The block size and the basic functionality of the algorithm have been chosen to realization on the prevailing Pentium computers. Therefore the algorithm scheme is constructed to utilize fully the both processor pipelines. An optimized program in assembler language should work only a little faster then a compiled program in a high-level language (we've chosen the C language to compare with, because it's the most popular among the software engineers).
At all that, in spite of the intendance to a specific computer family, the algorithm should be the most convenient to a realization on low-powered computers (e.g. smart cards), and provide the best performance after porting to a specialized microchip. In particular, that's why the algorithm doesn't include the operation of integer multiplication, though it's quite optimized on Pentium processors.
All the circle shift operations are explicitly applied. At the least, it gives rise to algorithm's performance, especially on low-discharged processors and specialized microchips, which have a fixed bridge among the neighboring parts of pipelines for the circle shift operation.

The algorithm is based on the principle of independence of the sequential operations. Each iteration consists of two parts: the linear transformation on a register and addition of the result of a Boolean function of other two registers, to the last register. Therefore, the transformations can be arranged for parallel processing..

There is a table for algorithm speed, implemented for Pentium. (ECB mode)

Block size in bits	Clock cycles for encrypting one block	
	«C» implementation	«Assembler» implementation
64	180	130
128	340	250