A Theory of Communicating Sequential Processes

S. D. BROOKES

Carnegie-Mellon University, Pittsburgh, Pennsylvania

AND

C. A. R. HOARE AND A. W. ROSCOE

Oxford University Programming Research Group, Oxford, England

Abstract. A mathematical model for communicating sequential processes is given, and a number of its interesting and useful properties are stated and proved. The possibilities of nondeterminism are fully taken into account.

Categories and Subject Descriptors: D.3.1 [**Programming Languages**]: Formal Definitions and Theory—*semantics*; *syntax*; D.3.2 [**Programming Languages**]: Language Classifications; D.3.3 [**Programming Languages**]: Language Constructs—*concurrent programming structures*; F.3.2 [**Logics and Meanings of Programs**]: Semantics of Programming Languages—*denotational semantics*

General Terms: Theory

Additional Key Words and Phrases: Communicating sequential processes, synchrony, asynchrony, nondeterminism, parallelism, deadlock, safety, liveness

1. *Introduction*

In the last decade there has been a remarkable growth in general understanding of the design and definition of computer programming languages. This understanding has been based upon a recognition that the text of each program expressed in the language should be given a mathematically defined meaning or denotation, in the same way as any other notational system of logic or mathematics. For a conventional sequential programming language, the simplest mathematical domain suitable for this purpose is the space of partial functions that maps from an abstract machine state before execution of a command to the state of the machine afterward. For a programming language with jumps, the appropriate mathematical domain is slightly more complicated, involving continuations. For a programming language in which subprograms are themselves assignable components of the abstract machine state, the appropriate reflexive domain of continuous functions has been

Part of this research was supported by grants from the Science and Engineering Research Council of Great Britain.

Authors' addresses: S. D. Brookes, Department of Computer Science, Carnegie-Mellon University, Schenley Park, Pittsburgh, PA 15213. C. A. R. Hoare and A. W. Roscoe, Oxford University Programming Research Group, 8-11 Keble Road, Oxford OX1 3QD England.