# Are all SSDs Equal?

**Background**

At the MUG Christmas market, RISC OS Bits were demonstrating their NVMe driver software. It was incomplete - the module had not yet been incorporated into the rom - but looked very promising. They were also showing off their 'FAST' SATA drive systems. So how do these systems perform with different drives?

**Technology**

There is a lot of technology here so let's just explain some of it. First of all we have had SATA drives for a while: the ARMX6 used a 2.5" SATA drive under SCSIfs, the Titanium used 2.5" SATA discs under ADFS and the FAST system used 2.5" SATA drives under ADFS.

SATA (Serial Advanced Technology Attachment) uses a serial interface with a native transfer rate of 6Gb/s (600MB/s) but depends on which generation of PCIe connector is used, by drive and computer.

NVMe (Non Volatile Memory Express) drives are faster than SATA drives and use an M.2 connector (formerly known as NGFF (Next Generation Form Factor) using a PCIe 3.0 or higher which may be up to 4 lanes.

M.2 connectors can be used for either a SATA or an NVMe bus interface so the drive type and connector type must match: an M.2 connector can be 'M-key' (supports PCIe x4), 'B-key' (supports PCIe x2) or 'M+B-key' (limited to PCIe x2). 'M-key' is what is used here.

The NVMe interface has been designed to capitalize on the low latency and internal parallelism of solid-state storage devices. Whereas SATA can run up to 6Gb/s, NVMe can run up to 6GB/s (about 7 times faster). However the CM4 only offers PCIe 2x1 which has a 4Gb/s (400MB/s) maximum capacity.

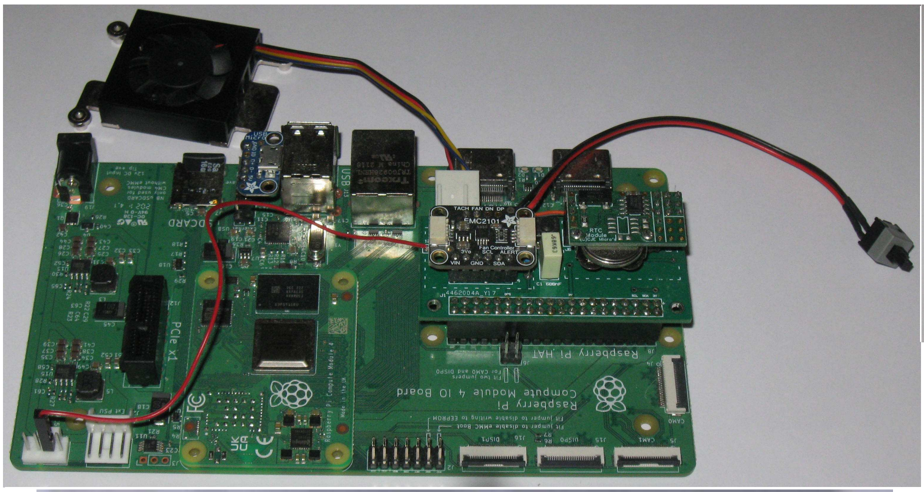

*The USB caddy that holds the NVMe drive - this makes it visible under RISC OS, but only at USB2 speeds. Placed in a PCIe slot, it will run at full speed under Linux.*

There is therefore considerable interest in developing an NVMe driver for RISC OS - note that its random read/write speed has the capacity to be better than SATA. Meanwhile the drives work under Linux and can be 'seen' by RISC OS (under SCSIfs) if placed in a USB caddy.

**Benchmarks**

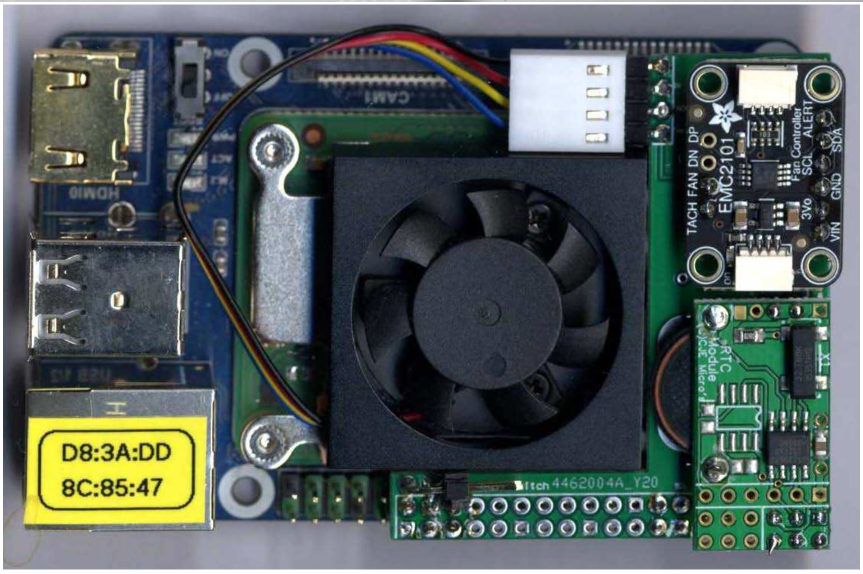
I thought I would do a simple comparison of block loads and saves and byte by byte transfers using RISCOSMark. Now that we have high speed drives (and the RiscPC is long gone), it might be better to quote performance against the fastest FileCore medium (RAMfs).

I have tested several SATA discs, as well as eMMc. I have also tested both SATA and NVMe drives connected over USB (this can make formatting easier




*Top Left:* The Pi Foundation IO board with RTC, fan control and a fan. The reset button doubles as a dual boot switch - press and hold for Linux, press/release for RISC OS.

***Top Right:*** *A Crucial 500GB M.2 2280 NVMe drive mounted on a PCIe adapter board.*

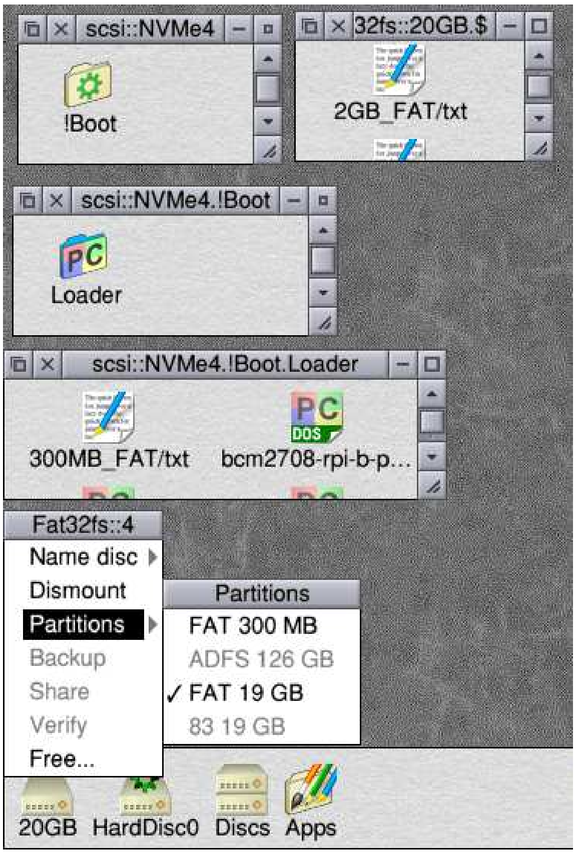



***Left:*** *A Waveshare Mini-B IO board with fan and RTC. A switch fitted between pins 29 and 30 allows it to dual boot into Linux (on the Sabrent 512GB NVMe drive mounted underneath) or into RISC OS (on the eMMc storage). The NVMe drive has 4 partitions: Loader; Filecore; 20GB FAT and ext4 (Linux).*

using HForm and SystemDisc, especially while NVMe drivers are not yet available for RISC OS). HForm can format only ADFS, SCSI and SDFS.

## Test set up

The test set up is either a Waveshare Mini-B IO board with M.2 NVMe drive mounted underneath, a Pi Foundation IO board with an M.2 NVMe drive mounted on a PCIe adapter or a DeskPi Mini Cube, again with an NVMe drive mounted on the board. The NVME drive has a four partition structure with Loader (used by




***Right:*** *the eMMc (HardDisc0) appears as SDFS::0 and the NVMe drive in its USB caddy appears as SCSI::4 and allows RISC OS to access either FAT partition. Note that partition type '83' is 'ext4' or 'Linux' whereas 'AD' is Filecore. RISC OS assumes the offset for the FileCore data rather than by reading the partition table.*

| Drive | Basic model | SD card/eMMc | EEPROM | Firmware | RISC OS | Rev. code | CM4 model # | HD Read | HD Write | FS Read | FS Write | HD Read | HD Write | FS Read | FS Write |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CM4 eMMc 32GB SDFS | | | | | | | | 6% | 2% | 42% | 14% | 23405 | 22546 | 2063 | 720 |
| Crucial BX500 SATA over USB | | | | | | | | 9% | 3% | 9% | 8% | 34133 | 37577 | 427 | 427 |
| Sabrent 512GB NVMe over USB | | | | | | | | 9% | 3% | 3% | 3% | 35929 | 37236 | 134 | 136 |
| Sabrent 512GB NVMe over USB | | | | | Linux | | | 9% | 3% | 135% | 33% | 37000 | 39800 | 6645 | 1672 |
| 4te fat32fs over USB | | | | | | | | 8% | 2% | 89% | 82% | 31507 | 30913 | 4364 | 4231 |
| NVMe Sabrent 512GB | W/s IO Mini-A | 8GB eMMc SDFS::0 | 26-Apr-2022 | 05-Apr-2023 | 5.29 12-Jun-2023 | C03141 | CM4004008 | 102% | 29% | 749% | 181% | 406000 | 395000 | 36900 | 9287 |
| NVMe WD 250GB | W/s IO Mini-B | 32GB card SDFS::4 | 02-Oct-2020 | 30-Aug-2022 | 5.29 04-Oct-2022 * | A03140 | CM4001000 | 53% | 28% | 381% | 92% | 211000 | 383000 | 18800 | 4736 |
| NVMe Kingston 256GB | W/s IO Mini-B | 32GB eMMc SDFS::0 | 11-Jan-2023 | 30-Aug-2022 | 5.29 04-Jan-2024 | D03141 | CM4108032 | 98% | 29% | 637% | 154% | 389000 | 390000 | 31400 | 7903 |
| NVMe Crucial Gen3x4 500GB PCIe | Pi IO | 16GB card SDFS::0 | 26-Apr-2022 | 30-Aug-2022 | 5.29 23-Jul-2023 | B03141 | CM4002000 | 97% | 29% | 576% | 139% | 386000 | 391000 | 28400 | 7140 |
| NVMe Sabrent 512GB | W/s IO Mini-B | 16GB eMMc SDFS::0 | 11-Jan-2023 | 30-Aug-2022 | 5.29 04-Jan-2024 | C03141 | CM4104016 | 96% | 29% | 621% | 150% | 382000 | 392000 | 30600 | 7710 |
| NVMe Integral 512GB | DeskPi Mini | 32GB eMMc SDFS::0 | 11-Jan-2023 | 30-Aug-2022 | 5.29 04-Jan-2024 | C03141 | CM4104032 | 97% | 28% | 593% | 143% | 385000 | 386000 | 29200 | 7359 |
| SanDisk 128GB SSD | Pi IO ADFS::5 | | | | | | | 36% | 12% | 31% | 27% | 141784 | 162217 | 1543 | 1374 |
| SanDisk 240GB SATA | Titanium :4 | 16GB card SDFS::0 | n/a | n/a | 5.24 16-Apr-2018 | n/a | n/a | 31% | 7% | 51% | 49% | 121663 | 91022 | 2524 | 2536 |
| Crucial MX200 250GB SATA | Titanium :5 | 16GB card SDFS::0 | n/a | n/a | 5.24 16-Apr-2018 | n/a | n/a | 30% | 6% | 40% | 48% | 118153 | 87487 | 1965 | 2439 |
| Crucial MX500 250GB SATA | Pi IO | 32GB eMMc SDFS::0 | 26-Apr-2022 | 14-Feb-2023 | 5.29 01-Feb-2023<br>5.29 24-Feb-2023 F | D03141 | CM4108032 | 89% | 24% | 29% | 32% | 353380 | 329286 | 1423 | 1647 |
| V series 240GB SATA | Pi IO ADFS::5 | | | | | | | 90% | 25% | 30% | 31% | 356879 | 337317 | 1489 | 1603 |
| V series 240GB SATA | Pi IO ADFS::5 | | | | Linux | | | 98% | 28% | 453% | 109% | 391000 | 386000 | 22300 | 5609 |
| Crucial BX100 120GB SATA | Pi IO | 32GB eMMc SDFS::0 | 02-Oct-2020 | 14-Feb-2023 | 5.29 31-May-2023<br>5.29 24-Feb-2023 F | A03140 | CM4001032 | 88% | 13% | 35% | 34% | 348768 | 181169 | 1703 | 1721 |
| Crucial BX500 240GB SATA | Pi IO ADFS::5 | | | | | | | 90% | 24% | 34% | 34% | 356879 | 324435 | 1696 | 1720 |
| Crucial MX200 250GB SATA | Pi IO ADFS::5 | | | | | | | 90% | 25% | 64% | 65% | 356879 | 345349 | 3177 | 3346 |
| RAMfs 1500MB | | | | | | | | 100% | 100% | 100% | 100% | 397433 | 1362629 | 4928 | 5134 |

Results from Linux shown thus

* - later ROMS can't read SD card

F - RISC OS Bits FAST rom (dual boots into eMMc-aware RISC OS)

*A comparison of the performance of different SATA drives using the 'FAST' rom on the Pi Foundation IO board with a PCIe to SATA adapter. When connected via a USB caddy the speeds are much reduced. Also shown are the speeds of NVMe drives under Linux. This shows that the limited PCIe Gen 2x1 available has a maximum nominal capacity of 4Gb/s, i.e. about 400MB/s.*

*A theoretical advantage of NVMe should be faster random read/write speeds (e.g. when copying files).*

Linux as the boot drive from which to load the kernel), a FileCore partition of about 110GB for RISC OS, a 20GB FAT partition for sharing files between RISC OS and Linux and an 'ext4' partition using the remaining space on the drive for the root file system for Linux.

Mounting the NVMe drive in a USB caddy allows use of HForm and SystemDisc on RISC OS and the 'dd' command under Linux to format and partition the drive. A small BASIC program then rewrites the MBR to put the partitions into numerical order.

When the NVMe drive is mounted in a USB caddy, the SCSIfs icon bar icon allows either Filecore or FAT partitions to be opened by a mouse click (SELECT or ADJUST) and allows one of several FAT partitions to be selected. The same choice is offered by PartMan - you can choose to mount one of the partitions displayed.

In the absence of a NVMe filing system for RISC OS, I have tested read and write speeds for various NVMe drives using Linux. I have also shown a couple of examples for SATA drives under Linux - this shows the filecore and filing system driver overheads for random access.

```
fio --randrepeat=1 --ioengine=libaio --direct=1 --
gtod_reduce=1 --name=test --
filename=random_read.fio --bs=4k --iodepth=1 --
size=250M --readwrite=randrw ==rwmixread=80

(use --bs=8M --readwrite=read for HD read)
(use --bs=8M --readwrite=write for HD write)
```

*The Linux commands to get measurements of disc speed comparable to ROMark in RISC OS.*

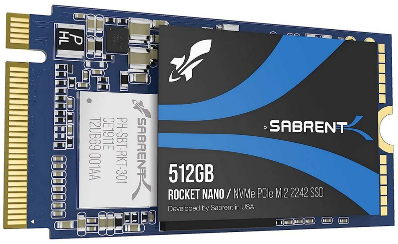
## Historical artefacts

In 1982 I was tinkering with CP/M on a Nascom 2 using floppy discs and delved into the BDOS to add date stamping for files - I had just added a real time clock so I thought it made sense. Not surprisingly disc capacity was measured by number of tracks (initially 35, then 40 then 80), heads (1 or 2) and sectors. Each track was formatted using special codes to mark an ID for the start of each sector and the disk controller would read or write a sector in the space just after the mark.

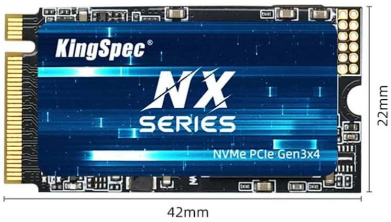
Performance could be improved by formatting the disc to suit the machine: sectors would normally be read or written sequentially and if the next sector just happened to be approaching the head when the controller had been asked to fetch it, you could save 8ms per sector (the


Western Digital 256GB M.2 2230 NVMEe PCIe Gen 3x4
Read/write speeds up to 2400/950 MB/s
£74 Model: PC SN530 NVMe WDC 256GB
Fits all except DeskPi (which only has a 42mm fixing)


Crucial P2 500GB M.2 2280 NVMe PCIe Gen 3
Read/write speeds up to 2400 MB/s
£38.99 Model: CT500P2SSD8
Only fits the PCIe adapter for the Pi Foundation IO board




Sabrent 512GB M.2 2242 NVMe PCIe Gen 3x4
Read/write speeds 1700/1550 MB/s
£59.99 Model: Sabrent SB-1342-512
Fits all.




KingSpec 256GB M.2 2242 NVMe PCIe Gen 3
Read/write speed 3400/1600 MB/s
£34.97 Model: NXM-256 2242
Fits all.


Integral 512GB M.2 2242 NVMe PCIe Gen 3x4
Read/write speed 3300/2700 MB/s
£39.95 Model: INSSD512GM2242G3
Will not fit Waveshare board as it has a thick underside.

*These are the drives that I have tried. Model numbers from output of 'nvme list' on Linux.*

*One important thing to stress is that these are theoretical speeds: SATA quote 600MB/s speeds but the RISC OS filecore system will not quite reach 400MB/s as RAMfs peaks at a read speed of 400MB/s. So although NVMe may quote 1600 or 3200 MB/s theoretical speeds for GEN 3x4, we only have a Gen 2x1 capability on a Compute Module 4 and it surely can't be faster than RAMfs anyway!*

time it takes for a disc at 3600rpm to go through 180°).

Disc addresses still use cylinders (track), head and sector despite the advent of SSD drives. It is possible there is an optimum arrangement for these (or an optimum choice of the LFAU, usually 2048 or 4096) but 16 heads and 63 sectors per track seem common (sector 0 is not usually used). That means an SSD may have over 100,000 heads or cylinders. RISC OS is currently limited to a drive capacity of 250GB per partition.

A parking cylinder is still specified even though there is no physical head to plough into the magnetic surface on a power cut during a disc access (it normally 'flew' over the track aerodynamically).

## Attraction of NVMe

The FAST SATA machines offer 300MB/s read and write speeds based on a Pi Foundation IO board with a PCIe Gen

2 x 1-lane socket and a PCIe to SATA adapter board and are currently the fastest storage medium available to RISC OS, but their footprint is large.

Smaller boards (such as the DeskPi Mini aka PiRO Qube and the Waveshare Mini IO board) offer an M.2 NVMe socket in the approximate footprint of a Pi model 4B. This makes a CM4 based computer with 32GB of eMMc and 256GB of NVMe storage come to just under £200 (depending which CM4 model you choose).

| Item | Cost | Source |
|---|---|---|
| **Items for DeskPi Mini** | | |
| DeskPi Mini | £59.99 | Amazon |
| **Items for Waveshare** | | |
| Waveshare IO Mini-A | £28.99 | Amazon |
| Waveshare 5V fan | £15.99 | Amazon |
| EMC2102 fan control | £5.40 | Pimoroni |
| **Items for both** | | |
| CM4 4G RAM W 16G eMMc | £64.80 | Pimoroni |
| CJE RTC | £10.00 | CJE |
| Sabrent 512G NVMe | £59.99 | Amazon |
| **Total (DeskPi)** | **£194.78** | |
| **Total (Waveshare)** | **£185.17** | |

## Conclusion

There are variations in speed between different SATA drives - some offer RAM cacheing on drive for instance.

NVMe drives are likely to offer RISC OS faster random read/write speeds than SATA drives, based on tests under Linux.

Whether this turns out to be the case or not, NVMe drives will still offer a compact solution.

The bright idea of having a four partition NVMe drive seems to work well. Holding it in a USB caddy allows RISC OS to see both FAT partitions and the filecore partition. With an NVMe driver for RISC OS loaded, it should work without the constraint of the slower USB connection.

**Chris Hall** *chris@svrsig.org*