2.7. Parallel - bit-serial architectures

In this chapter, the parallel implementations of the digital signal processing algorithms has been surveyed. After the introduction of the basic structure, like pipe-line, vector and array processors, in this section the basic properties of the parallel, concurrent bit-serial architectures will be summarised. Although it is clear, that generally the parallel architectures with bit-serial arithmetic are not represent so large computation power, as the fully parallel array processors, but their efficiency, low complexity, excellent price/performance makes necessary the analyzes of them, as a competitive, feasible alternative. This is especially true, if one takes into account the availability of the modern user programmable FPGA devices.

The bit-serial architectures are very good candidates of the direct VLSI hardware implementations. The low complexity, regularity and the small wiring i.e. communication cost of the basic arithmetic functional units frequently allows the implementation of the whole algorithm in hardware. This direct mapping of the signal flow graph to the hardware is typically makes possible a one-to-one correspondence between elementary operations and hardware functional units. This non time-multiplexed nature of the functional units also permits the dedicated, coefficient dependant implementation, what gives significant reduction in the complexity of the final hardware. The scheduling of the operations are generally provided by simple controllers, and this additional overhead can be easily tolerated. The first high level silicon compilers was developed for the bit-serial digital signal processing architectures, and their success was demonstrated by the implementation of some typical DSP algorithms.

Linear digital signal processing algorithms are dominated by the two basic arithmetic operation, addition, multiplication (completed by the bit size and the z^{-1} word size delay operators). The specialized digital signal processors using the built in multiplier-accumulator MAC architecture, are optimized for these computations. Depending on the precision, the internal arithmetic is realized by fixed, 16, 24 or 32 bit integer or floating point functional units. These dedicated units are characterized by the relatively high speed, typically single instruction cycle execution, and one or two level pipe-line. As the technology development is going ahead, higher and higher clock rate can be achieved, and this continuous increase in performance will allow the realization of more complex algorithms. But the inherently sequential execution of these software implemented algorithms on the single arithmetic units will always limit the available computational power. For example, the necessary operations of one tap of an FIR filter is executed in just one instruction cycle on a modern DSP processor. But as the order of the filter is increased, the larger order filters sampling rate is dramatically reduced. Although these processors employs a bit parallel arithmetic, the word serial implementation style of the algorithm limits the performance.

The bit-serial implementation of the digital signal processing algorithms are belongs to the low level massively parallel computer architectures. The bit serial operational units, the elementary arithmetic processors are ideal components for the direct hardware implementations. They have very low complexity, the communication cost is almost negligible. Analysing the bit serial arithmetic one can find, that it differs from the usual bit parallel arithmetic in the following: at a given time only one bit of the operands are processed. But because of the parallel, concurrent hardware architecture, these bits are processed in all of the arithmetic units. For comparison one can recall the earlier FIR filter example, in which all the taps of the filter is processed concurrently, and the result of the given sample iteration will be generated at the end of the operation cycle, almost independently of the order of the filter. In this case the execution time is long for small order filters, but for large order filters it is lower then necessary for the DSP processors. The advantages of the bit serial arithmetic is this massively parallel operation method.

There exist two different computation method, according to the processing order of the bits in the data word. These are called as least significant bit first (LSB first) or most significant bit first (MSB first). The LSB first scheme is the most familiar and obvious. The important arithmetic operations are easier to realize with the LSB first, because the direction of the carry propagation is coincident with the direction of the computation So, the generated carry bits are simply stored for one bit delay or clock cycle and used in the immediate next cycle. The only drawback of the LSB first arithmetic operators is, that they are exhibits extra delay on the multiplication operation. This delay can be tolerated in some algorithm implementation, but especially in case of recursive algorithms, it significantly increases the computation time of the critical loop. To avoid this bottleneck, the MSB first or on-line arithmetic has been developed. The MSB first arithmetic shows an general two clock cycle delay independently from the type of operation. Using MSB first arithmetic, it is necessary to use redundant number representation system, and number conversion at the front-end and back-end functional units. Because of the redundant number system, the complexity of the basic arithmetic functional units in MSB first scheme are much larger, typically a two fold factor is observable. This hardware cost together with the increased communication complexity generally makes prohibitive the use of MSB arithmetic.

The properties of the LSB first arithmetic is best exploited in case of the implementation of the inner product operation. If in this vector-vector multiplication, one operand is constant, i. e. it is a data-coefficient type operation, then the dedicated coefficient dependant functional unit syntheses is applicable. The realization of this inner product processors on FPGA devices will be introduced in Chapter 4, together other with examples, like FIR and IIR filters and the discrete cosine transform.