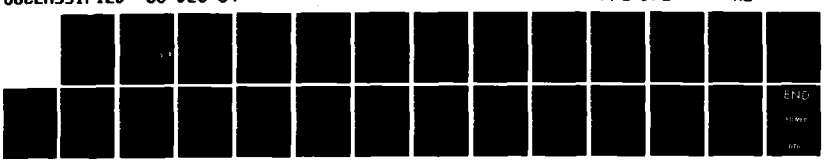
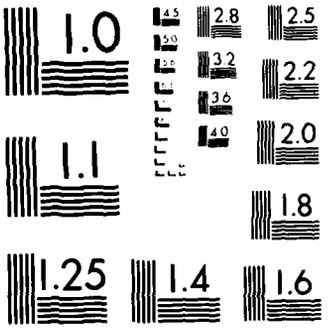
AD-A153 750 ADA (TRADEMARK) COMPILER VALIDATION SUMMARY REPORT: 1/1
ALSYS ALSYCOMP-0001(U) BUREAU D'ORIENTATION DE LA
NORMALISATION EN INFORMATIQUE ROCQUENCOURT (FRANCE)
UNCLASSIFIED 08 DEC 84 F/G 9/2 NL





BUREAU D'ORIENTATION
DE LA NORMALISATION EN INFORMATIQUE

6

Ada*Compiler Validation Summary Report :

AlsyCOMP_0001

December 8th, 1984

prepared By

BNI
Domaine de Voluceau - Rocquencourt
B.P. 105 - 78153 LE CHESNAY CEDEX

for

Ada Joint Program Office
400 Army-Navy Drive
Washington, D.C. 20301

ALSYS
29, Avenue de Versailles
78170 LA CELLE SAINT CLOUD
FRANCE

*Ada is a registered trademark of the U.S. Government (Ada Joint Program Office).

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1 REPORT NUMBER AD A 153 750	2 GOVT ACCESSION NO. / 3 RECIPIENT'S CATALOG NUMBER
4 TITLE (and Subtitle) Ada Compiler Validation Summary Report ALSYS ALSYCOMP-0001	5 TYPE OF REPORT & PERIOD COVERED December 8, 1984 to December 8, 1985 6 PERFORMING ORG. REPORT NUMBER
7 AUTHOR(s) Bureau D'Orientation De La Normalisation En Informatique (BNI)	8 CONTRACT OR GRANT NUMBER(s)
9 PERFORMING ORGANIZATION NAME AND ADDRESS BNI Domaine de Voluceau Rocquencourt B.P. 105 78153 LeChesnay Cedex FRANCE	10 PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11 CONTROLLING OFFICE NAME AND ADDRESS Ada Validation Office Institute for Defense Analyses 1801 N. Beauregard Street Alexandria, VA 22311	12 REPORT DATE December 8, 1984 13 NUMBER OF PAGES 23
14 MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Ada Joint Program Office Room 3D-139 (1211 S. Fern, C-107) The Pentagon Washington, D.C. 20301-3081	15 SECURITY CLASS. (of this report) Unclassified 15a DECLASSIFICATION/DOWNGRADING SCHEDULE

16 DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17 DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Unclassified

18 SUPPLEMENTARY NOTES

Final Version

19 KEY WORDS (Continue on reverse side if necessary and identify by block number)

Ada Compiler Validation Summary Report, VSR, Ada Language, Validation Testing, Ada Compiler Validation Capability, ACVC, BNI, Ada Validation Facility, AVF, Ada Validation Office, AVO, AJPO, Ada Joint Program Office, ALSYS, ALTOS, VAX, Motorola MC 68000 UNIX. ANSI MIL/SDT Ada, ACVC Version 1.4

20 ABSTRACT (Continue on reverse side if necessary and identify by block number)

The ALSYS, AlsyCOMP_0001 Compiler Version 1.0 was tested with Version 1.5 of the ACVC validation tests. Version 1.5 of the test suite contained 2231 tests, of which 1966 were applicable to this implementation. Of the applicable tests, 68 were withdrawn due to errors in the tests. Of the remaining applicable correct tests 1897 tests passed and one anomaly was discovered in the test suite.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

This report has been reviewed and is approved.

Thomas H. Probert

Thomas H. Probert, Ph. D.
Institute for Defense Analyses

Robert F. Mathis

Robert F. Mathis
Director, AJPO-AVF

N. Malagardis

N. Malagardis
Director, BNI/AVF

TABLE OF CONTENTS

		Page
0.	ABSTRACT	3
1.	INTRODUCTION	4
1.1	Purpose of the Validation Summary Report	4
1.2	Use of the Validation Summary Report	5
1.3	References	5
1.4	Definitions of Terms	5
1.5	Configuration used	7
1.6	Implementation dependent characteristics	7
2.	TEST ANALYSIS	8
2.1	Class A Testing	8
2.1.1	Class A Test Procedures	8
2.1.2	Class A Test Results	9
2.2	Class B Testing	9
2.2.1	Class B Test Procedures	9
2.2.2	Class B Test Results	9
2.3	Class C Testing	10
2.3.1	Class C Test Procedures	10
2.3.2	Class C Test Results	10
2.4	Class D Testing	10
2.4.1	Class D Test Procedures	10
2.4.2	Class D Test Results	10
2.5	Class E Testing	11
2.5.1	Class E Test Results	11
2.6	Class L Testing	11
2.6.1	Class L Test Procedures	11
2.6.2	Class L Test Results	11
3.	COMPILER NONCONFORMANCES	12
4.	ADDITIONAL INFORMATION	13
4.1	Compiler Parameters	13
4.2	Testing Information	14
4.2.1	Pre-Test Procedures	14
4.2.2	Control Files	14
4.2.3	Test Procedures	15
4.2.4	Test Analysis Procedures	15
4.2.5	Performance Information	16
4.4.6	Description of Errors in Withdrawn Tests	17
4.2.7	Description of Inapplicable Tests	19
4.2.8	Information derived from the Tests	20
4.2.9	Anomalous Test Results	22
4.2.10	Non conformances Detected	22
5.	SUMMARY AND CONCLUSIONS	23

Section 0 : ABSTRACT

The ALSYS, AlsyCOMP_0001 compiler version 1.0 was tested with version 1.5 of the ACVC validation tests. Version 1.5 of the test suite contained 2231 tests, of which 1966 were applicable to this implementation. Of the applicable tests, 68 were withdrawn due to errors in the tests. Of the remaining applicable correct tests 1897 tests passed and one anomaly was discovered in the test suite.

Section 1 : INTRODUCTION

1.1 Purpose of the Validation Summary Report

This report describes the results of the validation effort for the following Ada compiler :

Host Machine :	VAX-11/785, VAX-11/780, VAX-11/750 VAX-11/730, Micro VAX I
Operating System :	VMS version 4.0
Target Machine :	ALTOS ACS 68000, a Motorola MC 68000 based machine
Operating System :	ALTOS Operating System version 1, a UNIX* System III Operating System (UNIX* is a Trademark of AT&T Bell Laboratories)
Language Version :	ANSI / MIL - STD - 1815 A Ada
Translator Name :	AlsyCOMP_0001
Translator Version :	1.0
Validator Version :	1.5

Testing of this compiler was conducted by BNI under the supervision of the Ada Validation Office (AVO), at the direction of the Ada Joint Program Office. Testing was conducted from December 8th, 1984 through December 11th, 1984 at ALSYS, 29 Avenue de Versailles, 78170 La Celle Saint Cloud, France in accordance with AVO policies and procedures.

The purpose of this report is to document the results of the testing performed on the compiler, and in particular, to :

- identify any language constructs supported by the compiler that do not conform to the Ada standard.
- identify any unsupported language constructs required by the Ada standard.
- describe implementation-dependent behavior allowed by the standard.

1.2 Use of the Validation Summary Report

The Ada Validation Office may make full and free public disclosure of this report in accordance with the "Freedom of Information Act" (5 U.S.C. #552). The results of the validation are only for the purpose of satisfying United Sates Government requirements, and apply only to the computers, operating systems, and compiler version identified in this report.

The Ada Compiler Validation Capability is used to determine insofar as is practical, the degree to which the subject compiler conforms to the Ada standard. Thus, this report is necessarily discretionary and judgmental. The United States Government does not represent or warrant that the statements, or any one of them, set forth in this report are accurate or complete, nor that the subject compiler has no other nonconformances to the Ada standard. This report is not meant to be used for the purpose of publicizing the findings summarized therein.

Questions regarding this report or the validation tests should be sent to the Ada Validation Office at :

Ada Joint Program Office
Room 3D 139 (400 Army Navy Drive)
Pentagon
Washington, D.C. 20301

1.3 References

Reference Manual for the Ada Programming Language, ANSI/MIL-STD-1815A, February 1983.

Ada Validation Organization : Policies and Procedures, Mitre Corporation, June 1982, PB 83-110601.

Ada Compiler Validation Implementers' Guide, SoftTech, Inc., October 1980.

The Ada Compiler Validation Capability, Computer, Vol. 14, N°6, June 1981.

Using the ACVC Tests, SofTech, Inc., November 1981.

Ada Compiler Validation Plans and Procedures, SofTech, Inc., November 1981.

1.4 Definitions of Terms

Class A tests are passed if no errors are detected at compile time. Although these tests are constructed to be executable, no checks can be performed at run-time to see if the test objective has been met ; this distinguishes Class A from Class C tests. For example, a Class A test might check that keywords of other languages (other than those already reserved in Ada) are not treated as reserved words by an Ada implementation.

Class B tests are illegal programs. They are passed if all the errors they contain are detected at compile-time (or link-time) and no legal statements are considered illegal by the compiler.

Class L tests consist of illegal programs whose errors cannot be detected until link time. They are passed if errors are detected prior to beginning execution of the main program.

Class C tests consist of executable self-checking programs. They are passed if they complete execution and do not report failure.

Class D tests are capacity tests. Since there are no firm criteria for the number of identifiers permitted in a compilation, number of units in a library, etc., a compiler may refuse to compile a class D test. However, if such a test is successfully compiled, it should execute without reporting a failure.

Class E tests provide information about an implementation's interpretation of the Standard. Each test has its own pass/fail criterion.

ACVC : Acronym for the Ada Compiler Validation Capability.

AVO : The Ada Validation Office. In the context of this report, the AVO is responsible for directing compiler validation.

BNI : Bureau d'orientation de la Normalisation en Informatique. In the context of this report the BNI conducts Ada validations under contract to the AVO as a satellite facility.

CUSTOMER : The agency requesting the validation (ALSYS).

HOST : The computers on which the compiler executes are : VAX-11/785, VAX-11/780, VAX-11/750, VAX-11/730, Micro VAX I

IG : ACVC Implementers' Guide.

RM : The Ada Language Reference Manual.

TARGET : The computer for which a compiler generates object code is : ALTOS ACS 68000, a Motorola MC 68000 based machine.

VALIDATION : The process of validating a compiler. The term is used interchangeably with test or compiler test.

VALIDATION TESTS : The generic form used to refer to a set of test programs which evaluate how closely a compiler conforms to its language specification. In this report, the term will be used (unqualified) to mean the ACVC tests.

1.5 Configuration

The configuration of the VAX-11/780 which is used is the following :

- 8 Megabytes of Main Memory
- 2 disks : RP07 (512 Mbytes)
  RA81 (450 Mbytes)

The configuration of the ALTOS ACS 68000 14 is the following :

- 1 Megabyte of Main Memory
- 40 Megabytes of Disk

1.6 Implementation dependent characteristics

The following description of implementation dependent characteristics complies with the criteria enumerated in Appendix F of the Ada Reference Manual.

(1) The following pragmas are implementation dependent :

```
BEGIN_COMPILE
BUILT_IN
END_COMPILE
INSTANTIATE_GENERIC
NOW_COMPILE
STOP_COMPILE
TRACE_ON
TRACE_OFF
```

(2) There are no implementation dependent attributes.

(3) The package SYSTEM contains what is requested by the RM and contains overloaded operator declarations to handle operations on addresses. It also contains the two generics FETCH and STORE which are used to perform READ/WRITE operations in memory.

(4) Representation clauses are not supported.

(5) There are no implementation generated names.

(6) Address clauses are not supported, so their interpretation is irrelevant.

(7) Unchecked conversions are allowed between any types which are implemented on the same physical size.

(8) Input-output packages

(a) DIRECT_IO is not supported for unconstrained array types or types with discriminants. The instantiation is accepted, but any call to OPEN or CREATE will raise USE_ERROR.

(b) AlsyCOMP_0001 ignores the parameter FORM in any I/O subprogram call that uses it. The function FORM always returns a null string.

SECTION 2 : TEST ANALYSIS

A Summary of tests processed, by class, is given below, where :

Pr = processed.
NA = found to be inapplicable (not meaningful) for this implementation.
Er = found to be incorrect, and withdrawn from the validation.
P = passed.
A = anomalies.
F = failed.
FE = failed to execute to completion.
FC = failed to compile successfully.
Fs = total of all failures (i.e., F+FE+FC).

The following table shows that AlsyCOMP_0001 passed all applicable correct tests.

Test Class	Pr	NA	Er	P	A	F	FE	FC	Fs	%Pass
A	58	0	0	58	0	0	0	0	0	100
B	796	9	2	785	0	0	0	0	0	100
C	1295	242	66	986	1	0	0	0	0	100
D	14	0	0	14	0	0	0	0	0	100
E	7	0	0	7	0	0	0	0	0	100
L	61	14	0	47	0	0	0	0	0	100
Total	2231	265	68	1897	1	0	0	0	0	100

There were 265 tests in the suite processed and found to be not applicable to the AlsyCOMP_0001 compiler (see Section 4.2.7).

In addition, 68 tests were withdrawn from the test suite because they did not conform to the ANSI/MIL-STD-1815A Standard for the Ada Language standard (see Section 4.2.6 for details).

One test was anomalous (see Section 4.2.9).

2.1 Class A Testing

Class A tests check that legal Ada programs can be successfully compiled. These tests are executed but contain no excutable self-checking capabilities. There were 58 class A test programs processed in this validation.

2.1.1 Class A Test Procedures

Each class A test was separately compiled and executed. However, the only purpose of execution is to produce a message indicating that the test passed.

2.1.2 Class A Test Results

Successful compilation and execution without any error messages indicates the tests passed. Of the 58 class A tests, all 58 were passed.

2.2 Class B Testing

Class B tests check the ability to recognize illegal language usage. There were 796 class B tests processed.

2.2.1 Class B Test Procedures

Each class B test was separately compiled. The resulting test compilation listings are manually examined to see whether every illegal construct in the test is detected. If all errors are not detected, a version of the test is created that contains only undetected illegal constructs. This revised version is recompiled and the result analysed. If all errors are still not detected, the revision process is repeated until a revised test contains only a single illegal construct.

Similarly, if a legal construct is reported to be illegal, a version of the test is created that contains only legal constructs. This revised version is recompiled and the results analysed.

A B test is considered to fail only if a version of the test containing a single illegal construct is accepted by the compiler (i.e., an illegal construct is not detected) or a version containing no errors is rejected (i.e., a legal construct is rejected).

2.2.2 Class B Test Results

There were 796 class B tests presented to the compiler. Of these tests, 9 were found to be inapplicable to this implementation; 2 tests were found to be incorrect (i.e, a conforming compiler would have failed each of these tests). All 785 remaining class B tests passed.

See section 4.2.6 and 4.2.7 for the list of withdrawn tests and inapplicable tests.

Because all errors were not detected when compiling the original tests, the following 11 tests were modified and divided into 24 separate tests by removing the errors detected; the modified tests were then resubmitted to see if the remaining errors would be detected.

B32202A_B.ADA	B32202B_B.ADA	B32202C_B.ADA
B33006A_B.ADA	B37004A_B.ADA	B45102A_AB.ADA
B61012A_B.ADA	B62001B_AB.ADA	BA2002A2.ADA
BC1207A_B.ADA	BC3205C_B.ADA	

All illegal constructs were eventually detected.

2.3 Class C Testing

Class C tests check that legal Ada programs are correctly compiled and executed by an implementation. There were 1295 class C tests processed in this validation.

2.3.1 Class C Test Procedures

Each Class C test is separately compiled and executed. The tests are self-checking and produce PASS/FAIL messages. Any "failed" tests are individually checked to see if they are correct and if they are applicable to the implementation. Any tests that are inapplicable or that do not conform to the Ada standard are withdrawn.

2.3.2 Class C Test Results

Of the 1295 class C tests, 242 were found to be inapplicable to this implementation and 66 were withdrawn because of errors in the tests. One test was found anomalous. The remaining 986 tests passed.

See section 4.2.6, 4.2.7 and 4.2.9 for the list of withdrawn tests, inapplicable tests and anomalous test.

2.4 Class D Testing

Class D tests are executable tests used to check an implementation's compilation and execution capacities. There were 14 class D tests processed in this validation.

2.4.1 Class D Test Procedures

Each class D test is separately compiled and executed. The tests are self-checking and produce PASS/FAIL messages.

2.4.2 Class D Test Results

Of the 14 class D tests, all 14 tests passed.

2.5 Class E Testing

Class E tests are executable tests that provide information about an implementation's interpretation of the Standard in areas where the Standard permits implementations to differ. Each test has its own PASS/FAIL criterion. There were 7 class E tests used in this validation.

2.5.1 Class E Test Results

Of the 7 class E tests, all 7 tests passed.

2.6 Class L Testing

Class L tests check that incomplete or illegal Ada programs involving multiple separately compiled source files are detected at link time and are not allowed to execute. There were 61 test programs processed in this validation.

2.6.1 Class L Test Procedures

Each Class L test was separately compiled and execution was attempted. The tests produce FAIL messages if executed. All "failed" tests are individually checked to see if they are correct and if they are applicable to the implementation. Any tests that are inapplicable or that do not conform to the Ada standard are withdrawn.

2.6.2 Class L Test Results

Of the 61 class L tests, 47 passed and 14 were found to be inapplicable to this implementation (see Section 4.2.7).

SECTION 3 : COMPILER NONCONFORMANCES

For this implementation there were no non conformances and one anomaly to the Ada standard noted during this official validation. The compiler passed all applicable tests.

Section 4.2.9 provides a definition of an anomaly and detailed information describing the anomaly noted in the test suite during the official validation.

SECTION 4 : ADDITIONAL INFORMATION

This section describes in more detail how the validation was conducted.

4.1 Compiler Parameters

Certain tests do not apply to all Ada compilers, e.g., compilers are not required to support several predefined floating point types, and so tests must be selected based on the predefined types an implementation actually supports.

In addition, some tests are parameterized according to the maximum length allowed by an implementation for an identifier (or other lexical element ; this is also the maximun line length), the maximum floating point precision supported, etc. The implementation dependent parameters used in performing this validation were :

- extended_ascii_chars : "abcdefghijklmnopqrstuvwxyz!$%?@[\]^`{}~"
- illegal_external_file_name 1 : /
- illegal_external_file_name 2 : 123456789012345
- maximum digits value for floating point types : 6
- maximum lexical element length : 255
- non_ascii_char_type : (NON_NULL)
- predefined numeric types : SHORT_INTEGER, INTEGER, LONG_INTEGER, FLOAT
- source character set : ASCII

. INTEGER'FIRST :	-32768
. INTEGER'LAST :	32767
. SHORT_INTEGER'FIRST :	-128
. SHORT_INTEGER'LAST :	127
. SYSTEM.MIN_INT :	- (2 ** 31)
. SYSTEM.MAX_INT :	2 ** 31 - 1
. TEXT_IO.COUNT'LAST :	2 ** 31 - 1
. TEXT_IO.FIELD'LAST :	255
. SYSTEM.PRIORITY'FIRST :	1
. SYSTEM.PRIORITY'LAST :	10

4.2 Testing Information

Tests were compiled/executed at ALSYS, La Celle Saint Cloud, France.

4.2.1 Pre-Test Procedures

Prior to testing, the appropriate values for the compiler-dependent parameters were determined. These values were used by AVO to adapt tests that depend on the values. A magnetic tape containing the adapted tests was produced and brought to the testing site.

All tests are marked with a comment on line number 1:

```
-- AVF   ALSYS
```

The following tests were however discovered to be incorrectly adapted and had to be changed on the test site with the appropriate values:

B23003D_AB.TST,
B23003E_AB.TST,
B23003F_AB.TST:
  where the line length had to be extended to 256 characters
B26005A.ADA:
  where a CTRL-M was missing
CE2102C_B.ADA:
  where illegal external file names had to be replaced (see Section 4.1), as it is not classified as a TST test.

Split versions of some B tests (see Section 2.2.2) were produced on site.

4.2.2 Control Files

Alsys provided command procedures that compiled and executed tests automatically.

4.2.3 Test Procedures

All files from the version 1.5 tape were read onto disk. The package REPORT, the procedure CHECK-FILE and the package VAR_STRINGS were first compiled and the corresponding library saved. The tests checking the REPORT package and CHECK-FILE procedure were executed. Then all executable tests were grouped into batch jobs:

(a) batch job number 1 to perform compilation and binding of

- all A tests, except *.TST tests and AE2101A
- C tests up to test C87B19A_B

(b) batch job number 2 to perform compilation and binding of

- all B tests

(c) batch job number 3 to perform compilation and binding of

- C tests from C87B23A_B
- all A*.TST tests and test AE2101A
- all L tests, D tests, E tests

For executable tests from batch jobs number 1 and number 3, the output of the binding phase is sent over a serial link to the two target machines. Linking and execution were performed by a single SHELL procedure on each target machine.

4.2.4 Test Analysis Procedures

On completion of testing, all results were analyzed for failed class A, C, D, E, or L programs, and all class B compilation results were individually analyzed. Analysis procedures are described for each test class in chapter 2.

Tests found to contain errors were withdrawn.

4.2.5 Performance Information

The AlsyCOMP_0001 Source, written in Ada, is transformed into PL/1 by an internal tool, and then compiled by a PL/1 Compiler. This process impacts the present performance of the AlsyCOMP_0001. Performance for a future bootstrapped Compiler will be very much improved. Moreover, the AlsyCOMP_0001 Source comprises many internal traces and debugging routines which have not been suppressed in this version. These also impact the performance of the *AlsyCOMP_0001*.

The following timings have been noted during the validation (hh:mm):

(a) On the host machine: compilation, binding and transfer for all tests

- Total wall-clock time:
  60:04

- Total CPU time for compilation and binding:
  50:28

  (cumulative time for the 3 batch jobs, see Section 4.2.3)

- Total CPU time for transfer:
  05:15

(b) Cumulated on both target machines: linking (with target OS library) and execution:

- Total CPU time for linking:
  10:14

- Total CPU time for execution:
  01:17

Note: one withdrawn test (C38104A) loops at execution time for 20 minutes CPU before raising STORAGE_ERROR.

4.2.6 Description of Errors in Withdrawn Tests

The following tests in Version 1.5 of the ACVC did not conform to the ANSI Ada standard and were withdrawn for the reasons given below:

- C37011A-B: Sliding of array bounds is not permitted for the default initialization of array components of record objects (see 3.2.1(16)).
- C38104A-B: An incomplete type with discriminants was constrained before its full declaration occurred. An implementation is allowed to reject such subtype indications because of an ambiguity in the language (see AI-00007/04).
- C43103B-B: A non-null range had a bound that was outside the index subtype.
- B43201B-B: An OTHERS choice is only allowed as the second component association if the applicable index constraint is static, and an index constraint is only static if all the discrete ranges in the constraint are static.
- C45321*-B., C45521*-B: Incorrect values were used for values assigned to variables having a floating point subtype.
- C52001B-AB: An equality comparison for non-model numbers (e.g., 23.4 = 23.4) has an implementation defined value.
- C52007A-B: A comparison of INTEGER'LAST with SYSTEM.MAX_INT will raise NUMERIC_ERROR if SYSTEM.MAX_INT exceeds INTEGER'LAST, since the implicit conversion of SYSTEM.MAX_INT to INTEGER will raise NUMERIC_ERROR.
- C52102A-AB, C52102B-AB: The result of concatenating slices of an array of characters had an upper bound that did not belong to the array's index subtype because the array was declared to have an index subtype 1..10 (or 1..9) instead of subtype INTEGER.
- C52103X-B: A test assumed that a slice would be performed even if it raised NUMERIC_ERROR.

- C87B10A-B: Literal values were used that were outside an integer base type for some implementations.
- B87B23B-B: A tricky case of overload resolution marked OK was actually ambiguous.
- C930BDA-B: An attempt to activate a task before its body is elaborated should raise TASKING_ERROR, not PROGRAM_ERROR.
- C95008A: It was possible for an entry call to call a terminated task, depending on the implementation.
- C95009A: An unintended raise condition in a tasking test allowed a null access value to be dereferenced before the access variable was assigned the access value of an allocated task.
- CE3103A-B: A test would print a failed message if RESET raised USE_ERROR.
- CE3804E-B: A test contained a non-model number (1.35) for which an equality comparison was expected to always yield true.

Although withdrawn from ACVC version 1.5, these tests were submitted to AlsyCOMP_0001 during the validation. AlsyCOMP_0001 had a coherent behavior on all these tests.

4.2.7 Description of Inapplicable Tests

There were 265 tests that were found to be inapplicable. All of them have been compiled and eventually runned and AlsyCOMP_0001 behaved correctly for all these tests. These tests were :

B86001CP_AB.DEP	B86001CQ_AB.DEP	B86001DT_AB.TST
BC3204C*_B.ADA	BC3205D*_B.ADA	C24113C..Y_B.DEP
C34001F_B.DEP	C34001G_B.DEP	C35702A_AB.DEP
C35702B_AB.DEP	C35705C..Y_B.DEP	C35706C..Y_B.DEP
C35707C..Y_B.DEP	C35708C..Y_B.DEP	C35802C..Y_B.DEP
C45241C..Y_B.DEP	C45424C..Y_B.DEP	C45521C..Z_B.DEP
C45621C..Z_B.DEP	C52104X_B.ADA	C52104Y_B.ADA
C55B16A_AB.DEP	C86001F_B.DEP	C87B62A..C_B.DEP
LA3004A*_AB.DEP	LA3004B*_AB.DEP	

B86001DT_AB.TST : This test is not applicable because AlsyCOMP_0001 has no predefined numeric type other than INTEGER, FLOAT, SHORT INTEGER, LONG_INTEGER. The macro name $NAME was set to NO SUCH_TYPE and the declaration of a procedure name NO_SUCH TYPE is then legal.

BC3204C*_B.ADA
BC3205D*_B.ADA : These tests should be classified DEP tests because the body is a separate unit. These tests are inapplicable because AlsyCOMP_0001 requires the body to be in the same compilation as its declaration.

C24113C..Y_B.DEP
C35705C..Y_B.DEP
C35706C..Y_B.DEP
C35707C..Y_B.DEP
C35708C..Y_B.DEP
C35802C..Y_B.DEP
C45241C..Y_B.DEP
C45421C..Y_B.DEP
C45424C..Y_B.DEP
C45621C..Z_B.DEP : These tests are inapplicable because AlsyCOMP_0001 limits digits to 6.

B86001CP_AB.DEP
C34001F_B.DEP
C35702A_AB.DEP : These tests are inapplicable because AlsyCOMP_0001 does not support SHORT_FLOAT.

B86001CQ_AB.DEP
C34001G_B.DEP
C35702B_AB.DEP : These tests are inapplicable because AlsyCOMP_0001 does not support LONG_FLOAT.

C55B16A_AB.DEP
C87B62A..C_B.DEP : These tests are inapplicable because AlsyCOMP_0001 does not support representation clauses.

C86001F_B.DEP : This test is inapplicable because AlsyCOMP_0001 rejects the recompilation of SYSTEM at compilation-time.

LA3004A*_AB.DEP
LA3004B*_AB.DEP : These tests are inapplicable because AlsyCOMP_0001 does not support pragma INLINE. These tests ignore the pragma and are processed correctly.

C52104X_B.ADA
C52104Y_B.ADA : These tests are inapplicable because AlsyCOMP_0001 does not support pragma PACK. These tests ignore the pragma and are processed correctly.

4.2.8 Information Derived from the Tests

Processing of the following tests indicated support as described below for a variety of implementation options examined by the tests.

- E24101A-B.TST : if a based integer literal has a value exceeding SYSTEM.MAX_INT, an implementation may either reject the compilation unit at compile time or raise NUMERIC_ERROR at run-time. Raising NUMERIC_ERROR at run time is preferred, since it makes programs compilable for a wider variety of implementations (and the numeric literal might occur in an unexecutable portion of code). This test showed that AlsyCOMP_0001 raises NUMERIC_ERROR at run-time for a compilation unit containing an integer literal exceeding SYSTEM.MAX_INT.

- B26005A.ADA : This test contains all the ASCII control characters in string literals. AlsyCOMP_0001 rejected all these strings.

- D29002K-B.ADA : This test declares 713 identifiers and was successfully processed by the implementation.

- C36202A-B, E52103Y : These tests declare array types having a dimension whose length exceeds INTEGER'LAST. An implementation is allowed to raise NUMERIC_ERROR for such type declarations. AlsyCOMP_0001 implementation raised no exception condition for these tests.

- C4A002A-AB.ADA, D4A002B-AB.ADA,C4A004A-AB.ADA, D4A004B-AB.ADA : These tests contain universal integer calculations requiring 32 and 64 bits of accuracy, i.e., values that exceed SYSTEM.MAX_INT are used. An implementation is allowed to reject programs requiring such calculations; AlsyCOMP_0001 compiler passed all these tests.

- C52103X-AB.ADA, C52104X-B.ADA, C52104Y.ADA : These tests declare Boolean arrays with INTEGER'LAST+3 components. An implementation may raise NUMERIC_ERROR at the type declaration or STORAGE_ERROR when array objects of these types are declared, or it may accept the type and object declarations. AlsyCOMP_0001 compiler raised no exceptions.

- A series of tests (D55A03-AB) check to see what level of loop nesting is allowed by an implementation. Tests containing loops nested 65 levels deep were passed.

- D56001B-AB contains blocks nested 65 levels deep. This test was passed.

- C94004A-B.ADA : This test checks to see what happens when a library unit initiates a task and a main program terminates without insuring that the library unit's task is terminated. An implementation is allowed to terminate the library unit task, or it is allowed to leave the task in execution. This test showed that such library tasks terminate when the main program terminates.

AE2101C-B.DEP : This test can only be compiled if the sequential and direct input-output packages can be instantiated with unconstrained array types and unconstrained types with discriminants. AlsyCOMP_0001 compiled this test.

. EE3102C-B.ADA : This test confirmed that an Ada program can open an existing file in OUT_FILE mode, and can create an existing file in either OUT_FILE or IN_FILE mode.

. CE3111A-B.DEP : This test confirmed that more than one internal file could be associated with the same external file.

4.2.9 Anomalous Test Results

An anomaly is test behavior that suggests the implementation does not conform to the Standard. The test behavior is not, however, considered to demonstrate nonconformance to the Standard for purposes of this validation. The reasons for discounting an anomalous test result vary, depending on the anomaly. The existence of an anomaly sometimes means that new tests should be added to the suite to demonstrate the suspected nonconformance in a more convincing manner. Sometimes, when the anomaly is the result of procedural errors during the validation, the existence of an anomaly requires revisions in validation procedures.

One anomaly was noted for this validation:

Proper execution of test C94004A depends on the order of elaboration between package body REPORT and package body PKG. If package body REPORT has not been elaborated before task PKG.A is activated, the function call IDENT_INT(0) raises PROGRAM_ERROR (Access Before Elaboration) and TASKING_ERROR is raised in the body of package PKG.

It was demonstrated that if a pragma ELABORATE (REPORT) is inserted at the beginning of package body PKG, AlsyCOMP_0001 executed this test correctly.

This test shall be modified in a subsequent version of the test suite.

4.2.10 Nonconformances Detected

No Nonconformances were detected for this validation.

SECTION 5 : SUMMARY AND CONCLUSIONS

The Ada Validation Office identified 2231 of the ACVC version 1.5 tests as being potentially applicable to the validation of the AlsyCOMP_0001 compiler hosted on VAX-11/785, VAX-11/780, VAX-11/750, VAX-11/730 or MicroVAX I and targeted for ALTOS ACS 68000. Of these, 68 were withdrawn due to test errors, and 265 were determinated to be inapplicable after they were processed. One anomalous test correctly ran after modification.

The AVO considers these results to show acceptable compliance to ANSI/MIL-STD-1815A.

END

FILMED

6-85

DTIC