CONTROL DATA®
CYBER 170 SERIES ALGOL 4.0

CONTROL DATA ALGOL 4.0 is a convenient and easy-to-learn procedural programming language and compiler for translating programs into machine language for execution on CONTROL DATA CYBER 170 Series Computers. ALGOL 4.0 closely conforms to the definitions of the international algorithmic language, ALGOL, as defined in "The Revised Report on the Algorithmic Language, ALGOL-60." The report was published in *The Communication of the ACM,* 1963, volume 6, number 1. The ALGOL 4.0 input-output definition is in accordance with the International Organization for Standardization's *Draft ISO Recommendation No. 1538.*

The ALGOL 4.0 compiler is designed to run on the CDC CYBER 170 system under control of the CONTROL DATA Network Operating System (NOS).

ALGOL 4.0 provides the following: (1) precise instructional statements for program sequence and executions, (2) a language for expressing problem-solving order and procedure, and (3) a translator or compiler for developing machine sensible language from the many-for-one macrostatements the ALGOL language prescribes.

LANGUAGE FEATURES

The ALGOL 4.0 programming language is the complete language described in the ALGOL-60 Revised Report with the following exceptions. (Underlining indicates independent basic ALGOL symbols.)

- Integer labels are not allowed.
- Arithmetic expressions whose type cannot be determined during compilation are considered to be real.
- An object time transfer outside of the range of a switch causes an abnormal exit from the program.
- All formal parameters in a procedure must be specified in the procedure declaration.
- Dynamic own arrays (whose bounds are not constants) are not allowed.

The language includes such features as recursion, general call-by-name, and own variables which are globally allocated but local in scope. It also adds a wide variety of standard input-output procedures.

LANGUAGE ELEMENTS

Programmers' statements and declarations built from the ALGOL symbols specify operations and the variables on which the operations are to be performed.

There are three types of variables: real, integer, and Boolean. There are four kinds of variables: procedure, array, switch, and label. Procedure and array may both have a variable type.

Variables are set to values and change in value by assignment statements. The block structure in ALGOL permits the programmer to overlay the same storage locations with different variables during program execution.

Programmers can specify arithmetic, Boolean, and sequence control operations directly. Input-output and standard operations are specified with procedure calls to standard ALGOL library functions.

Arithmetic operations include addition, subtraction, multiplication, division, integer division, and exponentiation. Boolean operations include and, or, not, equivalence, and implication. Sequence control operations may alter the normal sequence of instruction execution either unconditionally or dependent on the value of an expression or a variable.

Input-output procedures transfer data between the computer memory and peripheral devices under format control. Complete buffering is possible in some of these operations. Functions may be evaluated with the standard ALGOL library functions such as sin (x), sign (y).

COMPILER FEATURES

The ALGOL 4.0 compiler is based in design on the ALGOL compiler developed by Regnecentralen, Copenhagen, Denmark. The design has been modified and extended to obtain the most advantageous features.

These features include:

- Implementation of the complete ALGOL-60 revised language (wherever feasible and not in conflict with other advantages).
- Comprehensive input-output procedures.
- Extensive compile-time and object-time diagnostics.
- Wide variety of compilation options, such as the ability to compile both ALGOL programs and ALGOL procedures.
- Ability to generate and execute the object program in either overlay or non-overlay form.
- Optimization facilities.

Source program input is normally a card deck. The source program may also be specified from a different device by a control card option. Source input can consist of both ALGOL source programs and ALGOL source procedures. More than one source program or source procedure can be compiled with a single call.

The compiler detects all source language infirngements and prints a diagnostic for each. It also incorporates further checking into the object program to detect program errors which can be found only at execution time. Regardless of the occurrence of source language errors, all compilations proceed to the end of the source deck with normal error checking. Object code generation is suppressed if any errors are detected during compilation.

The object program includes code to detect errors not detected during compilation. When an error is detected, an error message is issued and a data map displays current values of declared variables in a form which is easily related to the source program.

201,036 A LITHO IN U.S.A. 3/74