Disk on Module (DoM) Durability Testing

Project Manager: Senad Jordanovic
Tests and Content: Eldar Avdic

Chapter 1: Introduction

Disk on Module

Durability testing

Goals

DoM durability testing, which was done on one of the officeBOX machines, was designed to test the **D**isk **o**n **M**odule durability. Device was 512 MB and manufacturer claims for the device are:

- MTBF – 1,000,000 hours (**M**ean **T**ime **B**efore **F**ailure)
- 4,000,000 R/W cycles

From this information manufacturer extrapolated a 10 year life span for this device, which of course depends heavily on the application.

Like the manufacturer, we couldn't really test the DoM for 1,000,000 hours, but we could test Read/Write cycles by simulating extensive usage of storage device and the testing consisted of two parts:

- Test 1 was our custom made test where we placed the machine under our Stress Test integrated in PBXware. Stress test placed fake calls on the machine so we can speed up the process of several thousand calls being made.
- Test 2 used Badblocks as its main tool for testing.

So, our goal was to test the durability which means how „long“ the DoM device could work taking in concern manufacturer given specifications, and of course our tests.

NOTE: **D**isk **o**n **M**odule is a solid state disk which means that it is a pure flash memory with a standard connector and controller. There are no moving parts making it imune to vibrations and droping, also they don't make any noise and consume less than 25% power than conventional disk drives. Because of the NAND technology and random access nature, access time of any location in the DoM is less than 1 ms which makes it faster than any hard drive in everyday usage.

New wear leveling techniques, invented because the main problem of any flash drive is durability, extends their life span more than 10 years. Comparing that to hard disks 3-6 years, it's a great improvement. Of course all these figures are theory, but they are given with probability of 98% by their manufcturer.

Setup

The setup of these tests was really simple.

From the hardware point of view we needed just one officeBOX machine, which had 512 Mbytes of RAM and 512 Mbytes of DoM as stated above. Machine had our PBXware installed on it, and we used several linux tools for this: **vmstat**, **free**, **du**, **df** and **badblocks**.

First test was done to determine how the DoM will behave when exposed to every days work. Difference is that we ''compressed'' the time, meaning that we sped up the process of calls being made. For this we used our Stress Test which is available in PBXware, and set it to work as having 10 to 20 concurrent calls.

Second test was really a test of durability in a way that we took one specific part of DoM and wrote random data on it more than 100,000 times until the blocks from that section started to corrupt.

Chapter 2: Testing

Test 1: Stress Test

Like stated above this test was done to determine how the DoM is performing in every day use. Stress Test was chosen because of the time we had for testing. Making 120,000 calls by hand isn't very logical, but placing fake calls using stress test ability is very convenient in a way that you can place 5000 calls in two hours.

Of course, for this we didn't want it to be absolutely synthetic test, so we changed parameters a bit for every set of calls. This means that we changed number of calls per set, or we changed number of concurrent calls, or duration of calls.

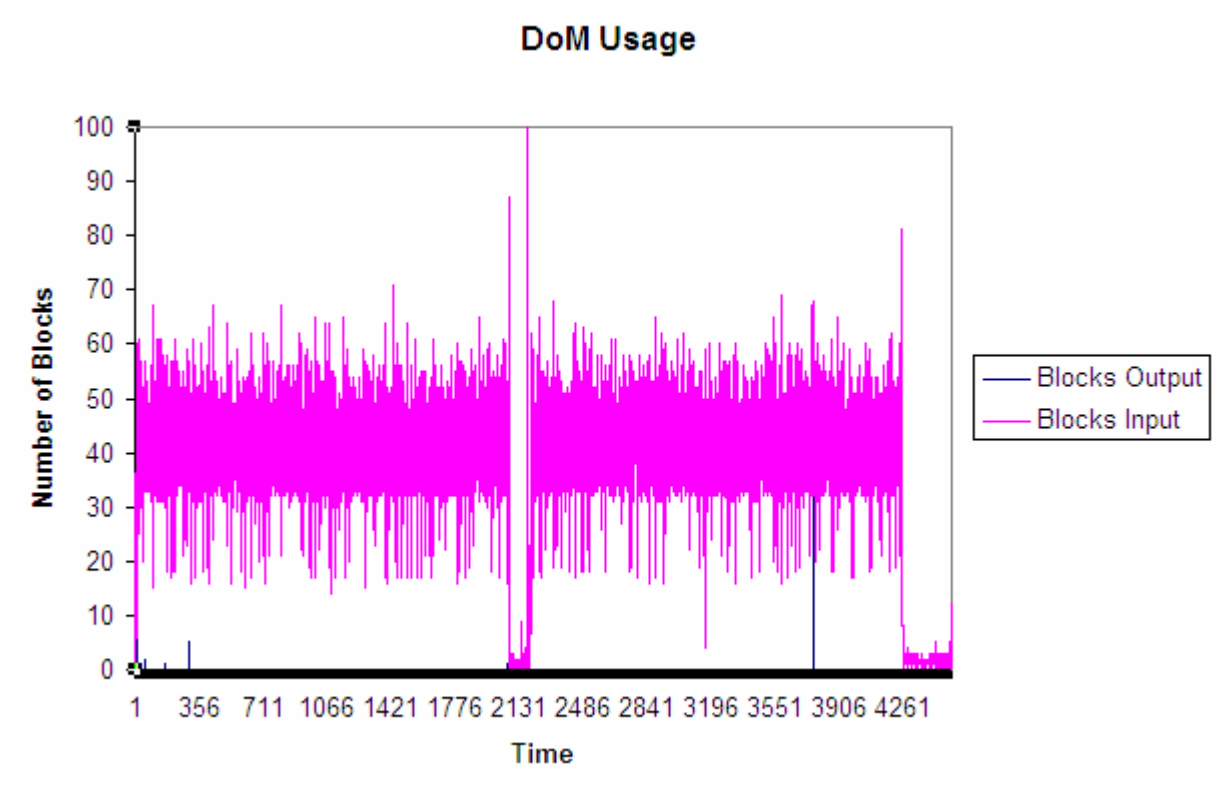
In this test we measured how much disk space given set of calls has taken, average disk space per call, and we examined DoM behavior during this test, meaning that we checked it's average speed – blocks written to DoM per second.

It's important to mention that the PBXware image takes about 80 MB of space and the PBXware system, when extracted, takes 167 MB of disk space. Now, our DoM was 512 MB in size which left us with 265 MB of empty space to test on.

After several set of tests we had to delete accumulated PBXware logs, which were about 78 MB per 10,000 calls, and taking this in account, we practically simulated Read/Write cycles, although the number of calls wasn't big enough to reach the 4,000,000 R/W cycles.

Following charts are for comparison of DoM write speed so we can see if there are any changes during this test.

1st set of calls



Number of calls: 20,000

Total number of calls: 20,000

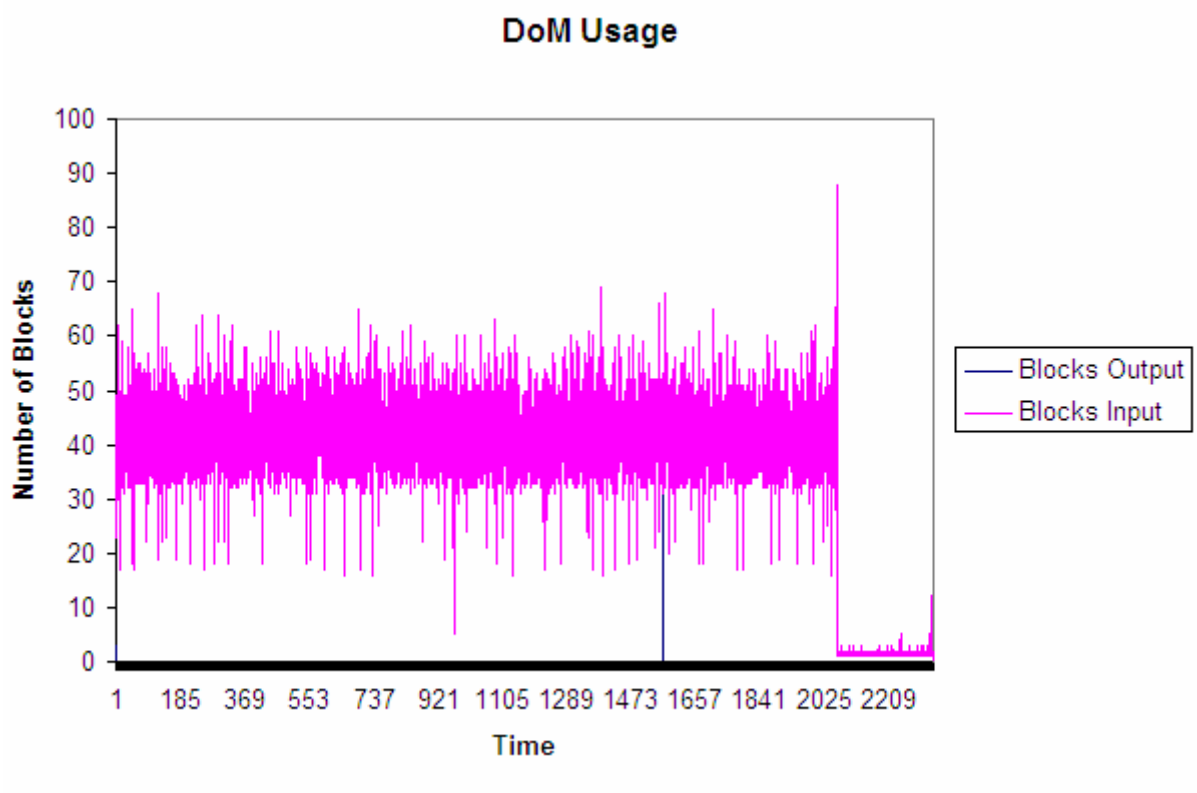
Duration of a call: 15 seconds

Data written to DoM: 158 MB

Average number of blocks written: 38

Average usage per call: 158 MB / 20,000 = 8 KB per call

2nd set of calls



Number of calls: 5,000

Total number of calls: 25,000

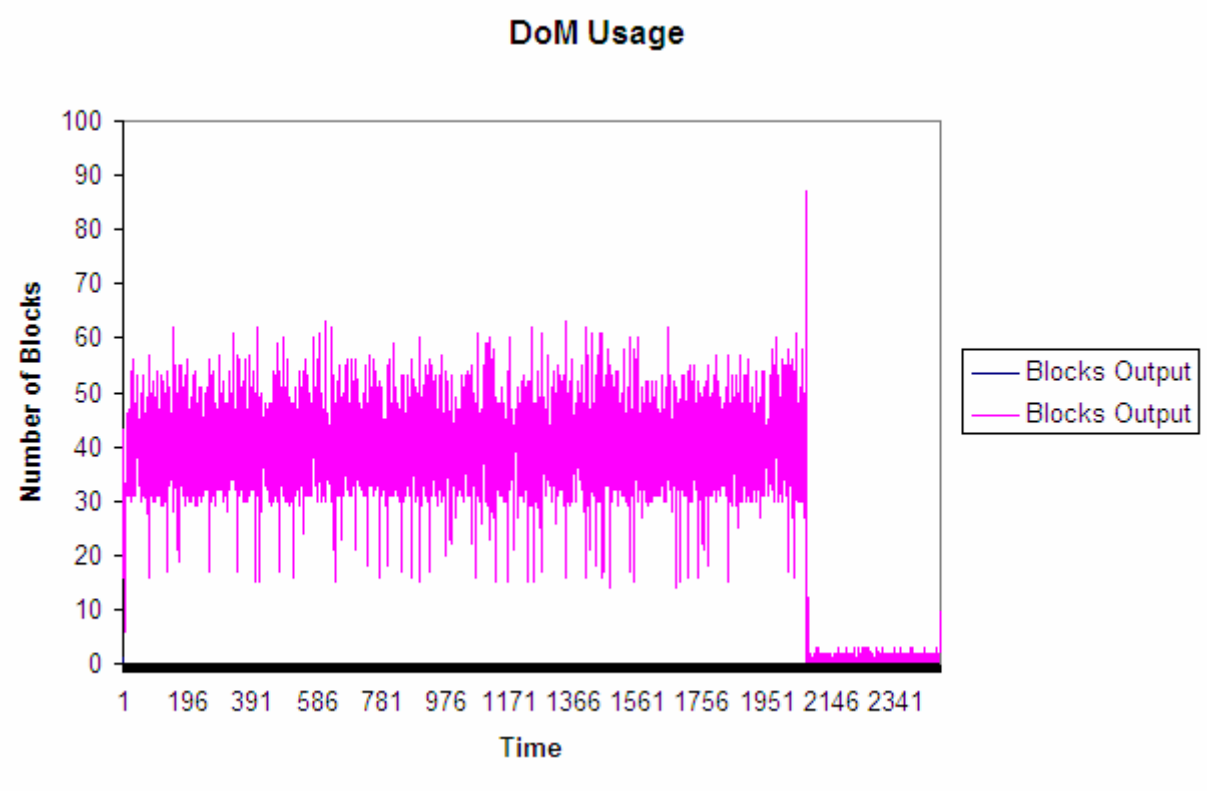
Duration of a call: 15 seconds

Data written to DoM: 39 MB

Average number of blocks written: 37

Average usage per call: 39 MB / 5,000 = 7,9 KB per call

3rd set of calls



Number of calls: 5,000

Total number of calls: 30,000

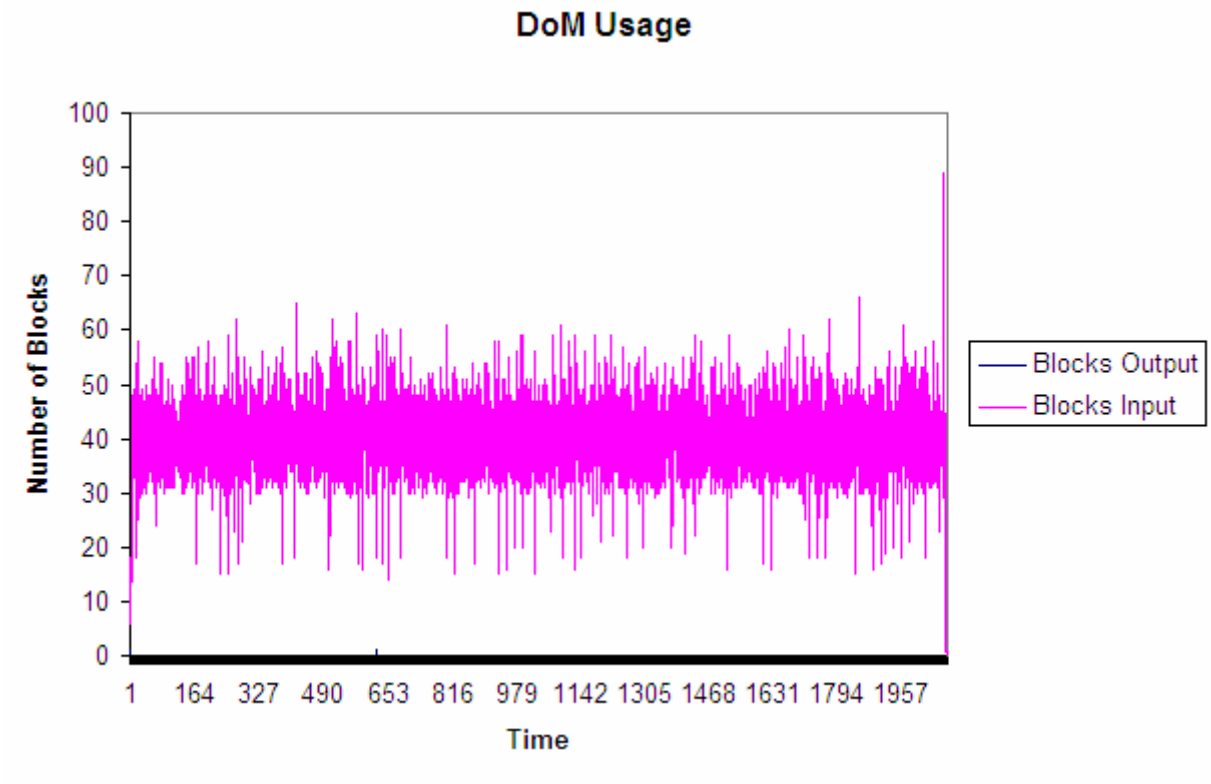
Duration of a call: 20 seconds

Data written to DoM: 40 MB

Average number of blocks written: 34

Average usage per call: 40 MB / 5,000 = 8,1 KB per call

4th set of calls



Number of calls: 5,000

Total number of calls: 35,000

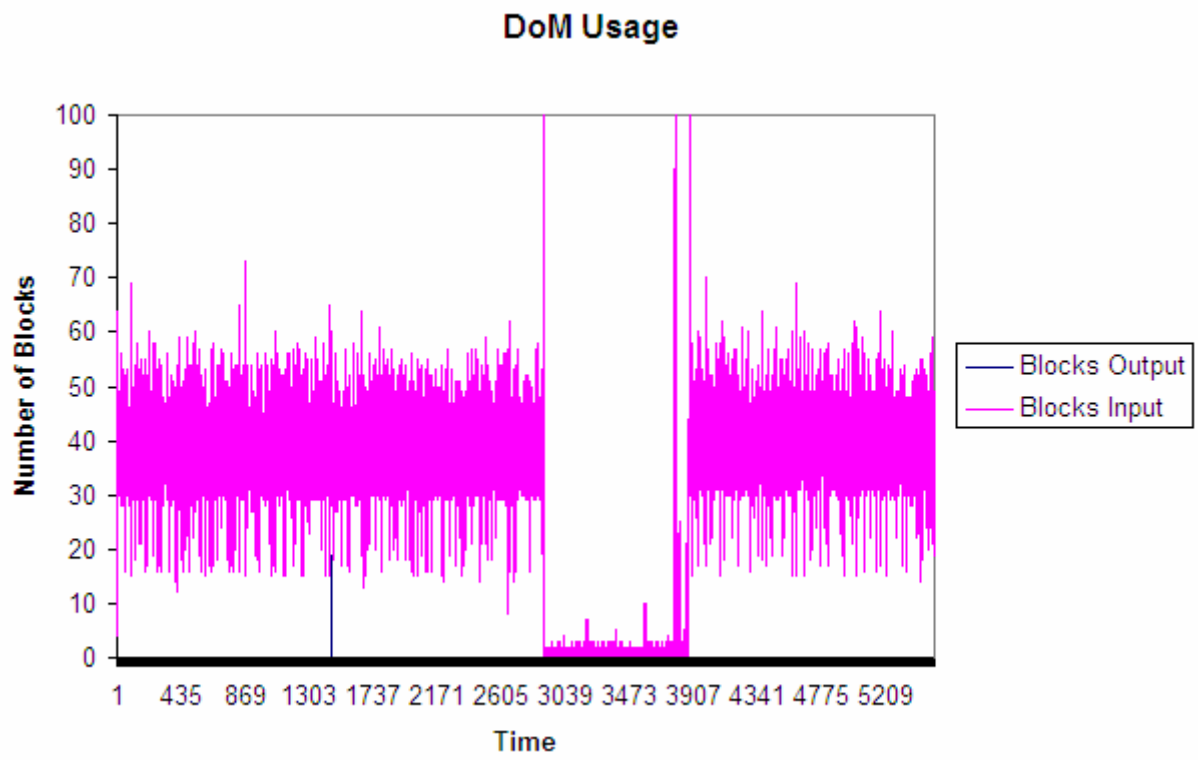
Duration of a call: 15 seconds

Data written to DoM: 39 MB

Average number of blocks written: 39

Average usage per call: 39 MB / 5,000 = 7,9 KB per call

5th set of calls



Number of calls: 10,000

Total number of calls: 45,000

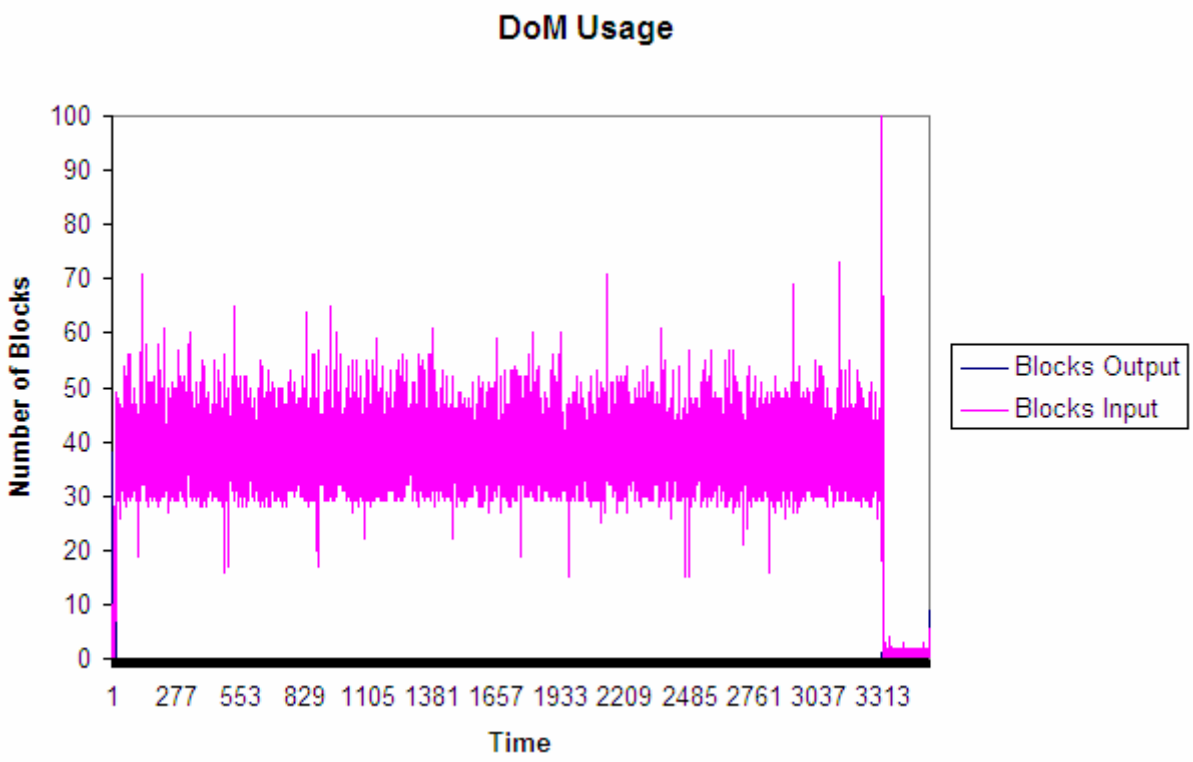
Duration of a call: 10 seconds

Data written to DoM: 78 MB

Average number of blocks written: 33

Average usage per call: 78 MB / 10,000 = 7,9 KB per call

6th set of calls



Number of calls: 7,000

Total number of calls: 52,000

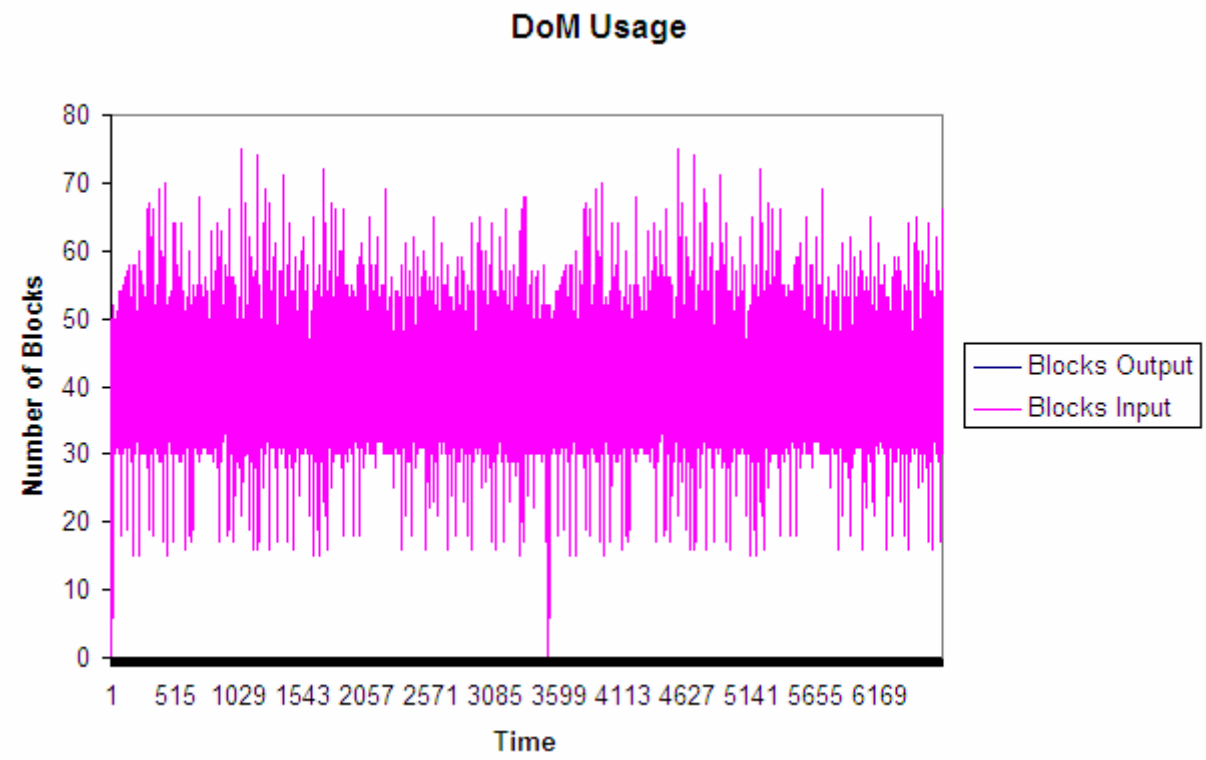
Duration of a call: 10 seconds

Data written to DoM: 54 MB

Average number of blocks written: 36

Average usage per call: 54 MB / 7,000 = 8 KB per call

7th set of calls



Number of calls: 13,000

Total number of calls: 65,000

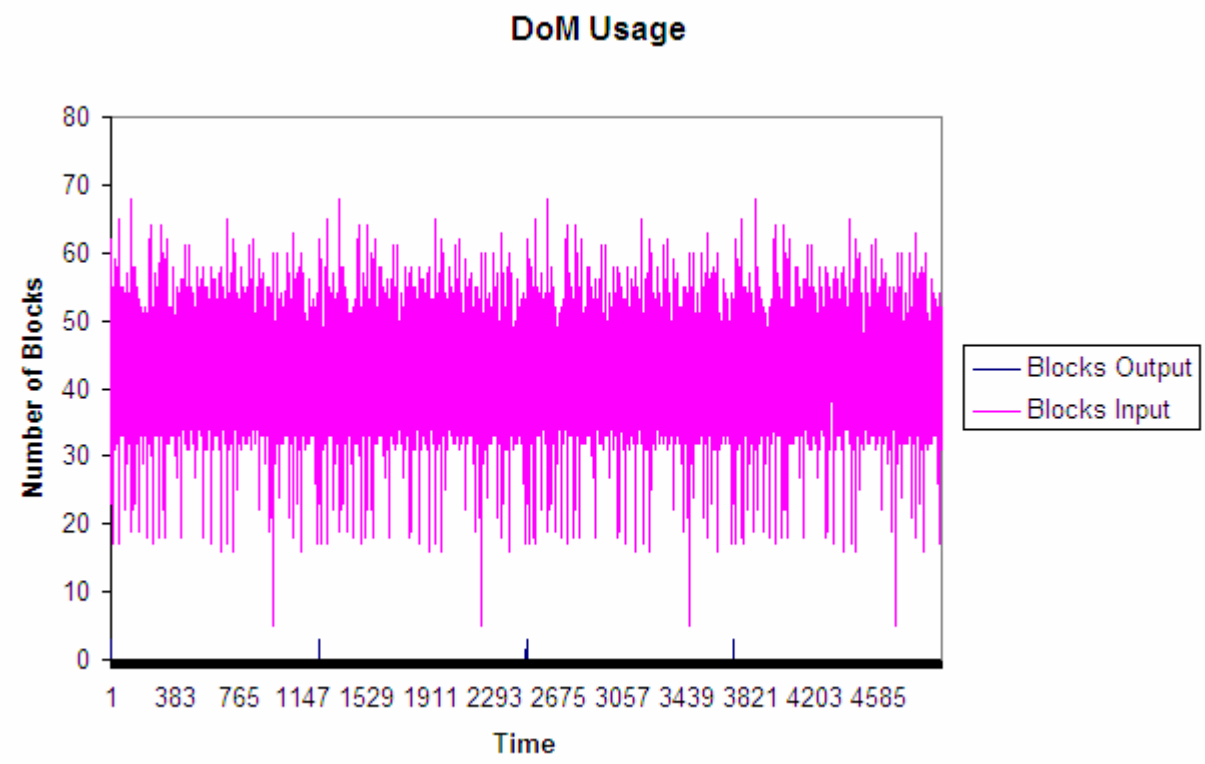
Duration of a call: 10 seconds

Data written to DoM: 102 MB

Average number of blocks written: 40

Average usage per call: 102 MB / 13,000 = 8,1 KB per call

8th set of calls



Number of calls: 8,000

Total number of calls: 73,000

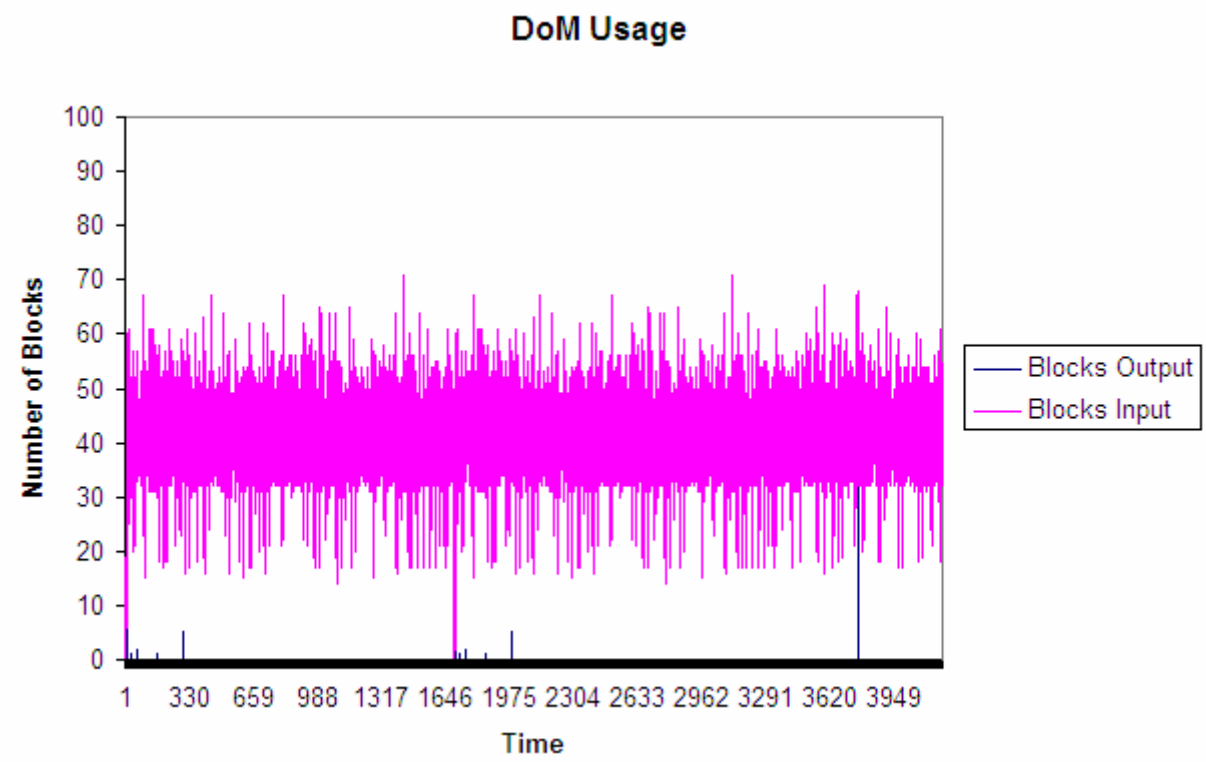
Duration of a call: 10 seconds

Data written to DoM: 60 MB

Average number of blocks written: 41

Average usage per call: 60 MB / 8,000 = 7,8 KB per call

9th set of calls



Number of calls: 7,000

Total number of calls: 80,000

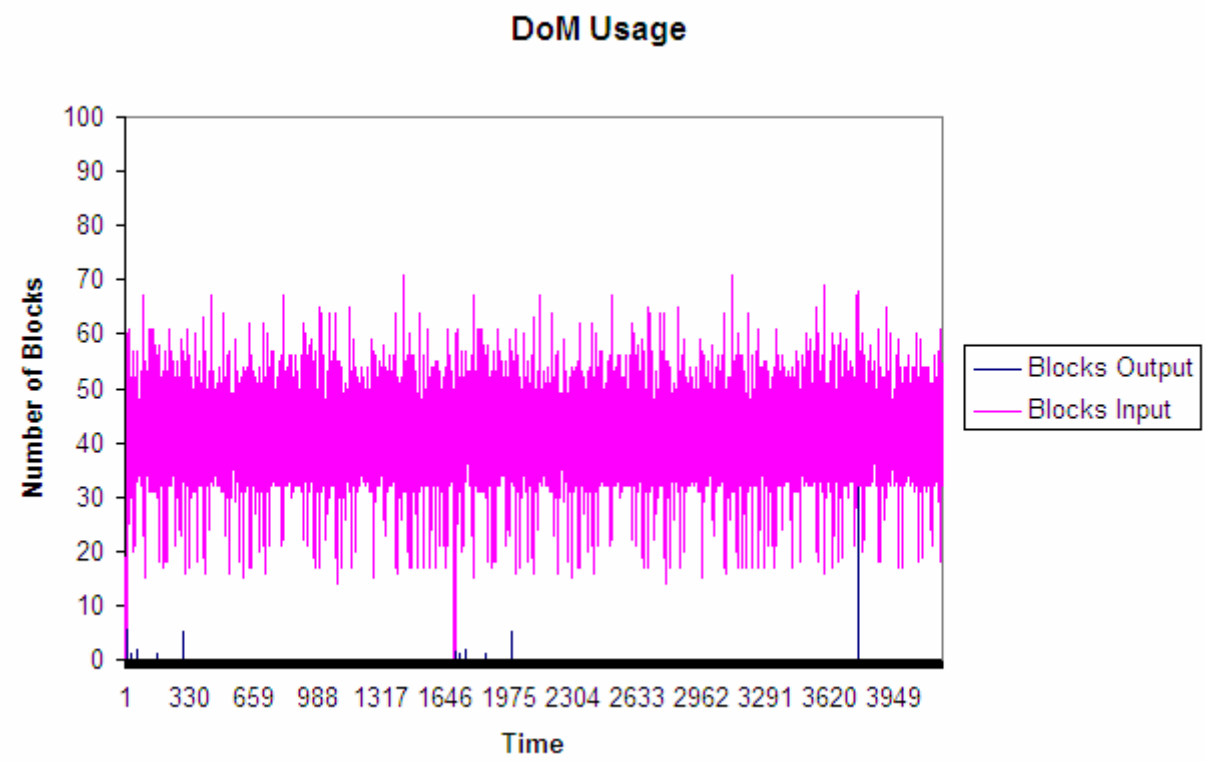
Duration of a call: 15 seconds

Data written to DoM: 55 MB

Average number of blocks written: 41

Average usage per call: 55 MB / 7,000 = 7,8 KB per call

10th set of calls



Number of calls: 10,000

Total number of calls: 90,000

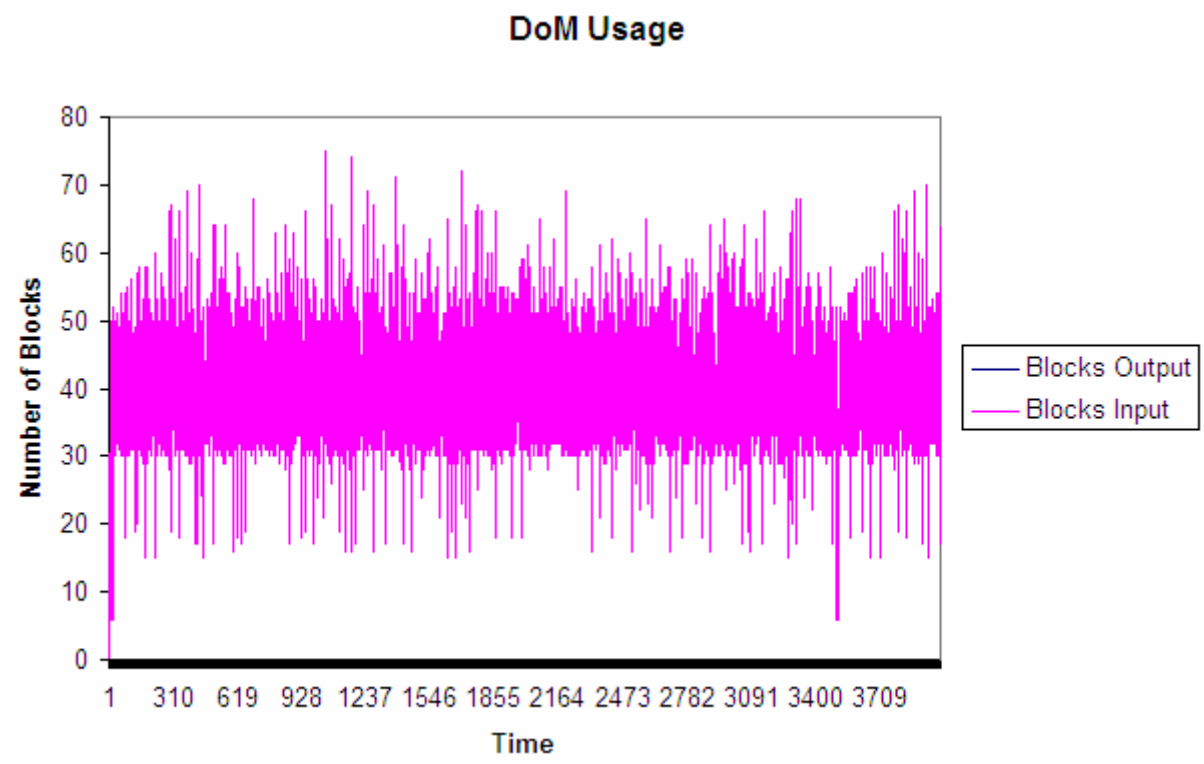
Duration of a call: 10 seconds

Data written to DoM: 76 MB

Average number of blocks written: 40

Average usage per call: 76 MB / 10,000 = 7,7 KB per call

11th set of calls



Number of calls: 10,000

Total number of calls: 100,000

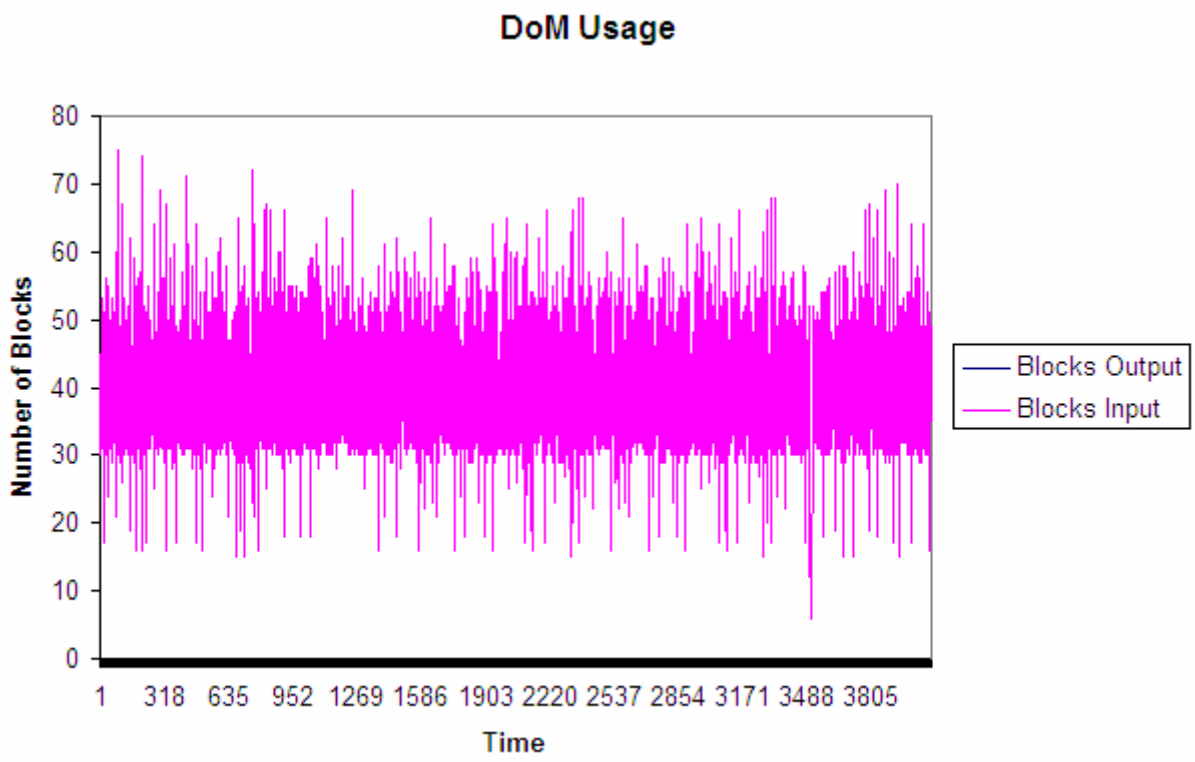
Duration of a call: 10 seconds

Data written to DoM: 76 MB

Average number of blocks written: 39

Average usage per call: 76 MB / 10,000 = 7,7 KB per call

12th set of calls



Number of calls: 10,000

Total number of calls: 110,000

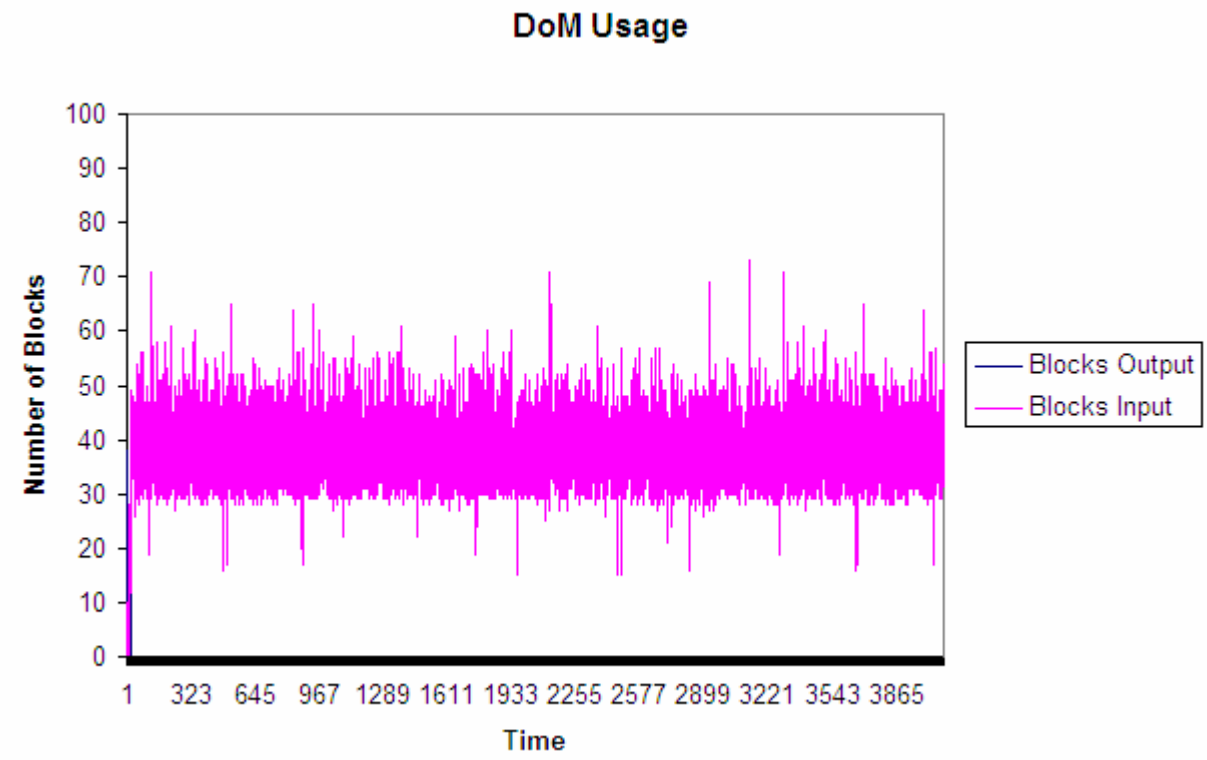
Duration of a call: 10 seconds

Data written to DoM: 76 MB

Average number of blocks written: 40

Average usage per call: 76 MB / 10,000 = 7,7 KB per call

13th set of calls



Number of calls: 10,000

Total number of calls: 120,000

Duration of a call: 10 seconds

Data written to DoM: 76 MB

Average number of blocks written: 38

Average usage per call: 76 MB / 10,000 = 7,7 KB per call

Conclusion

Test 1 showed us how the DoM is handling with every day usage. After 120,000 calls we saw that there wasn't any difference in using the DoM. It was still working as it was in the begining of the test. Average number of blocks written to DoM was 38. One block is 1 KB in size so we know that the average writting speed to DoM was 38 KB/s. Of course that's not its maximum speed, it's just the speed at which it was working. Calls aren't very demanding on disk space.

Also, DoM showed no slowing down during the test and the speed was within 33-41 KB/s range. For more stressful testing we had to do test 2.

Statistics Summary

Total number of calls: 120,000

Average duration of a call: 12,3 seconds

Total data written to DoM: 929 MB

Average number of blocks written: 38

Average disk usage per call: 7,8 KB

Test 2: BadBlocks Test

After test 1, it was obvious that we couldn't appropriately stress the DoM to the point where its blocks become corrupted. For this we used **badblocks** linux utility. This utility is widely used as a disk testing utility and we used it to test the DoM by writting random data on it.

Because **badblocks** needs that the tested partition is unmounted, PBXware was not running, and we couldn't run vmstat to gather additional data because when running badblocks the system shouldn't be doing anything that isn't apsolutely necessary.

We chose last 14960 blocks of the DoM to be written 100,000 times over with random data. This section was between 410000 and 424960 block. Because of such narrow space, wear leveling couldn't do much to prolong the life of blocks more then it is without wear leveling.

So, the test was writting random data on 14960 blocks and it did it 100,000 times over, and that procedure took about 30 hours. When it was finished, the DoM was still working and didn't have corrupt blocks, but when we started second set with next 100,000 writes, its blocks became corrupted.

Manufacturer stated that without wear leveling Disk on Module block has 100,000 read/write cycles, and if we compare that to our test we can conclude that DoMs are really that persistent. Now that we established this as a fact we can use the formula manufacturer provided and calculate the expected lifetime

$$Lifetime = \frac{(DoM\_Size\_KB - OS\_Size\_KB) * 100.000 * 0.95}{38KB/s * 60 * 60 * 24}$$

$$\frac{(512 - 247) * 1024 * 100000 * 0.95}{38 * 60 * 60 * 24} = 7851{,}8\,\text{days} \approx 21\,\text{years}$$

Also it's important to calculate how much data we have put through DoM during two days of testing:

$$14{,}960 blocks * 1KB * 100{,}000 = 1{,}496{,}000{,}000\,KB \approx 1.4TB$$

Conclusion

This test proved that the manufacturers claims should be right. Without wear leveling, blocks have 100,000 read/write cycle, and with theirs claim of 128KB/s of writting constantly it can go on for 6 years. But our calculated usage of 38KB/s gives it 21 year lifespan with 95% probabillity.

Also we must mention that during the testing we put **1,4 Terabytes** of data through the DoM which is very significant for such a device making it perfect for such uses.
It's small, uses 25% energy that normal disks use, makes no noise because there are no moving parts and it's not warming like normal disk drives.

So we can say that the DoMs are very reliable and can work without any problem in such demanding environments like our officeBOX machines.

Project Manager: Senad Jordanovic
Tests and Content: Eldar Avdic