TMS320 DSP Algorithm Interoperability Standard

QualiTI Compliance Testing

Compliance Test Report

Date	Thursday, June 13, 2002, 03:39, India Standard Time
Vendor	TI
Algorithm	EQ
Generic Interface	None
Version	V2.0
Serial Number	2002
Test Result	**PASS**

Test Notes

This report has been generated with QualiTI Version 2.0.2 Beta - the new XDAIS compliance test suite.

All references to Rules and Guidelines are from the literature numbered SPRU352D, January 2001.

The Rules 17, 18, 25, 26, 27, 28, 29, 30, IDMA 1, IDMA 2, IDMA 3, IDMA 4, IDMA 5 are not applicable and hence were not tested.

If worst case heap memory creation parameter is not provided explicitly then QualiTI tests rule 19 with default instance creation parameter

If you have any questions or comments about the test results below, please send an email message to: algtesters@list.ti.com

Documentation:

C:\Arnab\EQc55x\EQ Reference.doc

Header File(s):

C:\Arnab\EQc55x\Algorithm\Include\xdas.h
C:\Arnab\EQc55x\Algorithm\Include\eq_ti_prvt.h
C:\Arnab\EQc55x\Algorithm\Include\ialg.h
C:\Arnab\EQc55x\Algorithm\Include\ialgmods.h
C:\Arnab\EQc55x\Algorithm\Include\ieq.h
C:\Arnab\EQc55x\Algorithm\Include\std.h
C:\Arnab\EQc55x\Algorithm\Include\version.h
C:\Arnab\EQc55x\Algorithm\Include\eq_ti.h

Algorithm Archive(s):

C:\Arnab\EQc55x\Algorithm\Debug\eq_ti.l55

Source File Defining IMOD_PARAMS:

C:\Arnab\EQc55x\Interface\Src\ieq.c

Source File Defining Worst Case Heap Memory Parametrs:

C:\Arnab\EQc55x\Algorithm\Src\eq_ti_ialg.c

Screening Test: All algorithms must have at least one top level header file named mod_vend.h that includes all the other header file that are provided with the algorithm. In addition the following external symbols MOD_VEND_IALG and

MOD_VEND_IMOD / MOD_VEND_IGENERICINTERFACE must be present for the algorithm to be tested.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 1: All algorithms must follow the run-time conventions imposed by TI's implementation of the C programming language.

Report: **Passed**

Details: Vendor states compliance in the documentation

Rule 2: All algorithms must be reentrant within a preemptive environment (including time-sliced preemption)

Report: **Passed**

Details: Vendor states compliance in the documentation

Rule 3: All algorithm data references must be fully relocatable (subject to alignment requirements). That is, there must be no "hard coded" data memory locations.

Report: **Passed**

Details: Vendor states compliance in the documentation

Rule 4: All algorithm code must be fully relocatable. That is, there can be no hard coded program memory locations.

Report: **Passed**

Details: Vendor states compliance in the documentation

Rule 5: Algorithms must characterize their ROM-ability; i.e., state whether they are ROM-able or not.

Report: **Passed**

Details: Characterized by the vendor

Rule 6: Algorithms must never directly access any peripheral device. This includes but is not limited to on-chip DMAs, timers, I/O devices, and cache control registers.

Report: **Passed**

Details: Vendor states compliance in the documentation

Rule 7: All header files must support multiple inclusions within a single source file.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 8: All external definitions must be either API identifiers or API and vendor prefixed.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 9: All undefined references must refer either to the operations specified in Appendix B (a subset of C runtime support library functions and the DSP/BIOS) or other XDAIS-compliant modules.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 10: All modules must follow the naming conventions of the DSP/BIOS for those external declarations disclosed to the client.

Report: **Passed**

Details: Vendor states compliance in the documentation

```
QualiTI has produced a list of exposed external symbols for the user to
verify that DSP/BIOS naming conventions have been followed.

List of exposed external symbols
EQ_TI_init
EQ_TI_exit
EQ_TI_IALG
EQ_TI_IEQ
```

Rule 11: All modules must supply an initialization and finalization method.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 12: All algorithms must implement the IALG interface.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 13: Each of the IALG methods implemented by an algorithm must be independently relocatable.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 14: All abstract algorithm interfaces must derive from the IALG interface.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 15: Each XDAIS-compliant algorithm must be packaged in an archive which has a name that follows a uniform naming convention.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 16: Each XDAIS-compliant algorithm header must follow a uniform naming convention.

Report: **Passed**

Details: QualiTI passes this rule after testing

Rule 19: All algorithms must characterize their worst-case heap data memory requirements (including alignment).

Report: Passed

Details: QualiTI passes this rule after testing

Table of heap memory usage

memTab	Given Attribute	Calculated Attribute	Given Size (bytes)	Calculated Size (bytes)	Given Align	Calculated Align	Given Space	Calculated Space
0	PERSISTENT	PERSISTENT	20	20	0	0	IALG_DARAM0	IALG_DARAM0
1	PERSISTENT	PERSISTENT	200	200	0	0	IALG_DARAM0	IALG_DARAM0
2	PERSISTENT	PERSISTENT	24	24	32	32	IALG_SARAM	IALG_SARAM

Rule 20: All algorithms must characterize their worst-case stack space memory requirements (including alignment).

Report: Passed

Details: Characterized by the vendor

Rule 21: Algorithms must characterize their static data memory requirements.

Report: Passed

Details: QualiTI passes this rule after testing

Table of static data memory usage

Section Number	Object File Name	Section Name	Given Size (bytes)	Calculated Size (bytes)	Given Alignment
0	eqcoeff.obj	.const	222	222	0
1	eq_ti_vtable.ob	.const	64	64	0

Rule 22: All algorithms must characterize their program memory requirements.

Report: Passed

Details: QualiTI passes this rule after testing

Table of program memory usage

Section Number	Section Name	Given Size (bytes)	Calculated Size (bytes)	Given Alignment
0	.text	1036	1036	0
1	.text:algInit	182	182	0
2	.text:algAlloc	52	52	0
3	.text:algNumAlloc	4	4	0
4	.text:algFree	22	22	0
5	.text:exit	2	2	0
6	.text:init	2	2	0

Rule 23: All algorithms must characterize their worst-case interrupt latency for every operation.

Report: Passed

Details: Characterized by the vendor

Rule 24: All algorithms must characterize the typical period and worst-case execution time for each operation.

Report: **Passed**

Details: Characterized by the vendor

Rule 31: All C55X algorithms must document the content of the stack configuration register that they follow.

Report: **Passed**

Details: Characterized by the vendor

Rule 32: All C55X algorithms must access all static and global data as far data; also the algorithms should be instantiable in a large memory model.

Report: **Passed**

Details: Vendor states compliance in the documentation

Rule 33: C55X algorithms must never assume placement in on-chip program memory; i.e., they must properly operate with program memory operated in instruction cache mode.

Report: **Passed**

Details: Vendor states compliance in the documentation

Rule 34: C55X algorithms must document the instance number of the IALG_MemRec structure that is accessed by the B-Bus.

Report: **Passed**

Details: Characterized by the vendor