

TM5 TO OCCT CONVERSION GUIDE

01 First steps

This document will guide you through the process of converting TM5 .cfg files to a valid OCCT memory test sequence. OCCT can reproduce a TM5 test doing similar operations, although with a different implementation.

General rules

1. TM5 forces a check after every test configured in its cfg file

OCCT makes this optional.

Reproducing TM5 behavior means every test block in OCCT should be followed by a Check block

2. TM5 adds a RefreshStable test automatically between each test

OCCT makes this optional.

Reproducing TM5 behavior means every block should be followed by a RefreshStable block and a Check block.

3. The two abovementioned rules must be combined

Combining the two rules, this means that a simple MirrorMove in TM5 will look like this in OCCT :

- MirrorMove
- Check
- RefreshStable
- Check

4 BE CAREFUL, test order is controlled by Test Sequence in TM5

Tests may be named Test0, 1, 2 etc. in a cfg file of TM5, but the actual execution Order is defined by the TestSequence attribute.

If TestSequence = 3,5,1 , this means TM5 will execute Test3, Test5, Test1 in this order.

Take care of reproducing the same order within OCCT, which uses a basic top-down approach for execution.

5. TM5 uses a complex calculation for defining the number of iterations of each test

You can calculate the number of iterations in TM5 with the following formula :

Iterations = Tg * (Ti * C)

- Tg is "Time (%)" in [Main Section]
- Ti is "Time (%)" in the individual [TestX] section
- C is a constant specified for each test type

SimpleTest = 5

MirrorMove = 4

MirrorMove128 = 4

You can recreate the number of iterations using a loop block in OCCT.

02 Test parameters

All tests also have parameters that change their behavior.

We will describe how they can be ported to OCCT - If a parameter isn't mentioned in this list, it means it's unused in this context.

1 - Simple test

Pattern Mode (TM5) changes the type of block in OCCT :

0 → TM5 Simple

1 → TM5 Inverted

2 → TM5 Arithmetic

any other value will fall back to 0

Pattern Param0 (TM5) maps to Seed 1 in OCCT

- only used by TM5 Arithmetic
- any value larger than 8 hex characters will get truncated by TM5, keeping the first 8 hex characters

Pattern Param1 (TM5) maps to Seed 2 in OCCT

- only used by TM5 Arithmetic
- any value larger than 8 hex characters will get truncated by TM5, keeping the first 8 hex characters

Parameter (TM5) maps to Jump in OCCT

- TM5 is in base 10, OCCT uses hex, so convert it accordingly

2 - MirrorMove

Parameter (TM5) changes the type of block in OCCT

1 → Mirror Move 1

2 → Mirror Move 2

4 → Mirror Move 4

3 isn't supported by OCCT as of today

Any other value will fall back to 1

3 - MirrorMove128

Parameter (TM5) maps to Jump (OCCT)

- TM5 is in base 10, OCCT uses hex, so convert it accordingly

Test Block Size (TM5) isn't supported in the current beta of OCCT.

As an example, here is a valid replication of a sequence:

```
[Main Section]
Config Name=OCCT Demo
Config Author=Dracau
Cores=0
Tests=2
Time (%)=400
Cycles=1
Language=0
Test Sequence=2,1

[Test0]
Enable=1
Time (%)=100
Function=RefreshStable
DLL Name=bin\MT0.dll
Pattern Mode=0
Pattern Param0=0x0
Pattern Param1=0x0
Parameter=0
Test Block Size (Mb)=0

[Test1]
Enable=1
Time (%)=50
Function=MirrorMove128
DLL Name=bin\MT0.dll
Pattern Mode=0
Pattern Param0=0x0
Pattern Param1=0x0
Parameter=2
Test Block Size (Mb)=0

[Test2]
Enable=1
Time (%)=100
Function=SimpleTest
DLL Name=bin\MT0.dll
Pattern Mode=2
Pattern Param0=0xA50CC75A
Pattern Param1=0x50CBA5EA
Parameter=32
Test Block Size (Mb)=0
```

Block #	Function	Parameter 1	Parameter 2	Parameter 3
#1	Loop Start	20		
#2	TMS Arithmetic	0x20	0xA50CC75A	0x50CBA5EA
#3	Check			
#4	Loop End			
#5	TMS Refresh	SSE		
#6	Check			
#7	Loop Start	8		
#8	Mirror Move 128	SSE	0x20	0xA50CC75A, 0x50CBA5EA
#9	Check			
#10	Loop End			

You can repeat this process while following the whole Test Sequence in [Main Section], then you can start testing or modifying the config with different test blocks or parameters !

