Preface

Signal processing and especially digital signal processing is one of the most rapidly developing discipline today. Advanced digital signal processing methods are applied in very different fields from such simple tasks as the determination of the frequency of a periodic signal to very complex and computation intensive tasks as automatic recognition of objects from measurements (e.g. face recognition) or identification and control of complex systems (e.g. control of a nuclear plant), etc.

There are a lot of signal processing tasks where rather complex algorithms must be run in real-time. Such examples are tasks involved image processing e.g. reconstruction of three-dimensional images, processing of radar signals to classify flying objects, identification of time varying systems in real-time, etc. These and similar examples need to process vast amounts of data or to perform a large number of calculations quickly (or at least within a reasonable length of time).

Other such applications include testing of complex systems, exploration of natural resources, economic planning, managing large databases, biomedical analysis, real-time speech recognition, robotics, and the solution of large systems of partial differential equations arising from numerical simulations in disciplines as diverse as seismology, aerodynamics, nuclear, and plasma physics. No computer exists today that can deliver the processing speeds required by these applications.

Over the past forty years dramatic increases in computing speed were achieved. Most of these were largely due to the use of inherently faster electronic components. Unfortunately, it is evident that this trend will soon come to an end. The limiting factor is a simple law of physics that gives the speed of light in vacuum. Now, assume that an electronic device can perform 10^{12} operations per second. Then it takes longer for a signal to travel between two such devices one half of a millimeter apart than it takes for either of them to process it. In other words, all the gains in speed obtained by building superfast electronic components are lost while one component is waiting to receive some input from other one. Why then not put the two communicating components even closer together? Again physics tells us that the reduction of distance between electronic devices reaches a point beyond which they begin to interact, thus reducing not only their speed but also their reliability.

It appears that the only way to solve this problem is to use parallelism. The idea is that if several operations are performed simultaneously than the time taken by the computation can be significantly reduced.

Even in the field of computation, the idea of parallelism is not new and has taken many forms. Since the early days of information processing, people realised that it is greatly advantageous to have the various components of a computer do different things at the same time. Typically, while the central processing unit is doing calculations, input can be read from a magnetic tape or from an analog-digital converter and output produced in a line printer, displayed in a monitor. In more advanced machines there are several simple processors each specialising in a given computational task, like operations on floating-point numbers for example. Some of today's most powerful computers contain two or more processing units that share among themselves the jobs submitted for processing.

Only during the last ten-twenty years has parallelism become truly attractive and a viable approach to the attainment of very high computational speeds. The declining cost of computer hardware has made it possible to assemble parallel machines with millions of processors. Inspired by the challenge, computer scientists began to study parallel computers both in theory and practice. In the last decade many parallel computer architectures were developed and based on these results many parallel computers were realised. With the availability of the hardware, the most pressing question in parallel computing today is: How to program parallel computers to solve problems efficiently and in a practical and economically feasible way? In general parallel computing requires algorithms, programming languages and compilers, as well as operating systems in order to actually perform computations on the parallel hardware.

Parallel computation in digital signal processing, however, has some specialities. The most important digital signal processing algorithms have some inherent parallelism, so in addition to the general highly parallel solutions some special algorithm-conform solutions have been developed. A further speciality of parallel digital signal processing is, that because of the inherent parallelism of many signal processing algorithms we can usually avoid looking for special programming tools which may help us to map our problems into parallel architectures. So the field of parallel digital signal processing, implementing digital signal processing algorithms in parallel machines, the field of parallel signal processors are more or less different from those of the general parallel algorithms and parallel architectures. The basic feature of these solutions is that they do not need general supercomputers with hundreds or more general-purpose processors, instead, using a few or much more, but special processors can give better results or such results where the price/performance ratio is much lower than using supercomputers.

Among these special processors we can find the so called digital signal processors (DSPs) which are based on general purpose complex processors with *some special additional functional units* highly required by signal processing algorithms. Such functional units are *high-speed multipliers*, *adders*, the *combination of these units: multiplier and accumulator (MAC) units*, *special data addressing units*, *extra data paths* to ease the intensive data transfer needed by many signal processing algorithms, etc. Parallelism can also be found in these digital signal processors. This chip parallelism means that some functional units of DSPs are operating parallel in time. In addition to chip parallelism DSPs are used to form complex multi-DSP systems where different DSPs are responsible for solving different subtasks of the whole signal processing algorithms. These multi-DSP systems can be considered as some forms of supercomputers, however, the application of these supercomputers has some specialities. They are not general purpose computers, where the algorithms to be implemented must be parallelised fitting the architecture of the computer, instead the multi-DSP parallel computer architectures are constructed according to the needs of the algorithms: *the architectures are fitted to the algorithms*.

An other group of special processors is much more special one. These processors cannot solve general computing tasks easily, but they were developed for solving special signal processing tasks as inner products of vectors, matrix multiplications, additions of many data in parallel, etc., very efficiently. Usually they are rather simple processors or processing elements and a great number of them are integrated in a system e.g. a VLSI circuit.

Recently a new programmable digital device family was developed. Programmable gate arrays or field programmable gate arrays (FPGAs) are comparable in complexity to VLSI circuits, however, the implementation of a complex algorithm using FPGAs usually needs much less time and design experience than VLSI circuit development. These circuits can be used very efficiently for prototype or small series realisation or to implement such new DSP algorithms for which the architectures of the existing DSPs are not the best. To reduce the complexity of the required FPGAs a relatively new - or redeveloped - approach, the parallel/bit-serial computations can be used. In this case the elementary arithmetic operations are calculated in a bit-serial form while many such operations are implemented in parallel form in the same FPGA device. In price/performance comparison FPGAs are similar to or better then DSPs, mainly in such cases where special functions, many arithmetic operations must be realised relatively fast. This solution is also a good or the only possibility if complex signal processing algorithms must be realised as a part of an instrument, where embedded solutions must be used. In such cases the very expensive multi-processor or supercomputer solutions cannot be afforded.

This material is designed as a text for undergraduate courses in digital signal processing and their efficient implementations. In digital signal processing there are a lot of very good books, lecture notes. In many books the questions of parallel implementations are also discussed. So this material will not cover all the questions of our topic. It can be considered as an addition to the good textbooks, where after a short introduction of general DSP algorithms and general parallel architectures the emphasis will be put on special algorithms and special solutions. For a course of *Parallel Digital Signal Processing* this material forms some selected chapters, which with the basic classical works such as [1,2,3,4] will cover all important questions of digital signal processing and its efficient implementations.

This teaching material were developed and edited by the staff members of Technical University of Budapest, Department of Measurement and Information Systems and The Kando Polytechnic, Szekesfehervar in co-operation with Prof. Sergios Theodoridis, University of Athens, Department of Informatics in the framework of a TEMPUS project (S_JEP-08333-94). The material will be used in undergraduate courses of electrical engineering curriculum, and some parts in the graduate education in the following years. We are sure that some critics will come from the students reading and learning it. We hope that using these critical remarks an improved version can be prepared, which will be more complete, more clearly understandable, so more useful for the students.

[1] S. Y. Kung: *VLSI Array Processors* Prentice Hall, Englewood Cliffs, NJ, 1988.

[2] I. Pitas, A. N. Venetsanopoulos: *Nonlinear Digital Filters. Principles and Applications* Kluwer Academic Publishers, Boston, 1990.

[3] Rabiner, Gold: *Theory and Application of Digital Signal Processing* Prentice Hall, 1975.

[4] S. Haykin: *Neural Networks a Comprehensive Foundation* Macmillan College Publishing Company, New York, 1994.