High-Level Synthesis - ASAP and ALAP Scheduling

1st Kristina Bala
Hochschule Hamm-Lippstadt
Electronic Engineering
4th Semester
kristina.bala@stud.hshl.de

***Abstract*—**

***Index Terms*—**

I. First References

[1]()
[2]()

II. Introduction

Synthesis and optimization of circuits at the architectural level.

Techniques for transforming an abstract model of circuit behavior into data path and control unit.

III. Architectural-Level Synthesis and Optimization

Synthesis and optimization of circuits at the architectural level.

Techniques for transforming an abstract model of circuit behavior into data path and control unit.

Data path - interconnection of resources (implementing arithmetic or logic functions) whose execution times and I/O data are determined by the control unit according to a SCHEDULE.

Architectural synthesis:

- Structural view of the circuit, its data path
- A logic-level specification of its control unit

Circuit implementation:

- Area
- Cycle-time (clock period)
- Latency (i.e. nr of cycles to perform all operations)
- Throughput (i.e. the computation rate)

Circuit specifications for architectural synthesis

1. Behavioral-level circuit models
2. Details about the recourses being used
3. Constraints

IV. Scheduling

Scheduling is a very important problem in architectural synthesis.

-the task of determining the start of times, subject to the precedence constraints specified by the sequencing graph

Sequencing graph – prescribes only the dependencies among operations

Scheduling of sequencing graph:

- determines the precise time of each task
- concurrency of resulting implementation
- affects the area of implementation (the max nr of concurrent operations of any given type at any step of the schedule is a lower bound on the number of required hardware resource)

A. Scheduling without resource constraints

applied when:

- dedicated resources are used
  when operators differ in their types
  or when their cost is marginal when compared to that of steering logic, registers. Wiring and control
- resource binding is done prior to scheduling and resource conflicts are solved by serializing the operations that share the same resource.
  The area cost of implementation is defined before and independently from the scheduling step

Used to:

-derive bounds on latency for constrained problems

the minimum latency of a schedule under some resource constraint is at least as large as the latency computed with unlimited resources

V. Allocation

VI. Binding

VII. Unconstrained Scheduling: The ASAP Scheduling Algorithm

- The start time for each operation is the least one allowed by the dependencies
- Optimize the overall latency of the computation without caring about the number of resources required
- Finding the longest path between each operation and the source node
- Starting each operation in a CDFG (Control Data Flow Graph) as soon as its predecessors have completed

VIII. Latency-Constrained Scheduling: The ALAP Scheduling Algorithm

- Schedule operations at the latest opportunity
- Seeking the longest path between each operation and the end or “sink” node.
- Schedule start times can be derived by subtracting the longest path time from the desired overall latency constraint

IX. Application Example

X. Outlook

A. *Drawbacks*

B. *Benefits*

Acknowledgment