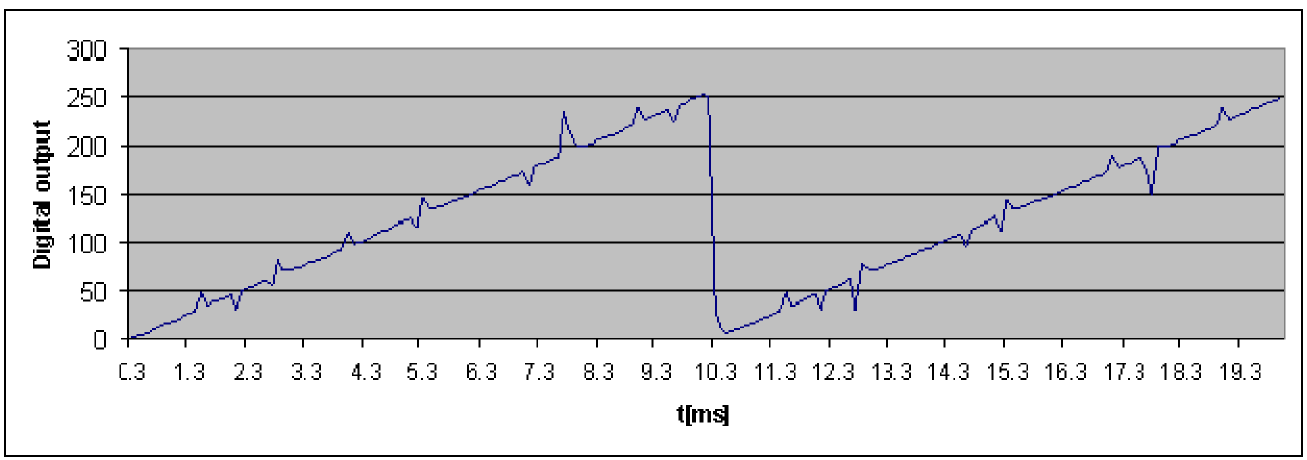
Simulation of A/D Converters

8-bit Pipelined A/D Converter – Simulation results



Imperfections:

- **Synchronization errors**
- **Input offset voltage**



Imperfections:

- **Stability of V_{ref}**
- **Gain error**
- **Input offset voltage**

Summary

- New approach in behavioral simulation of A/D Converters
- New simulation algorithm based on combination of an event driven scheme and data flow technique
- Advanced method for encapsulating BBMs in DLL modules
- Significant reduction of circuit preparation and simulation time
- Open simulator architecture, which allows adding new BBMs without modification of the simulator core
- Simulation package capable to simulate various architectures of A/D converters as well as analog, digital and other mixed-signal circuits

Future work

- Implementation of load effect
- Construction of BBMs designated to support simulation of D/A converters (current source, analog switch, etc.)
- Construction of post-processing module for D/A converters
- Implementation of an interface to PSpice simulator
- Implementation of an interface to Matlab and Simulink
- Development of distributed simulation framework using Local Area Networks (LANs) or Universal Serial Bus (USB)
- Implementation of BBMs for system level design (RAM, EPROM, etc.)

Questions ?