EE5332: Mapping DSP Algorithms to Architectures

May – August 2022

Instructor: Dr**.** Nitin Chandrachoodan

Course Pre-requisites : None

Format: 2 lectures

Abstract

Digital Signal Processing typically involves repetitive computations being performed on streams of input data, subject to constraints such as sampling rate or desired throughput. Often such systems need to be implemented under tight constraints on factors such as timing, resources, power or cost. When they are used in embedded systems, it is often worth the effort to design custom architectures that have much better cost tradeoffs than general purpose computing architectures.

This course deals with the analysis of such algorithms, and mapping them to architectures that are either custom designed or have specific extensions that make them better suited to certain kinds of operations. Topics covered include fundamental bounds on performance, mapping to dedicated and custom resource shared architectures, and techniques for automating the process of scheduling. Aspects of architectures such as memory access, shared buses, and memory mapped accelerators will be studied.

Assignments : Assignments will cover various aspects of the design process, starting from implementing and testing specifications, to synthesis and scheduling using high level synthesis tools, and analyzing and improving the resulting architectures.

Pre-requisites:

- Digital design: binary number representations; combinational and sequential circuits; timing analysis; finite state machine design and implementation; power consumption

- Digital signal processing: FIR, IIR filters and basic block diagram architectures; transforms (FFT, DCT etc.)

- Computer programming: basic processor architecture; execution pipeline of a processor; memory layout of a computer system; loading and executing a program on a processor

Topics

- Background: brief review of pre-requisites and recap of important concepts

- Representations and Models for DSP systems; metrics of quality and fundamental bounds

- Hardware implementation: dedicated hardware; transforms; resource sharing
- Scheduling: time and resource bounds; allocation, binding, scheduling; techniques
- Architectures: programmable systems; FSMs and microprograms; instruction extensions

- Memory and communication systems: bus structures; DMA; networks-on-chip

- Specialized architectures: Systolic arrays; CORDIC; GPU