

ELLIOTT 900 SERIES SIMULATOR

COMPILING AND RUNNING ALGOL PROGRAMS.

ALGOL Entry points.

In all cases the ALGOL systems consist of a translation phase that compiles ALGOL source code into intermediate code that is subsequently executed by an interpreter.

There are three optional elements to translation.

1. Check functions (checki, etc.) can be included in the intermediate code (by default they are excluded)
2. The program can be executed against the standard library or alternatively in "library mode" so that at runtime only procedures required by the program are loaded, leaving more space for program and data
3. A table of labels and corresponding addresses in the compiled program can be obtained by translating in "report mode".

Two Pass ALGOL Systems.

The two pass system consists of a translator program, which produces an intermediate "relocatable binary" tape with is then loaded by an interpreter program together with the run-time library to execute the translated program. The interpreter will fail if the program fails to translate correctly or if required library procedures are missing.

TRANSLATOR (TAPE 1)

- 8 Normal translate.
- 9 Continue translation after halt (in any mode).
- 10 Report mode. (No intermediate code is produced).
- 11 Normal translate, include check functions.
- 12 Translate in library mode.
- 13 Translate in library mode plus check functions.
- 14 Translate in normal mode and give reports.

ELLIOTT 900 SERIES SIMULATOR

- 16 Freeze name list and punch out a copy of the translator.
- 17 Enable call-by-name extensions (AJH ALGOL only).
- 18 Disable call-by-name extensions (AJH ALGOL only).

In library mode the library tape (Tape 3) should be read in by entry at 9 after the user's program has been translated. The translator will copy the procedures it references from the library tape to the intermediate code output tape.

INTERPRETER (TAPE 2)

- 8 Normal, load translated program.
- 9 Continue after halt.
- 10 Run program.
- 11 Load relocatable binary code (i.e., precompiled ALGOL procedures or SIR code procedures).
- 12 Reload the standard library (TAPE 3) after entry at 13.
- 13 Load a large program overwriting library.
- 14 Dump the interpreter and any loaded code, suitable for reloading under initial instructions.

Entry at 13 is intended for use with a program translated in library mode.

Load and Go Systems.

The load and go systems combine the translator and interpreter into a single system intended for 16K machines or larger. The translator resides in the first module and the interpreter in the second, with remaining memory available for data.

The load and go systems have the following entry points:

- 8 Normal translation.
- 9 Continue translation after halt.
- 10 Run program.
- 11 Load and go translation with check functions enabled and reports.
- 12 Load and go translation in library mode.
- 13 Load and go translation in library mode, with check functions and reports.
- 14 Add separately compiled procedures in relocatable binary format.
- 15 Re-establish built-in library (903 ALGOL only).

ELLIOTT 900 SERIES SIMULATOR

- 16 Translate in report mode. (No executable code is produced).
- 17 Translate to relocatable binary code format paper tape, (used to generate separately compiled procedures and/or to run programs using the "Large Program" system).
- 18 Re-establish the library (HUNTER & AJH ALGOL only).
- 19 Checksum the store and alter location 8 to check the checksum before next translation begins (session to session preservation).
- 23 Enable call-by-name extensions (AJH ALGOL only).
- 24 Disable call-by-name extensions (AJH ALGOL only).
- 25 Freeze current translator name list (i.e., will not be overwritten by next entry at 8 or 11).

Large Program Systems.

The large program system is a modified interpreter that is able to use additional modules of memory to allow for larger programs and data structures to be accommodated. Programs should be compiled to relocatable binary intermediate code using either the translator from the 2-pass system, or the translator within the load-and-go system. The program can be translated in any mode except library mode (or library mode plus checks).

The large program interpreter has the following entry points:

- 8 Load an ALGOL program maintaining existing library and code procedures.
- 9 Load an ALGOL program maintaining existing library only.
- 10 Load an ALGOL program overwriting existing library and code procedures.
- 11 Load a code procedure maintaining existing code procedures.
- 12 Load a code procedure overwriting existing code procedures.
- 13 Load the library overwriting existing library and code procedures.
- 14 Dump a copy of the interpreter, library and code procedures to paper tape, suitable for reloaded by initial instructions.

Any of the entries at 11, 12 or 13 that may be necessary must be made before an entry is made at 8, 9 or 10.

ELLIOTT 900 SERIES SIMULATOR

If the ALGOL program is loaded successfully at 8, 9 or 10 a prompt character (_) is output on the control teletype. The operator should then type one of the following command letters:

R to run the program
D to dump the interpreter, library, procedures and program
 to paper tape suitable for loading by initial
 instructions.

Errors are reported as standard and then a prompt given. The operator should type either C to continue execution or D to produce a dump.

903 Load and Go System: memory available.

The store is allocated as follows:

ALGOL Interpreter	0 to 4000 (approx.)
Space for program, library	4000 to 7500
Translator and loader	7500 to 16200

The space available for the largest program is about 3,500 words. The translator and loader may be overwritten by data (arrays) allowing up to 12,300 words. The message RELOAD TAPE2 will be displayed if an attempt is made to compile a subsequent program after the translator has been overwritten in this way.

If large arrays are used, locations 8180 to 8191 will be used. The array data will not be preserved if any entry is made to initial instructions.

903 Large Program system: memory available.

The interpreter occupies core locations from 0 to 4400 approximately. The library and code procedures are loaded from 4400 onwards but cannot extend beyond location 8179. The loader is approximately 750 words long and is located at the top of store module 1. The loader dictionary extend downwards from the beginning of the loader, but together with the loader

ELLIOTT 900 SERIES SIMULATOR

can be overwritten at run-time. The ALGOL program can extend from the end of the SIR code procedures to the beginning of the loader dictionary but must not exceed 8192 words in length.

Report Mode.

Report mode produces a table of addresses for every procedure, function and label in the translated program. In addition to showing the size of the translated program, this information is useful for tracking down the source of run-time errors.

For example if you edit HUNTER/DEMO9 to have an initial entry at 11 instead of entry at 8 (thereby obtaining HUNTER\DEMO12) you will obtain the following:

QUICKS		
P EXCHAN	ADR	70
E	ADR	82
P WICHMA	ADR	84
E	ADR	187
P RANDOM	ADR	189
E	ADR	237
P PARTIT	ADR	239
L UP	ADR	265
L DOWN	ADR	288
L CHANGE	ADR	312
E	ADR	348
E	ADR	379
E	ADR	410
E	ADR	410
P QUICKS	ADR	412
P LOCAL	ADR	417
E	ADR	452
E	ADR	452
E	ADR	458
E	ADR	555
E	ADR	556
PROGRAM	585	
SCALARS	17	

This shows that the program titled Quick Sort (i.e., QUICKS) has various procedures called EXCHAN, WICHMA, RANDOM. PARTIT,

ELLIOTT 900 SERIES SIMULATOR

QUICKS and LOCAL indicated by their names being preceded by the letter P. Note that although the program name does not conflict with the procedure name, it could conflict with the name of a library procedure. Labels are preceded by the letter L and "END"s are given by the Es, together with the overall length and the space occupied by all the scalars.