BRINCHHANSEN82
Brinch Hansen P.; Programming a Personal Computer; Prentice-Hall Inc. 1982.

In This book, Brinch Hansen presents a complete system for a personal computer, including a programming language with compiler, an operating system, and an abstract code language. The programming language is called Edison, and the abstract code language is called Alva. Included in the book is a listing (in Edison and Alva) of the code for the compiler assembler, and operating system. The book contains a chapter each to explain about the design, and give the formal definition for the language Edison, the system, and the abstract code Alva. There are in addition various chapters giving, and explaining the code for the compiler, kernel etc.

The language Edison is derived from Concurrent Pascal, using many of its concepts. The enumerated types have been expanded, and reals removed from the language. Edison also has no pointer types. The record has been retained with no attempt to make it safer. Brinch Hansen in fact seems to view the records as a means of reinterpreting the bits of memory. The control flow constructs are based on Dijkstra's Guarded statements rather than on the Pascal model of loops and if-then-else.

The language also has the capability for concurrency using a cobegin statement. The mutual exclusion is implemented through a "WHEN expression DO statement list END" statement. The statement list is executed when the expression becomes true. The system does not use any form of interrupts, so context switches are performed only after a statement has executed a WHEN statement.

In the design of Edison, Brinch Hansen displayed some of the same thoughts which went into the design of Turing. He wanted the language to be simple, but did not want to allow unsafe constructs, such as general pointers, and goto statements. His response for the safeness issue, was in general to exclude various features. He also excluded and simplified various features, since Edison is a "system" language, and hence they are not needed. (i.e. real numbers, variant records)

The way that the concurrency is implemented in the system seems limiting. The WHEN construct requires that the system repeatedly evaluate the condition. Further problems, are tied in with the way which the context switches are carried out.

The system he describes contains an editor, the Edison compiler, and a text formatter in addition to the operating system. The operating system is designed to run on a small system with minimal equipment. It allows problems to be written in Edison, compiled and run. The system does not handle the passing of any parameters to a program, so that the program must handle this itself.

In the design of the system, a great amount of effort was spent on the design of the disk system. The result are simple but efficient and effective disk accesses. One drawback, is that it is not possible to change disks during the execution of a program.

The system as presented is a useful software package for a small personal computer. It could be used effectively by a person with limited goals for the system i.e. text editing, reports generating and simple programs. The problems in the design probably preclude its effective use for most applications beyond this.