Chapter 2
Introduction to Control Flow

Ivan Ratković
https://orcid.org/0000-0002-0524-7227
Esperanto Technologies, Serbia

Miljan Djordjevic
University of Belgrade, Serbia

ABSTRACT

Modern supercomputer designs fall into distinct categories – data and control flow supercomputer design paradigms. Control flow stands as the go-to design philosophy in all von Neumann machines dominating the market thus far. New types of problems demand a different flow mindset – this is where the data flow machines come in play. This chapter introduces control-flow concept as well as its state-of-the-art examples. Introduction section goes over definitions of terms used in succeeding chapters and gives their brief explanations. A brief explanation of supercomputing as a whole given in the Introduction section is then followed by explanations of the data and control flow design philosophies, with real-world examples of both – multi-core, many-core, vector processors, and GPUs in Control Flow Paradigm section. The third section covers real-world processing unit examples, with a rundown of the best standard and low power commercial and supercomputing market representatives.

INTRODUCTION TO SUPERCOMPUTERS AND THEIR TWO DESIGN PARADIGMS

Supercomputers represent the most powerful computing systems on the planet. First supercomputers were introduced in the 1960s (Cray-1) – the term was broader at that time and the fact that a computer works faster than other general-purpose computers was enough to classify it as a supercomputer (Oyanagi, 2002). Computing has gone so far that chips found in today's pocket devices are more powerful than supercomputers of the past. Initial improvements in supercomputer design involved adding more cores, either general purpose or specialized cores used for highly parallelized task execution. Among the first breakthroughs were the vector machines – processors which operate on whole arrays of data – their single instruction can do as much calculation as a whole program on a normal computer. At the start of

DOI: 10.4018/978-1-7998-7156-9.ch002

the 21st century, general-purpose CPUs become powerful enough to invoke a design philosophy shift in supercomputer architectures – today's supercomputers are built as systems consisting of large amounts of general-purpose market CPUs. With the rapid improvements in GPU power, the latest innovations in supercomputing also take advantage of market class GPUs in order of achieving unparalleled performance gains on specific types of tasks. Unlike general-purpose computers, supercomputers are used for formidable tasks (large scale simulations, weather prediction, cryptanalysis, machine learning, big data problems) which require serious speed and power. Supercomputer's performance is measured in FLOPS (floating-point operations per second) instead of MIPS (million instructions per second) like in general-purpose computers. Today's supercomputers reach up to 100 petaFLOPS and building a supercomputer whose power would be measured in exaFLOPS presents a major challenge of 2020 and beyond.

Out of all applications supercomputers are used for, problems involving manipulating and analyzing massive amounts of data, where normal processing techniques in many cases fail to deliver any results at all, today's so-called big data problems in machine learning and computer vision (deep learning, speech recognition, data mining, genomics, object recognition…), are the primary motivators for dividing supercomputer architecture design philosophies into two classes, those being:

- Control Flow Super Computing Paradigm
- Data Flow Super Computing Paradigm

Computers of the past primarily followed the control flow principle, with programs being executed linearly, in one dimension. With Big Data problems becoming more and more important, with the amount of overall data collected in the past few years surpassing all data collected up to that point, a new approach in computer design was introduced – data flow computing, which essentially differs from the traditional control flow computing in the way program is executed. Unlike control flow, data flow machines use specific hardware capabilities to allow for two-dimensional program execution, in which all task execution is carried out in parallel, while program execution linearity is not guaranteed. With the rise in interest in Machine Learning problems, their high computation cost and only increasing datasets used in trainings, data flow has become increasingly relevant, and for those kinds of applications more important that control flow computing.

CONTROL FLOW PARADIGM

Traditional processing units follow the control flow design philosophy – the emphasis is put on the strict program execution flow which guaranties the instruction execution order predictability. Those processing units aren't limited by power consumption restrains and are often top of the line CISC processors. Today's CPUs are prime examples of control flow architecture in practice – their efficiency is based on caches (instruction and data), data and instruction locality, and branch predictions. Program execution is one dimensional and mostly sequential. Programs executed on these types of machines would be traditional programs with one-dimensional execution.