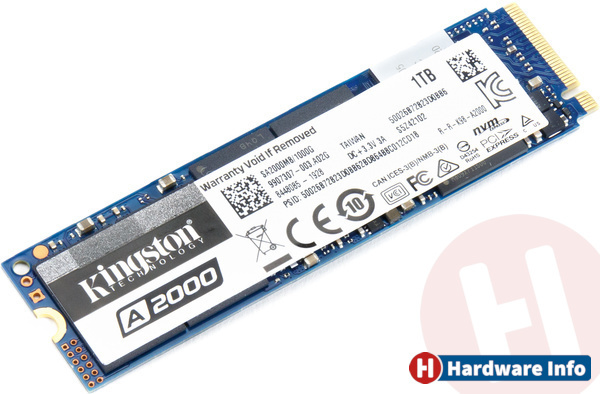
**Hoe werkt een SSD?**

Tomas Hochstenbach 4 juli 2020 09:41[21 reacties](https://nl.hardware.info/artikel/9973/hoe-werkt-een-ssd#reacties)

**Inleiding**

Wat is de grootste verandering in een pc ten opzichte van het vorige decennium? Van de buitenkant bekeken is dat vermoedelijk de opkomst van rgb-verlichting, die anno 2020 welhaast deel uitmaakt van de standaarduitrusting van een krachtig systeem. Tegelijkertijd zijn belangrijke onderdelen als de videokaart en processor weer een klap sneller geworden, geen overbodige luxe met de opkomst van high-res monitoren met een hoge refreshrate. Toch rust het echte antwoord op deze vraag in een ander onderdeel, dat tien jaar geleden nog was voorbehouden aan de vroegste early adopters maar nu in elke pc terug is te vinden is: de *solid state drive*, oftewel ssd.

**

Als je nu een high-end ssd koopt, krijg je een exemplaar met doorvoersnelheden van vele duizenden megabytes per seconde. Tien jaar geleden waren we al tevreden met een tiende daarvan. Toch is het niet de toegenomen snelheid die de ssd-doorbraak heeft geforceerd: zelfs een ‘langzame’ ssd is qua gebruikservaring al een wereld van verschil ten opzichte van een mechanische harde schijf, en voor veel gebruikers zal het onderlinge verschil tussen ssd’s zelfs amper merkbaar zijn. Nee, het feit dat je hem nu in praktisch elke desktop en laptop aantreft heeft de ssd vooral te danken aan enorme prijsdalingen.

  
*OCZ leverde enige tijd stickers mee in de verpakking van hun ssd's om als gebruiker anderen subtiel te kunnen wijzen op de voordelen van een solid state drive.*

Tien jaar geleden was de Intel X25-M Postville de ssd die je wilde hebben: een sata-ssd met leessnelheden tot 250 MB/s (en een schrijfsnelheid van zelfs maar 100 MB/s). Hoogtepunt was daarnaast de ondersteuning voor het trim-commando, waardoor de ssd ook op langere termijn snel bleef. Voor het grootste exemplaar, met een opslagcapaciteit van 160 GB, was je een slordige 400 euro kwijt, oftwel 2,50 euro per gigabyte.

  
*In 2010 was de Intel X25-M ‘Postville’ de ssd die je wilde hebben: 160 GB voor maar 400 euro.*

Sinds de meest recente dalende prijstrend, die zo’n beetje heel 2018 en 2019 besloeg, betaal je voor de goedkoopste ssd’s nog maar een dikke tien cent per gigabyte, terwijl een echt leuk model ongeveer dubbel zo veel kost. In het voorbije decennium is de prijs van een ssd dus met een factor 25 gedaald, tot een niveau waarbij je ongeveer vier keer zo veel betaalt als voor een harddisk. Nu is de harde schijf sinds die tijd ook wel iets goedkoper geworden, maar in 2010 was een ssd in vergelijking daarmee nog ruim veertig keer zo duur.

Een deel van die prijsdaling wordt gevormd door fundamentele economische beginselen – er worden nu simpelweg veel meer ssd’s geproduceerd dan destijds – maar een groter deel hebben we te danken aan technologische innovaties die hun weg naar de ssd, en meer algemeen naar flashgeheugen, hebben gevonden. In dit artikel zetten we van bit tot terabyte uiteen hoe een ssd werkt en laten we verschillende ontwikkelingen rond ssd’s de revue passeren.

**

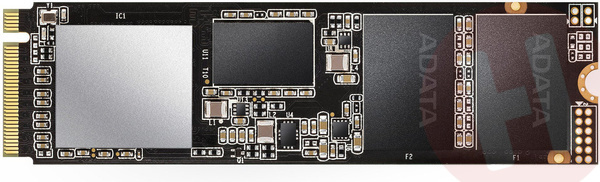
**Hoe werkt een SSD?**

Tomas Hochstenbach 4 juli 2020 09:41[21 reacties](https://nl.hardware.info/artikel/9973/2/hoe-werkt-een-ssd-flashgeheugen-nand#reacties)

**Flashgeheugen (NAND)**

Waar harde schijven bestaan uit snel ronddraaiende magnetische schijven (platters), beweegt er in een ssd niets. Vandaar dus de naam solid state drive. Een doorsnee ssd bestaat uit weinig anders dan een printplaatje met daarop een controller, diverse flashgeheugenchips en soms een extra cache-chip.

Een ssd slaat alle gegevens op in zogenaamd flashgeheugen oftewel nand, genoemd naar de relatief simpele elektronische schakelingen waar het uit bestaat. Flashgeheugen werd al lange tijd gebruikt voor bijvoorbeeld geheugenkaartjes en usb-sticks, waar de chips evidente voordelen qua fysiek formaat en stootvastheid boden boven traditionele harde schijven. Anders dan bij dram is flashgeheugen niet volatiel, wat betekent dat de lading niet verloren gaat als er geen continue stroomtoevoer is. In elke geheugencel, in feite een mosfet-transistor, wordt een bepaalde spanning bewaard. In de basis staat een hoge spanning voor ‘1’ en een lage spanning voor ‘0’. Bij het schrijven wordt de cel, afhankelijk van de huidige staat, ingesteld op 0 of juist gereset naar 1. Om de cel uit te lezen, wordt het spanningsniveau gemeten en geïnterpreteerd als een 0 of een 1. In feite is dit de werking van zogenaamd *single level cell,* oftewel slc-geheugen.

  
*Op een ssd-printplaat vind je doorgaans een controller (links), eventueel een dram-cache (midden) en één of meerdere flashgeheugenchips (rechts, tweemaal).*

**Hoe werkt een SSD?**

Tomas Hochstenbach 4 juli 2020 09:41[21 reacties](https://nl.hardware.info/artikel/9973/3/hoe-werkt-een-ssd-slc-mlc-tlc-en-qlc#reacties)

**SLC, MLC, TLC en QLC**

Eén van de drijvende krachten achter het steeds goedkoper worden van ssd’s is dat er in de loop der tijd technieken zijn ontwikkeld om meer dan één bit in een geheugencel op te slaan. De voordelen spreken voor zich: in dezelfde chip kun je hierdoor veel meer gegevens opslaan. Andersom geredeneerd heb je minder chips nodig om een bepaalde opslagcapaciteit te bereiken. Naast slc zijn mlc, tlc en qlc ontwikkeld, die respectievelijk staan voor multi-, triple- en quad level cell. De gangbare definitie van mlc is twee bits per geheugencel, al neemt met name Samsung daarmee een loopje door van (technisch enigszins correct) ‘3-bit mlc’ of ‘4-bit mlc’ te spreken. Hoe dan ook, de truc is om niet met twee spanningsniveaus te werken, maar met (veel) meer verschillende niveaus.

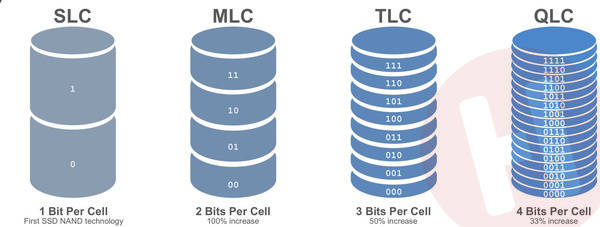
Bij mlc-geheugen is er dus sprake van vier spanningsniveaus, die zich laten vertalen in 00, 01, 10 of 11. Het eerste getal staat steeds voor de waarde van de eerste bit, het tweede getal voor die van de tweede bit. Het aantal niveaus verdubbelt bij elke toegevoegde bit: bij tlc zijn er acht spanningsniveaus, bij qlc zelfs zestien. Daar staat tegenover dat de stap van slc naar mlc nog dubbel zoveel capaciteit opleverde, terwijl dat bij mlc naar tlc nog maar 50% was en het toevoegen van een vierde bit zelfs maar 33% extra ruimte oplevert. Het opslaan van meer bits in dezelfde geheugencel kan echter niet straffeloos. Het verhogen van het aantal spanningsniveaus gaat gepaard met achteruitgang op twee belangrijke vlakken: de snelheid waarmee de geheugencel kan worden uitgelezen en beschreven (dat moet immers met meer precisie), en de levensduur van de cellen en daarmee van het complete flashgeheugen.

Vanuit slc-geheugen kan zeer snel worden gelezen: er hoeft alleen onderscheid te worden gemaakt tussen ‘vol’ of ‘leeg’. Hetzelfde geldt voor schrijfacties. Bij het uitlezen van qlc-geheugen moeten maar liefst zestien spanningsniveaus van elkaar worden onderscheiden. Dat gebeurt in de praktijk relatief inaccuraat, waardoor er geavanceerde foutcorrectie-algoritmes nodig zijn. Het uitvoeren daarvan kost steeds meer tijd en met name daarom is flashgeheugen bestaande uit meer bits per cel een stuk trager. Schrijfacties moeten eveneens met veel meer accuratesse worden uitgevoerd, wat doorgaans zelfs een nog grotere impact op de prestaties heeft dan bij lezen.

**Nadelen**

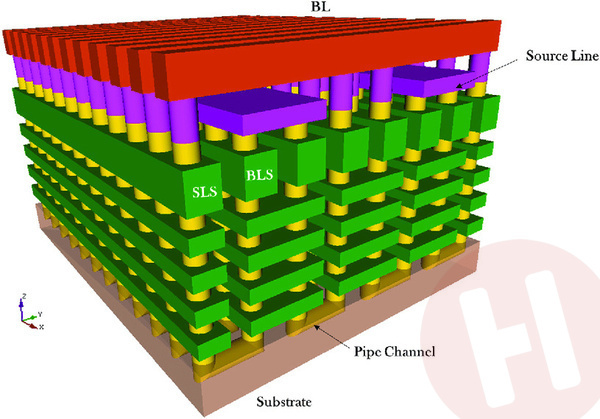
Om met name dit eerste nadeel van mlc-, tlc- en qlc-geheugen op te vangen, maakt vrijwel elke ssd gebruik van een zogenaamde slc-cache. Een deel van het flashgeheugen wordt daarbij aangestuurd als slc, dus in deze buffer wordt slechts één bit per cel opgeslagen. Een schrijfopdracht kan daardoor op ‘slc-snelheid’ worden verwerkt, om op een later moment te worden verplaatst naar het reguliere nand. Deze slc-buffer is tegenwoordig vaak dynamisch van formaat, wat betekent dat hij groter wordt als de ssd niet al te vol zit en er dus veel overtollig flashgeheugen is dat als slc-cache kan worden ingezet. Mede hierdoor kan een ssd echter wel langzamer worden als hij vol raakt.

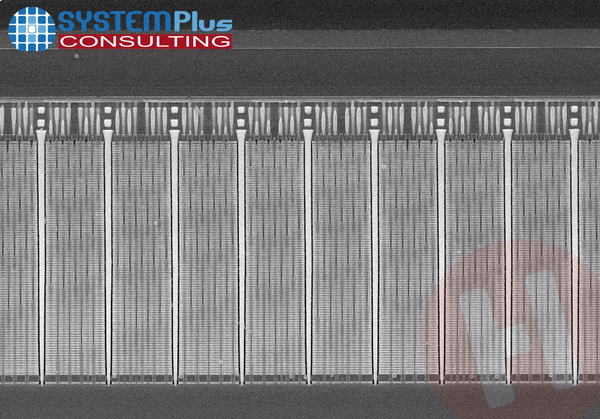
Het tweede nadeel van het verhogen van het aantal bits dat in een geheugencel wordt opgeslagen, is dat de levensduur van het flashgeheugen wordt verkort. Flashgeheugen kan per definitie slechts een bepaald aantal keer worden beschreven, totdat het verschil tussen 0 en 1 door restlading niet meer goed is vast te stellen, waarover later meer. Hoe dan ook, bij nand met meer bits en dus meer spanningsniveaus, komt het moment dat twee naastliggende niveaus niet meer van elkaar kunnen worden onderscheiden steeds eerder. Er is immers een veel kleinere marge dan bij slc-geheugen.

  
*Des te meer bits per cel, des te groter de opslagcapaciteit, maar ook met des te meer spanningsniveaus een ssd moet werken.*

Een andere manier om flashgeheugen goedkoper te maken, is van oudsher het verkleinen van het productieproces. Dat kennen we van veel andere chips, zoals processors: meer transistors op dezelfde oppervlakte leidt tot lagere kosten, hogere prestaties en een toenemende efficiëntie. Bij de moeizame overstap naar productieprocessen van ordegrootte 20nm kwamen de geheugenfabrikanten echter tot de conclusie dat verdere verkleining een te grote impact zou hebben op de levensduur. Met het fysiek kleiner worden van geheugencellen neemt de hoeveelheid restlading waartegen ze bestand zijn immers in rap tempo af.

Sindsdien worden geheugencellen daarom niet of nauwelijks nog kleiner gemaakt om de dichtheid te verhogen, maar worden er meerdere lagen geheugencellen bovenop elkaar geplaatst. Dat vereiste significante investeringen in productiemethoden, en soms ook in gebruikte materialen, waarover we bij de [**introductie van 3d-nand uitgebreid schreven**](https://nl.hardware.info/artikel/5765/3d-v-nand-de-hoogte-in-met-flash-opslag). Het leverde echter wel weer voldoende schalingsmogelijkheden op om in de voorzienbare toekomst steeds meer geheugencellen op dezelfde oppervlakte kwijt te kunnen.

  
*Bij 3d-nand worden er meerdere lagen geheugencellen op elkaar geplaatst.*

  
*Een röntgenopname van een courante Samsung 3d-nand-chip met 92 lagen.*

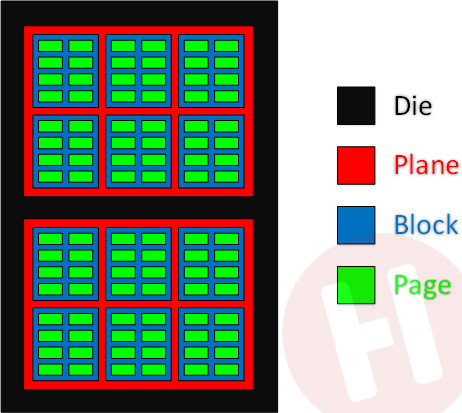
**Hoe werkt een SSD?**

Tomas Hochstenbach 4 juli 2020 09:41[21 reacties](https://nl.hardware.info/artikel/9973/4/hoe-werkt-een-ssd-de-achilleshiel-pages-en-blocks#reacties)

**De achilleshiel: pages en blocks**

Op transistorniveau wordt een geheugencel geprogrammeerd door middel van twee lijnen waar spanning op kan worden gezet, de bitline en de wordline. Door spanning te zetten op de wordline springen elektronen over naar de *gate* of isolator. Door vervolgens juist spanning te zetten op de bitline kan een cel weer worden gereset. Bij moderne flashchips zijn de cellen gegroepeerd in *pages* van 4 kilobyte of een veelvoud daarvan, die mooi uitlijnen met de clustergrootte waarmee courante bestandssystemen werken. Een page is de kleinste hoeveelheid data die kan worden uitgelezen of beschreven. Zelfs voor de kleinste aanpassing moet de page compleet worden uitgelezen en opnieuw geschreven.

Dé achilleshiel van flashgeheugen is echter dat er weliswaar per page kan worden gelezen en geschreven, maar niet kan worden *herschreven*. Dat kan uitsluitend per *block*, een groep van een bepaald aantal pages. Bij 2d-nand waren dat doorgaans 128 pages, bij 3d-nand soms het dubbele tot wel tienvoudige daarvan. Een bepaald aantal blokken bij elkaar vormt een complete flashchip die een *plane* wordt genoemd. Binnen één fysieke chip worden doorgaans meerdere (vaak twee, vier of acht) planes gecombineerd. Die fysieke chip is de geheugenpackage die je op de printplaat van een ssd ziet zitten en waarvan er door het steeds verder verhogen van het aantal pages, blocks en planes steeds minder nodig zullen zijn om een bepaalde capaciteit te bieden. Ook het opslaan van steeds meer bits per cel draagt daar vanzelfsprekend aan bij.

  
*Een versimpelde weergave van hoe pages, blocks en planes zich tot elkaar verhouden binnen één die (chip).*

Het karakteristieke dat nand alleen per block kan worden herschreven, heeft in theorie grote invloed op de prestaties. Ga maar na: een lege ssd is uit de doos lekker snel omdat hij voor elke schrijfopdracht een frisse, nieuwe page kan aanspreken. Is er eenmaal zo veel geschreven dat de capaciteit van de drive is bereikt, dan moet de ssd voor elke schrijfactie een flinke onderneming uitvoeren. Hij moet op zoek naar een block met (inmiddels weer) lege pages, de inhoud daarvan kopiëren naar de cache, het volledige block wissen en tot slot de combinatie van de oude en nieuwe pages naar dit block schrijven.

Belangrijke crux is dat gegevens die je in het besturingssysteem wist, van oudsher niet direct fysiek van een harde schijf werden weggehaald. Daar was simpelweg geen noodzaak toe: een mechanische harddisk herschrijft immers net zo snel als hij een leeg oppervlak beschrijft. Zodra elk block eenmaal was beschreven, moest de ssd dus voor elke schrijfopdracht bezig met de zojuist omschreven tetris-oefening, waardoor de schrijfprestaties dramatisch inzakten. Bovendien werd de prestatie-impact alleen maar groter terwijl er steeds meer van de steeds grotere pages in een block werden gestopt.

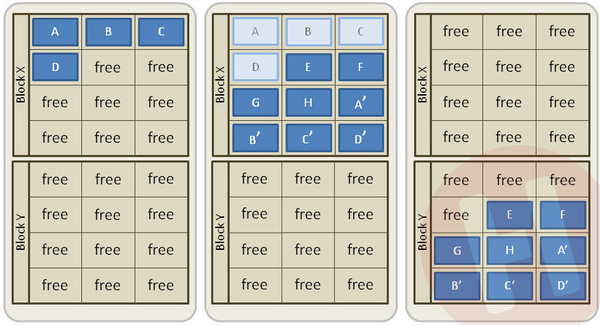
**Hoe werkt een SSD?**

Tomas Hochstenbach 4 juli 2020 09:41[21 reacties](https://nl.hardware.info/artikel/9973/5/hoe-werkt-een-ssd-controller-de-ssd-trukendoos#reacties)

**Controller: de SSD-trukendoos**

Het flashgeheugen dat aan de basis staat van ssd’s heeft dus gunstige karakteristieken, zoals het feit dat de gegevens bewaard blijven zonder continue spanning, en de veel lagere toegangstijd ten opzichte van ronddraaiende platter, maar ook een aantal nukken. De belangrijkste taak van de ssd-controller, letterlijk de processor die de ssd aanstuurt, is dan ook om zo slim mogelijk om deze minder gunstige eigenschappen heen te werken. De trukendoos van een ssd-controller heeft ruwweg twee doelen: de levensduur van het flashgeheugen verlengen en de snelheden ervan verhogen.

De meest voor de hand liggende truc die specifiek voor ssd’s wordt toegepast, lost het herschrijfprobleem waar we het zojuist over hadden goeddeels op. Het trim-commando, dat in de scsi-standaard de meer beschrijvende naam ‘unmap’ heeft, laat de ssd expliciet weten wanneer een bestand in het OS wordt verwijderd, zodat de betreffende pages ook daadwerkelijk kunnen worden gewist. Ondersteuning hiervoor zit sinds Windows 7 in het besturingssysteem van Microsoft, en sinds 2011 ook in macOS. Nu is het trim-commando slechts een deel van de oplossing. Door het fysiek wissen van verwijderde gegevens ontstaan weliswaar gaten, maar die blijven doorgaans verspreid over de hele ssd. Wil de ssd iets met zo’n ‘echt lege’ page, dan moet er dus alsnog een heel block worden herschreven. Aanvullend wordt daarom een techniek toegepast die *garbage collection* wordt genoemd: het zo efficiënt mogelijk verzamelen van gegevens die nog courant zijn, door blocks opnieuw in te delen als er zich ‘gewiste data’ in een block bevindt. Aan het einde van die herindeling blijven volledig lege blocks over, die dus op volle snelheid kunnen worden beschreven als er nieuwe schrijfopdrachten binnenkomen.

  
*Links worden vier pages geschreven naar een block. In het midden worden die vier pages bewerkt (en dus herschreven) en worden nog eens vier nieuwe pages geschreven. Rechts zie je dat de eerste vier pages, die nu niet meer nodig zijn, pas kunnen worden vrijgemaakt als alle pages die bewaard moeten blijven zijn verplaatst naar een ander block.*

**Afvalstoffenheffing: de levensduur**

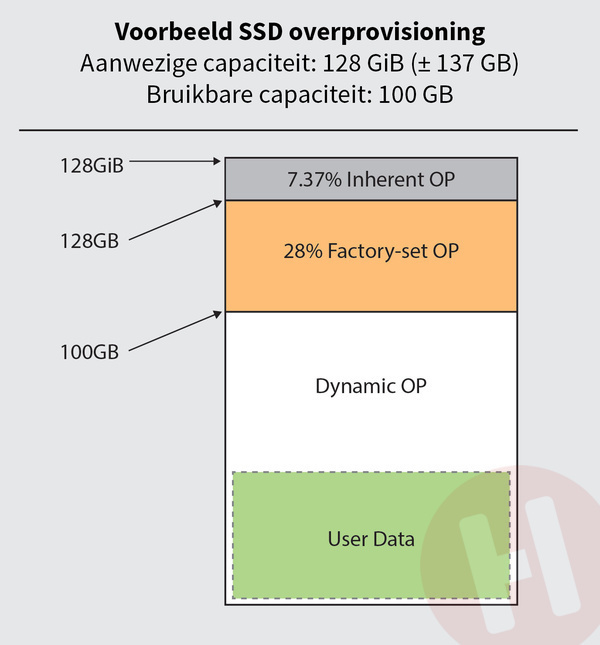
De vuilnisophaaldienst van de ssd klinkt als de beste uitvinding sinds het gesneden brood, maar ook deze techniek is niet zonder nadelen. Voor de snelheid zou het ideaal zijn als de ssd op elk rustmoment aan de slag ging met deze herindelingen, maar dat conflicteert met een andere minder fijne karakteristiek van flashgeheugen die we eerder beschreven, namelijk dat het slechts een eindig aantal keren kan worden beschreven. Bij dit fenomeen spreken we over het aantal *program/erase cycles*, oftewel het aantal keren dat een geheugencel kan worden geprogrammeerd (naar 0) en gereset (naar 1). Toen slc nog de standaard was en er slechts één bit per cel werd opgeslagen, waardoor er slechts twee spanningsniveaus hoeven te kunnen worden onderscheiden om de inhoud van een cel te bepalen, kon een cel makkelijk honderdduizend keer opnieuw worden beschreven. Dat was ruim voldoende om in de praktijk geen problemen te veroorzaken.

In theorie neemt het aantal p/e-cycles dat een geheugencel aankan, echter met een drastische tien keer per extra bit af. Volgens die logica kan een mlc-cel nog tienduizend keer mee, een tlc-cel slechts duizend, en houdt een qlc-cel ermee op na honderd keer te zijn beschreven. In de praktijk valt dat door verbeteringen bij zowel productie als aansturing gelukkig mee: moderne tlc-chips kunnen zo’n drieduizend p/e-cycles doorstaan, een qlc-chip ongeveer duizend. Zelfs dat is mogelijkerwijs nog aan de pessimistische kant. Toen we in 2013 de proef op de som namen met een van Samsungs eerste tlc-ssd’s, duurde het immers al drieduizend cycles voordat er onherstelbare fouten optraden.

Naast het voorkomen van onnodige writes is het tevens van belang dat de geheugencellen gelijkmatig verslijten. Een ssd-controller zal nieuwe schrijfopdrachten daarom bij voorkeur onderbrengen in delen van het flashgeheugen die nog relatief weinig zijn gebruikt, een techniek die *wear leveling* wordt genoemd. Zo wordt de levensduur van de ssd als geheel gemaximaliseerd. Het zou immers zonde zijn als een ssd onbruikbaar wordt door enkele kapotte geheugencellen terwijl andere cellen nog maar op een kwart van hun levensduur zitten.

Al deze zaken – wear leveling, garbage collection en het gegeven dat data uitsluitend per block kunnen worden gewist – zorgen ervoor dat een ssd meer schrijfacties uitvoert dan je besturingssysteem ernaartoe stuurt. Het verschil daartussen wordt gevangen in de WAF, de *write amplification factor*. Bij typisch gebruik ligt die factor rond de 3. Hier ontstaat de interessante paradox waarbinnen een ssd-controller opereert: meer schrijven vermindert in de basis de levensduur van een ssd, maar gelijkmatig schrijven en slim herindelen – waarvoor schrijfacties nodig zijn – verlengen die levensduur juist.

Zelfs als de levensduur van flashcellen normaal verdeeld is, zullen er uitschieters en maandagmorgenmodellen tussen de miljoenen geheugencellen in de gemiddelde ssd zitten. Om vroege defecten op te vangen, wordt flashgeheugen standaard met reservecellen uitgerust. Afhankelijk van het exacte type geheugen en de fabrikant gaat het om enkele procenten tot rond de tien procent extra bits per page of block. Als een geheugencel om welke reden dan ook niet meer bruikbaar is, kunnen de back-upcellen als plaatsvervanger dienen. Het plaatsen van meer geheugencellen dan strikt noodzakelijk, in jargon *overprovisioning*, wordt ook op een hoger niveau toegepast. Veel ssd’s worden bijvoorbeeld verkocht met een capaciteit van 480 of 500 GB, terwijl er fysiek 512 GB flashgeheugen aanwezig is. Strikt genomen gaat het bij dat laatste getal zelfs om GiB, oftewel veelvouden van 1024, waardoor er per definitie meer capaciteit aanwezig is dan er aan het OS wordt gepresenteerd. Bij zakelijke ssd’s wordt vaak zelfs nog veel meer geheugen achtergehouden.

  
*Een voorbeeld van een ssd waarbij additionele overprovisioning wordt toegepast.*

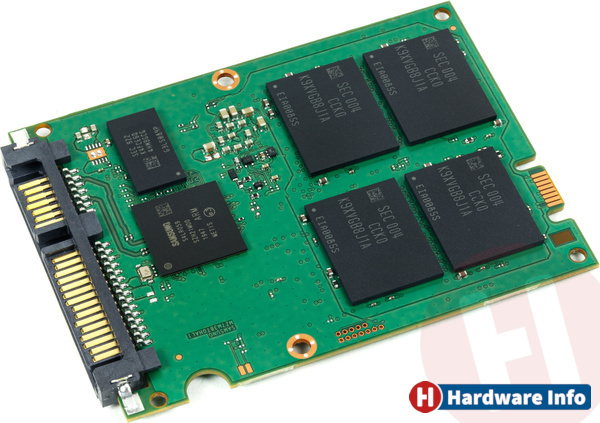
Overprovisioning biedt als het ware ademruimte aan de ssd-controller voor het uitvoeren van alle besproken trucs. Zelfs als de ssd vol is kan er bijvoorbeeld naar lege pagina’s in het achtergehouden flash worden geschreven, of naar het niet-dynamische deel van de slc-cache. Daarnaast kunnen defecten bij de reguliere flashchips worden opgevangen, een proces dat we *reallocation* noemen. Daarom is de toename van het aantal *reallocated sectors*, zoals je die in de SMART-gegevens kunt uitlezen via software als CrystalDiskInfo, doorgaans een indicatie dat je ssd richting het einde van zijn levensduur gaat. Je kunt overigens relatief simpel zelf (extra) overprovisioning toepassen, door slechts een deel van je ssd te partitioneren.

**Hoe werkt een SSD?**

Tomas Hochstenbach 4 juli 2020 09:41[21 reacties](https://nl.hardware.info/artikel/9973/6/hoe-werkt-een-ssd-dram-cache#reacties)

**DRAM-cache**

Bij duurdere ssd’s wordt de controller bijgestaan door een dram-cache. Dit geheugen is veel sneller dan nand, maar raakt net als regulier intern geheugen wel zijn gegevens kwijt zodra de stroom eraf gaat. Als de ssd bezig is kan de dram-cache, die doorgaans ongeveer een gigabyte per terabyte opslagcapaciteit omvat, kleine schrijfopdrachten opvangen en zo als buffer dienen totdat de gegevens naar het nand kunnen worden geschreven. Een ander doel van de dram-cache is het snel kunnen benaderen van de *translation table*. Door alle trucs die een ssd-controller toepast is er doorgaans een groot verschil tussen waar het bestandssysteem van het OS denkt dat gegevens zijn opgeslagen, en waar ze werkelijk fysiek zijn opgeslagen. In deze ‘plattegrond’ houdt de controller bij welke data waar staan.

**

Bij veel goedkope ssd’s ontbreekt een dram-cache, wat uiteraard tot mindere prestaties leidt. Het continu raadplegen van de translation table in het flashgeheugen zelf zou echter desastreuze gevolgen hebben voor de prestaties – die is immers nodig voor letterlijk elke lees- en schrijfoperatie die een ssd uitvoert. Sommige voor deze ‘dram-less’ ssd’s bedoelde controllers hebben daarom zelf een kleine hoeveelheid geheugen ingebouwd, waarop uitsluitend deze translation table past. Sinds versie 1.2 van het nvme-protocol, dat voor de meeste moderne ssd’s wordt gebruikt, is er echter ook een alternatieve oplossing beschikbaar: *host memory buffer*. Hierbij wordt het reguliere werkgeheugen van het systeem gebruikt als buffer voor de ssd. Dat is qua latency vanzelfsprekend veel minder ideaal dan onboard dram, maar het is te prefereren boven het bewaren van deze tabel in het flashgeheugen zelf.

Het grote nadeel van dram ten opzichte van flashgeheugen is dat het volatiel is. Het wordt dus geleegd zodra de stroom uitvalt of zodra je pc om wat voor reden dan ook wordt uitgeschakeld. Sommige fabrikanten rustten hun ssd’s daarom uit met condensators die voldoende stroom opsloegen om de buffer te kunnen wegschrijven bij een *power loss*. Anno 2020 komen we dit type *power loss protection* vrijwel alleen nog tegen in enterprise-ssd’s. Voor consumenten is de impact van hooguit enkele seconden dataverlies doorgaans te verwaarlozen. Wel zijn er maatregelen genomen in het ontwerp van flashgeheugen, waardoor actieve bewerkingen bij stroomuitval niet tot datacorruptie leiden.

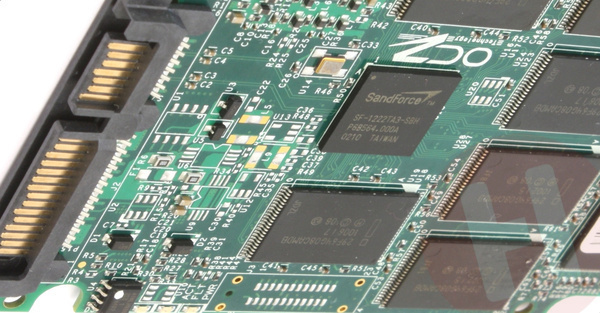
**Hoe werkt een SSD?**

Tomas Hochstenbach 4 juli 2020 09:41[21 reacties](https://nl.hardware.info/artikel/9973/7/hoe-werkt-een-ssd-encryptie-en-compressie#reacties)

**Encryptie en compressie**

Een van de laatste functies die een ssd-controller kan bieden, is hardwarematige versnelling voor on-the-fly encryptie van gegevens. Dat gaat vlotter dan software-encryptie, vergelijkbaar met hoe het decoderen van video hardwarematig kan worden versneld. Deze hardware-encryptie bleek echter niet altijd foutloos. Sinds kort moet je in Windows dan ook expliciet aangeven dat je deze functionaliteit wil gebruiken, anders gebruikt het OS van Microsoft softwarematige encryptie, zelfs als je ssd wel een hardwarematige implementatie heeft.

Tot slot kunnen we de mogelijkheid van compressie niet negeren. Controllerfabrikant SandForce, die tien jaar geleden een grote rol speelde in ssd-land en via diverse overnames bij Seagate terechtkwam, pionierde deze techniek om zowel de snelheid als betrouwbaarheid van een ssd te verhogen. Dankzij compressie en deduplicatie wordt er immers simpelweg minder geschreven. De belangrijkste consequenties zijn dat een ssd trager zal zijn als er met data wordt gewerkt die niet verder comprimeerbaar zijn en dat de prestaties daardoor minder consistent zijn. Helaas is deze techniek behoorlijk goed dichtgetimmerd met patenten, waardoor we hem in de consumentenwereld nog nauwelijks tegenkomen.

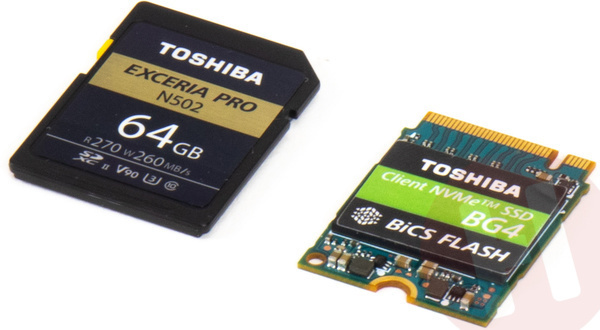
  
*Sandforce-controllers staan bekend om hun uitgebreide compressie-mogelijkheden.*

**Hoe werkt een SSD?**

Tomas Hochstenbach 4 juli 2020 09:41[21 reacties](https://nl.hardware.info/artikel/9973/8/hoe-werkt-een-ssd-de-toekomst#reacties)

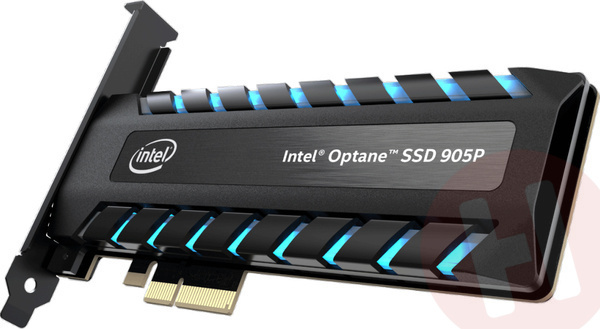
**De toekomst**

De ssd die je anno 2020 koopt, ziet er in veel opzichten anders uit dan het exemplaar waar je tien jaar geleden reuze trots op mocht zijn. Het complete ecosysteem van opslag, dat was bedacht op de mechanische harddisk, heeft zich aangepast aan de ssd: de formfactor (van 2,5 of 3,5” naar m.2), de interface (van sata naar pci-express) en het protocol (van ahci naar nvme). De controller die het geheel aanstuurt is veel slimmer en sneller geworden. Ook het flashgeheugen steekt door de komst van 3d-nand significant anders in elkaar.

  
*De Toshiba BG4, een opvallende m.2-ssd die dankzij zijn formaat (m.2 2230) vrijwel even groot is als een sd-kaart.*

Op de korte termijn is het groeipad van de ssd-techniek daarom vrij concreet: er worden steeds meer lagen gebruikt. 96 lagen begint nu langzaam de standaard te worden, 128 lagen komt eraan en volgens de roadmaps van diverse flashfabrikanten naderen we tegen 2025 de 500 lagen. Tegelijkertijd zijn de fundamentele nukken van flashgeheugen nog altijd dezelfde, met beperkingen op de gebieden van snelheid en betrouwbaarheid. Met een arsenaal aan slimme trucs leveren ssd-controllers daar strijd tegen. De nieuwe uitdagingen die uit de doorontwikkeling van 3d-nand voortkomen zullen vereisen dat de controllers steeds geavanceerdere algoritmes gebruiken om ze te compenseren.

De voorzienbare toekomst van nand-flashgeheugen mag dankzij 3d-nand dan wel zijn veiliggesteld, op de langere termijn hebben we een opvolger nodig die het liefst afrekent met de aloude nadelen van flash. Het door Intel en Micron ontwikkelde 3D XPoint leek een belangrijke kandidaat, maar is en blijft vooralsnog zo duur dat het alleen in de grootzakelijke wereld een rol speelt. Intels pogingen om het met kleine cachingmodules in consumentenhanden terecht te laten komen, zijn weinig succesvol gebleken. Naar technisch geavanceerde alternatieven als reram, mram en pcm is veel onderzoek gedaan, maar tot economisch haalbare massaproductie is het nog niet gekomen.

  
*3D XPoint probeert de beste karakteristieken van dram en nand te combineren, maar ssd’s die er gebruik van maken zijn vooralsnog peperduur.*

Voorlopig is dat voor de consument ook niet zo relevant. Waar de harde schijf zich in de loop der tijd tot een duidelijke flessenhals heeft ontwikkeld, zal de gemiddelde consument al moeite hebben om een sata- en nvme-ssd uit elkaar te houden, laat staan dat er behoefte is aan nog betere prestaties. Opslag is nu zo snel dat de bottleneck weer bij andere componenten in de pc is komen te liggen, of misschien wel bij de gebruiker zelf. Tegelijkertijd is de pc-ssd lang niet meer de grootste bron van vraag naar flashgeheugen. Dat is tegenwoordig de smartphone. Toch hoeft dat geen obstakel te zijn voor verdere innovatie: ook bij opslag voor mobiele apparaten is er vraag naar meer capaciteit, snelheid en energiezuinigheid. Hooguit liggen de accenten iets anders dan bij ssd’s voor laptops en desktops.

# Solid state drive

[](https://commons.wikimedia.org/wiki/File:2008_Intel_Developer_Forum_Taiwan_Day2_Showcase_Intel_SSD_Prototype.jpg)Een prototype van een SSD[](https://commons.wikimedia.org/wiki/File:14-06-11-ssd-RalfR-N3S_7886-03.jpg)mSATA-SSD, met ernaast een mSATA-USB-adapter

Een **solid state drive** (**SSD**) is een medium waarop [digitaal](https://nl.wikipedia.org/wiki/Digitaal) gegevens bewaard kunnen worden met behulp van [niet-vluchtig](https://nl.wikipedia.org/wiki/Niet-vluchtig_geheugen) (zoals [flash](https://nl.wikipedia.org/wiki/Flashgeheugen)) of [vluchtig](https://nl.wikipedia.org/wiki/Vluchtig_geheugen) [geheugen](https://nl.wikipedia.org/wiki/Computergeheugen) (bijvoorbeeld [SDRAM](https://nl.wikipedia.org/wiki/SDRAM)). SSD's worden voornamelijk gebruikt in [computertoepassingen](https://nl.wikipedia.org/wiki/Computer) waar traditioneel een [harde schijf](https://nl.wikipedia.org/wiki/Harde_schijf) gebruikt werd. SSD's staan bekend om hun korte zoek- en toegangstijd. Tegenwoordig hebben SSD's een toegangstijd van amper 0,1 [milliseconde](https://nl.wikipedia.org/wiki/Milliseconde).

## Snelheid

Kenmerkend aan solid state drives is dat er geen bewegende onderdelen gebruikt worden die wel in harde schijven te vinden zijn, zoals een roterende schijf of bewegende [lees- en schrijfkoppen](https://nl.wikipedia.org/wiki/Magneetkop). Hierdoor treden er (weinig tot) geen mechanische fouten meer op en behoort het wachten op de schijf en het positioneren van de koppen tot het verleden. Gegevens kunnen – onafhankelijk van waar ze zijn opgeslagen – altijd even snel gevonden worden. [Defragmenteren](https://nl.wikipedia.org/wiki/Defragmentatie) is niet meer nodig; bestanden die uit duizenden fragmenten bestaan worden nagenoeg net zo snel gelezen als een aaneengesloten bestand. Ook het wegschrijven van gegevens gaat veel sneller bij een SSD ten opzichte van een harde schijf. Er zitten vaak meerdere geheugenmodules in een SSD, die onafhankelijk van elkaar data kunnen wegschrijven.

Een SSD kan worden aangesloten via een [SATA-aansluiting](https://nl.wikipedia.org/wiki/Serial_ATA) op een [moederbord](https://nl.wikipedia.org/wiki/Moederbord). Er zijn ook SSD's die via [PCI](https://nl.wikipedia.org/wiki/Peripheral_Component_Interconnect) en [PCI Express](https://nl.wikipedia.org/wiki/PCI_Express) worden aangesloten op het moederbord. Deze kunnen nog hogere snelheden halen, maar zijn vaak ook duurder dan de via SATA aangesloten exemplaren.

Vanwege de hoge snelheden worden SSD's vaak als opstartschijf gebruikt, hierbij zet men het [besturingssysteem](https://nl.wikipedia.org/wiki/Besturingssysteem) op de SSD. Het voordeel hiervan is dat het besturingssysteem veel sneller opstart. Ook zware programma's die veel data nodig hebben om goed te werken, worden vaak op SSD's gezet. Een [pc](https://nl.wikipedia.org/wiki/Personal_computer) kan veel winst halen uit een SSD omdat de harde schijf meestal een bottleneck vormt voor een computer met recente [hardware](https://nl.wikipedia.org/wiki/Hardware). Dit komt doordat de standaard draaisnelheid van harde schijven al jarenlang 7.200 [RPM](https://nl.wikipedia.org/wiki/Omwentelingen_per_minuut) is. Er bestaan ook schijven van 10.000 of 15.000 RPM, maar ook deze zijn trager dan SSD's. Het is moeilijk om harde schijven nog sneller te maken door de rotatiesnelheid te verhogen, aangezien bij nog hogere draaisnelheden van de interne schijf er te grote krachten optreden die storingen kunnen opwekken. Het is wel mogelijk harde schijven op andere manieren sneller te maken, bijvoorbeeld door de datadichtheid (hoeveelheid gegevens per vierkante centimeter) te verhogen.

## SSD intern

[](https://commons.wikimedia.org/wiki/File:Embedded_World_2014_SSD.jpg)Binnenkant van een industriële SSD. De rijen condensatoren aan de rechterkant voorkomen dataverlies bij stroomuitval

Een solid state drive wordt gebruikt voor opslag van gegevens. Daarvoor is geheugen nodig. Bij SSD's heeft men hiervoor de keuze uit twee soorten geheugen: [DRAM](https://nl.wikipedia.org/wiki/Dynamic_random-access_memory) en [flashgeheugen](https://nl.wikipedia.org/wiki/Flashgeheugen).

### DRAM

DRAM is hetzelfde soort geheugen als het geheugen dat als werkgeheugen ([RAM](https://nl.wikipedia.org/wiki/Random-access_memory)) in een computer gebruikt wordt. Een op DRAM gebaseerde SSD is daardoor ook relatief eenvoudig te [upgraden](https://nl.wikipedia.org/wiki/Upgrade), namelijk door er [modules](https://nl.wikipedia.org/wiki/Module) in te steken die een grotere capaciteit hebben.

Het nadeel van DRAM-SSD's is dat ze ofwel een batterij ofwel een aparte stroomtoevoer nodig hebben, aangezien het geheugen vluchtig is. Zonder deze voorziening zou een DRAM-SSD bij het uitvallen van de stroom alle gegevens verliezen.

### Flashgeheugen

SSD's met flashgeheugen zijn doorgaans trager dan SSD's met DRAM, maar ze zijn wel goedkoper.

Een SSD met flashgeheugen is opgebouwd uit:

* het flashgeheugen zelf: dit is voor de opslag van gegevens;
* een geheugencontroller: deze zorgt ervoor dat de SSD gegevens kan wegschrijven in of verwijderen van het flashgeheugen;
* [cache](https://nl.wikipedia.org/wiki/Cache_(tijdelijk_geheugen)): een SSD gebruikt hiervoor een kleine hoeveelheid DRAM. Dit verhoogt de prestaties van de SSD: gegevens kunnen gebufferd worden vóór ze door de geheugencontroller worden weggeschreven naar het flashgeheugen;
* een batterij: deze zorgt ervoor dat de SSD de gegevens die nog in de cache staan niet verliest als de computer per ongeluk uitvalt door een elektriciteitsstoring. De batterij zorgt er dan voor dat de data ofwel wordt weggeschreven ofwel wordt bewaard totdat er weer stroom is.

## MLC versus SLC

Solid state drives zijn onder te verdelen in twee typen: SLC en MLC. Een SSD bestaat uit verschillende cellen. Iedere cel heeft een analoge waarde. Deze analoge waarde, doorgaans een lading, spanning of weerstand, wordt onderverdeeld om tot een digitale waarde te komen.

Bij Single-Level-Cell SSD's (SLC) wordt de analoge waarde van een cel verdeeld in twee bereiken: een hoog bereik en een laag bereik. Hierdoor slaat iedere cel effectief één bit op (0 of 1).

Bij Multi-Level-Cell SSD's (MLC) wordt de analoge waarde in meer bereiken verdeeld. Hierdoor worden effectief meer bits per cel opgeslagen. Met vier analoge waarden zijn dat twee bits per cel (00, 01, 10 of 11), met acht waarden zijn dat er drie (000, 001, 010, 011, 100, 101, 110 of 111) enzovoorts.

Dit verschil heeft tot gevolg dat SLC's betrouwbaarder, duurzamer en sneller zijn, terwijl MLC's juist als voordeel hebben dat ze data veel compacter kunnen opslaan. Hierdoor kunnen MLC's met dezelfde opslagcapaciteit goedkoper worden geproduceerd dan SLC's.

Er zijn ook cellen die nog meer bits op kunnen slaan, dit zijn TLC (Triple Level Cell) en QLC (Quad Level Cell).

## TRIM

Wanneer SSD's veel gebruikt worden, worden ze trager. Bij het schrijven in eerder gebruikte ruimte moet dit gebied eerst gewist worden, en deze handeling kost extra tijd. Daarom heeft men een nieuwe technologie ontwikkeld die dit tegengaat, namelijk TRIM. TRIM zorgt ervoor dat het besturingssysteem zoekt naar gebieden op de SSD die niet meer gebruikt worden. Vervolgens krijgt de controller van de SSD opdracht om deze gebieden alvast te wissen, zodat er zonder vertraging weer op geschreven kan worden.

De recentste SSD's zijn bijna allemaal uitgerust met deze TRIM-functie. Vóór TRIM was er wel al een alternatieve techniek die de SSD resette. Het nadeel hiervan was dat deze techniek alle data wiste die op de SSD stond, waardoor hij niet praktisch was om te gebruiken.

TRIM is geïntegreerd in de [Linuxkernel](https://nl.wikipedia.org/wiki/Linuxkernel" \o "Linuxkernel) vanaf versie 2.6.33, [macOS](https://nl.wikipedia.org/wiki/MacOS" \o "MacOS) vanaf [versie 10.7](https://nl.wikipedia.org/wiki/OS_X_10.7) *Lion* en bij Windows vanaf [Windows 7](https://nl.wikipedia.org/wiki/Windows_7) en vanaf [Windows Server 2008 R2](https://nl.wikipedia.org/wiki/Windows_Server_2008_R2).

## Voor- en nadelen

Dit zijn de voornaamste voor- en nadelen van een SSD ten opzichte van een harde schijf.

### Voordelen

* Snelheid: een SSD heeft een zeer korte toegangstijd en zeer hoge lees- en schrijfsnelheid.
* Stilte: doordat een SSD geen bewegende onderdelen bevat, produceert hij geen geluid.
* Gewicht en formaat: een SSD is vele malen lichter dan een harde schijf en kan ook zeer compact geïntegreerd worden in bijvoorbeeld een laptop.
* Zuiniger: een SSD heeft minder vermogen nodig om te werken dan een harde schijf, doordat hij geen draaiende delen bevat.
* Koeler: geen bewegende onderdelen dus minder warmteproductie.
* Minder kwetsbaar: omdat een SSD geen bewegende onderdelen heeft is deze minder gevoelig voor fysieke schokken en stoten.

### Nadelen

* Prijs: een SSD is (gerekend in prijs per [gigabyte](https://nl.wikipedia.org/wiki/Gigabyte)) nog steeds duurder dan een harde schijf (vroeger zelfs vele malen duurder).
* Degradatie (bij flashgeheugen): De cellen zullen na een aantal malen schrijven hun geheugencapaciteit verliezen. MLC's kunnen ongeveer 5000 keer beschreven worden. Hoe meer bits per cel worden opgeslagen, des te sneller deze cellen zullen slijten. TRIM zorgt ervoor dat alle cellen ongeveer gelijk belast worden. Zo gaat de hele schijf langer mee.

Opgeloste nadelen:

* Capaciteit: de capaciteit van een SSD was lange tijd niet zo groot als die van een harde schijf. Met de introductie van 2TB-SSD's en hoger, tot 32 TB, is dit probleem echter opgelost.
* Verlies van snelheid: als geen maatregelen (zoals bijvoorbeeld [TRIM](https://nl.wikipedia.org/wiki/Solid_state_drive#TRIM)) getroffen worden, wordt een SSD na veel herschrijven van data trager. TRIM wordt bij de meeste besturingssystemen (zoals Windows 7 en later) standaard geactiveerd voor SSD's.

## Toepassingen

SSD's hebben de mechanische harde schijven grotendeels vervangen. SSD's die gebaseerd zijn op vluchtig geheugen zoals SDRAM worden gekenmerkt door snelle toegang. Er zijn SSD's met toegangstijden van minder dan 0,01 milliseconde; meer dan 400 keer zo snel als de snelste harde schijven met anno 2012 een toegangstijd van 4 milliseconden. Voor consumenten die een doorsnee SSD kopen ligt de toegangstijd rond de 0,1 milliseconde. Omdat de prijs per gigabyte opslagcapaciteit vooralsnog hoger is dan bij conventionele harde schijven wordt een SSD soms gecombineerd met een "gewone" harde schijf voor het opslaan van (grote hoeveelheden) data.

Solid state drives kunnen applicaties die in hun snelheid beperkt worden door beperkingen van de harde schijf versnellen. Ze worden ook gebruikt als "boot disk" (opstartschijf). Ook hier spelen de hogere lees- en schrijfsnelheden een rol.

Ook kunnen SSD's nuttig zijn in computers die al de maximum toegelaten hoeveelheid RAM gebruiken. Sommige [x86](https://nl.wikipedia.org/wiki/X86-instructieset)-architecturen hebben bijvoorbeeld limieten van 4 GB aan RAM. Door gebruik te maken van een [wisselbestand](https://nl.wikipedia.org/wiki/Virtueel_geheugen) op een SSD kan dit probleem omzeild worden. Door beperkingen op de bandbreedte van de bus waarmee zo'n SSD met de rest van de computer verbonden is zal de SSD weliswaar niet de snelheid van het hoofdgeheugen halen, maar de snelheid blijft vele malen hoger dan wanneer het wisselbestand op een conventionele harde schijf zou staan. Dit komt echter de levensduur van de SSD niet ten goede, gezien het beperkte aantal schrijfcycli bij flashgeheugencellen.

Op DRAM gebaseerde SSD's kunnen ook als [cache](https://nl.wikipedia.org/wiki/Cache_(tijdelijk_geheugen)) gebruikt worden. Wanneer data naar een harde schijf weggeschreven moet worden, zal het overeenkomstige blok als gewijzigd (jargon: *vuil* of *dirty*) gemarkeerd worden. Alle gewijzigde blokken kunnen dan naar de harde schijf gesynchroniseerd worden op basis van een van de volgende strategieën:

* tijd, bijvoorbeeld elke 10 seconden;
* drempel, wanneer het percentage gewijzigde blokken een bepaalde vooraf gedefinieerde waarde overschrijdt;
* een combinatie hiervan.

Eind 2011 werd de eerste supercomputer met 1024 SSD's in gebruik genomen, goed voor de 48e plaats in de top 50 van supercomputers.

## SSHD

Indien een solid state drive als flashgeheugen in één fysieke unit wordt gecombineerd met een mechanische harde schijf, spreekt men van een [hybride schijf](https://nl.wikipedia.org/wiki/Hybride_schijf) of solid state hybrid drive (SSHD). Het aanwezige, vaak kleine flashgeheugen wordt ingezet als [cachegeheugen](https://nl.wikipedia.org/wiki/Cache_(tijdelijk_geheugen)). De rest van de schijf wordt gevormd door de harde schijf, die vaak een hogere opslagcapaciteit heeft. Hierbij combineert men de snelheid van een SSD met een goedkopere en ruimere harde schijf.

## Zie ook

* [USB-stick](https://nl.wikipedia.org/wiki/USB-stick)
* [Extern geheugen](https://nl.wikipedia.org/wiki/Extern_geheugen)

# 3D V-NAND: de hoogte in met flash-opslag

David van Dantzig 30 november 2014 05:59[14 reacties](https://nl.hardware.info/artikel/5765/3d-v-nand-de-hoogte-in-met-flash-opslag#reacties)

[4 besproken producten](https://nl.hardware.info/artikel/5765/3d-v-nand-de-hoogte-in-met-flash-opslag#besprokenproducten)

## Inleiding

Het belang van flashgeheugen kan nauwelijks overschat worden. Van USB-stick tot smartphone, overal treffen we deze schokvaste en snelle vorm van vaste opslag aan. Nu de verkleining van het productieprocedé tegen de natuurkundige grenzen aanloopt, komen fabrikanten als Samsung met een driedimensionale uitwerking. Hoe werkt dat eigenlijk precies? Dat proberen we je in dit artikel uit te leggen.

Samsungs 850 Pro SSD-serie, enige maanden geleden geïntroduceerd, zette niet alleen een nieuwe benchmark neer op het gebied van snelheid. Ook was het de eerste voor consumenten beschikbare SSD die gebruikmaakt van wat Samsung 3D V-NAND noemt. Dit is, heel simpel gezegd, een vorm van flashgeheugen waarbij de transistors niet in een enkele laag (tweedimensionaal) gerangschikt zijn, maar ook in lagen boven elkaar geplaatst zijn. V-NAND staat dan ook voor ‘vertical NAND’. Dat klinkt als een eenvoudige oplossing, maar feitelijk is dit een technisch hoogstandje van de bovenste plank.

De reden voor de ontwikkeling van 3D flashgeheugen (de versie in de 850 Pro’s is Samsungs tweede generatie) is dat het verkleinen van het productieproces op korte termijn niet meer mogelijk is. Zoals bekend is een groot deel van de vooruitgang in de wereld van halfgeleiderchips gebaseerd op het fenomeen van verkleining van het productieprocedé: kleinere transistors betekenen dat er meer op een wafer passen, waardoor dezelfde hoeveelheid transistors (en dus opslag, of rekenkracht) minder kost. Omgekeerd werkt het ook: voor hetzelfde geld krijg je meer opslag of rekenkracht. Dit is feitelijk wat de Wet van Moore stelt: het aantal transistors in een geïntegreerde schakeling verdubbelt elke twee jaar dankzij deze technologische vooruitgang.

Voor flashgeheugen, dat er in meerdere soorten is maar waarbij we ons hier focussen op het NAND-type, betekent dit dus dat een flashchip bij gelijkblijvende grootte meer capaciteit heeft, naarmate het productieprocedé kleiner wordt. Deze procedés worden tegenwoordig in nanometers aangegeven: 40 nm, 28nm, 22nm, en zo verder. Die aanduiding strookt niet geheel met de realiteit. Het geeft een indicatie van het kleinste onderdeel van het procedé, wat bijvoorbeeld ook de ruimte tussen twee transistors kan zijn. Desalniettemin hebben we het ontegenzeggelijk over zeer kleine afmetingen.

Het verkleinen stelt de halfgeleiderfabrikanten voor steeds grotere uitdagingen én kosteninvesteringen om die te overwinnen. Daarbij zijn twee zaken fundamenteel lastig: in de eerste plaats, hoe fabriceer je het überhaupt en in de tweede plaats, hoe zorg je ervoor dat het blijft werken? Op beide zaken komen we verderop in dit artikel terug. Het aantal bedrijven dat in staat is de kosten voor de overstap naar een nieuw, kleiner productieprocedé op te hoesten, is inmiddels tot een handvol geslonken. Op flashgebied zijn Samsung, Micron, Sandisk/Toshiba en SK Hynix de namen waar het om draait. Welk merk SSD je ook koopt, grote kans dat er flashchips van een van deze vier merken in zit.

Alle zetten in op verkleining van het productieprocedé, maar men weet ook dat daar een einde aan komt. Op een gegeven moment worden transistors zo klein, dat ze uit nog maar enkele atomen bestaan. Op dat punt houden de klassieke natuurkundige wetten op en worden de wetten van de kwantumfysica van toepassing – en die maken voorspelbaar gedrag bijzonder lastig. Dat is een probleem, want je wilt zeker weten dat een bit is weggeschreven: “misschien” is geen antwoord dat je wilt hebben als antwoord op de vraag of je kostbare data is veiliggesteld. Maar ook daarvoor ontstaan er al problemen, waarover verderop meer.

Door allerlei slimme trucs, waaronder ingenieus materiaalgebruik en constructies, lukt het de fabrikanten toch nog om hun productieprocedé te verkleinen. In 2003 voorspelde Intel dat 60 nm de grens was, maar inmiddels heeft het al 14nm transistors in massaproductie. Voor flashgeheugen is Toshiba al op 19nm beland voor massaproductie. In beide gevallen geldt dat formaat zoals gezegd alleen voor de afstand tussen transistors, de transistors zelf zijn marginaal groter. Alle drie de flashfabrikanten werken aan 16nm procedés; voor NAND-transistors lijkt de limiet vooralsnog op zo’n 15nm te liggen. Hoewel voorspelde grenzen dus eerder al werden overschreden, komt het einde van de mogelijkheden van miniaturisatie nu toch echt in zicht. Om die reden werken de bedrijven ook aan alternatieve oplossingen, en Samsung loopt daarmee voorop. Voordat we daarop ingaan, eerst wat over hoe transistors in NAND-geheugen werken en in elkaar steken.

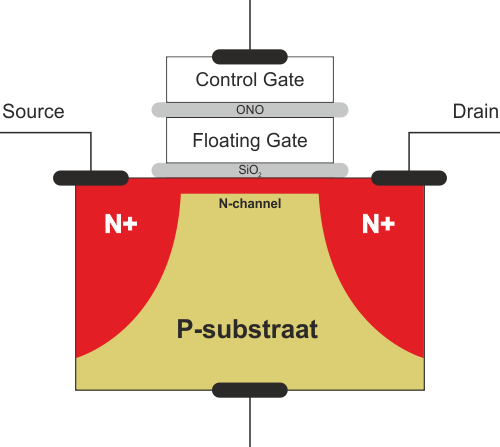
# 3D V-NAND: de hoogte in met flash-opslag

David van Dantzig 30 november 2014 05:59[14 reacties](https://nl.hardware.info/artikel/5765/2/3d-v-nand-de-hoogte-in-met-flash-opslag-transistors#reacties)

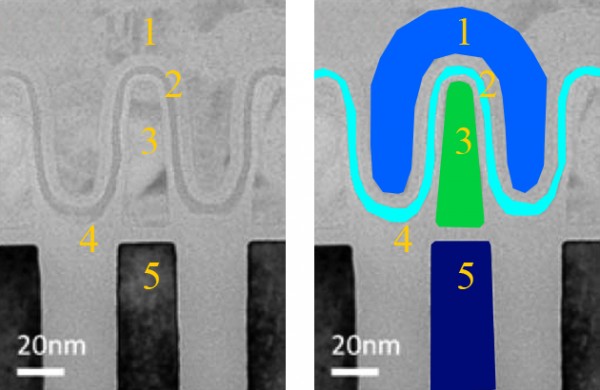
[4 besproken producten](https://nl.hardware.info/artikel/5765/2/3d-v-nand-de-hoogte-in-met-flash-opslag-transistors#besprokenproducten)

## Transistors

In de basis zijn er maar twee soorten transistors: bipolaire en (unipolaire) veldeffect-transistors. Van het laatste type maakt NAND-geheugen gebruik, om precies te zijn van n-channel MOSFETs (figuur 1). Heel kort door de bocht: NAND-geheugen slaat data op door elektronen op te vangen in een zogenaamde floating gate van zo’n mosfet. Hoe dat in detail werkt, leggen we hieronder uit.



Figuur 1: een schematische weergave van een NAND mosfet. ONO is oxide-nitride-oxide, ook wel Inter Poly Dielectric, een isolerende laag. Het laagje SiO2 ofwel siliciumdioxide is ook een isolerende laag, waar elektronen bij toepassing van een hoge spanning op de control gate doorheen kunnen ‘tunnelen’, naar de floating gate.



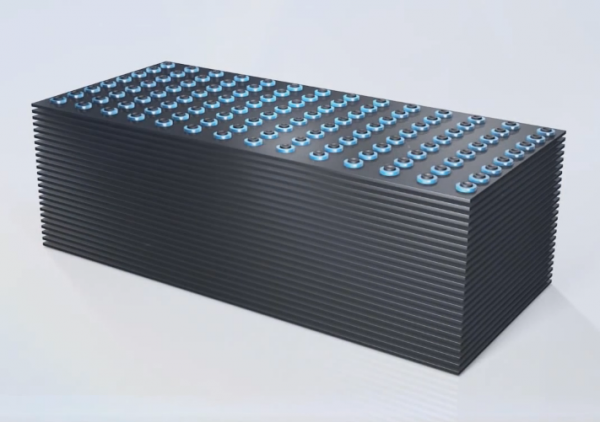
Figuur 2: een dwarsdoorsnede van een NAND mosfet. 1: control gate, 2: oxide-nitride-oxide, 3: floating gate, 4: siliciumoxide (tunneling oxide), 5: bitline (substraat)

Het n-channel komen we zo op terug, de afkorting MOSFET zelf staat voor Metal-Oxide-Semiconductor Field-Effect Transistor. Ofwel: een metaal-oxide halfgeleider veldeffect transistor. Een veldeffect transistor is een type transistor met maar één pool. Standaard loopt er dan ook geen stroom door een mosfet. Hij heeft drie aansluitingen: een source, een drain en de gate. Daarnaast dient het substraat als vierde ‘aansluiting’, dat intern verbonden is met de source.

De gate dient om een kanaal te creëren waardoor stroom kan lopen tussen de source en de drain. Dat gebeurt door het elektrische veld van de spanning op de gate te wijzigen. Hoe, daar komen we zo op. Eerst de n-channel: dat is simpelweg een geleidingskanaal tussen source en drain.

In mosfets in NAND-geheugen is de gate elektrisch geïsoleerd, om welke reden het een ‘floating gate’ wordt genoemd. De isolatie van de source en de drain is een dunne laag siliciumoxide. De floating gate wordt aangestuurd door een control gate. Dat is in moderne mosfets onderdeel van dezelfde transistor, maar het kan ook een aparte transistor zijn. Ook met de control gate is er geen directe elektrische verbinding: een laagje oxide-nitride-oxide ofwel ONO, ook wel Inter Poly Dielectric ofwel IPD genaamd (gewoon een ingewikkelde aanduiding voor een slecht geleidend materiaal) zorgt voor een strikte scheiding. De control gate beïnvloedt de floating gate louter via een capacitieve verbinding, een heel zwakke, indirecte elektrische verbinding.

In NAND-geheugen is een hele reeks mosfets met elkaar verbonden in een raster. In één richting zijn ze verbonden via de control gates. Die verbinding noemen we de wordline. Haaks daarop loopt een verbinding via het substraat, de bitline.

  
Render van schematische weergave van 3D V-NAND

# 3D V-NAND: de hoogte in met flash-opslag

David van Dantzig 30 november 2014 05:59[14 reacties](https://nl.hardware.info/artikel/5765/3/3d-v-nand-de-hoogte-in-met-flash-opslag-flash-programmeren#reacties)

[4 besproken producten](https://nl.hardware.info/artikel/5765/3/3d-v-nand-de-hoogte-in-met-flash-opslag-flash-programmeren#besprokenproducten)

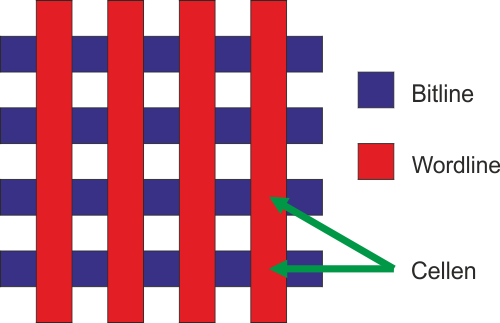
## Flash programmeren

Nu de basisstructuur geschetst is, kunnen we uitleggen hoe een flash-cel wordt geprogrammeerd. Bekijk hiervoor figuur 3. Dit gebeurt door een relatief hoge spanning te zetten op de wordline (en dus op een rij control gates) – bijvoorbeeld 20 volt. Hierdoor wordt een elektrisch veld gecreëerd, waardoor elektronen kunnen ‘tunnelen’ (zeg maar overspringen) van de bitline, door het silicumoxide, naar de floating gate. Die wordt dus voorzien van een elektrische lading. Wanneer dat proces compleet is en de gate dus negatief geladen is, is de cel geprogrammeerd en wordt het voltage van de wordline weer naar 0 verlaagd.

Omdat de wordline een hele reeks cellen met elkaar verbindt, moet je voorkomen dat al die cellen geprogrammeerd worden met dezelfde data. Dat doe je door tegelijkertijd via de bitline een lagere spanning van 6V te zetten op de naburige cellen; alleen de cel die je wilt programmeren blijft op 0V. Doordat het spanningsverschil tussen 20 en 0 groter is dan tussen 20 en 6, springen de elektronen in het eerste geval wel, en in het tweede geval niet over.

Wissen van een cel gaat omgekeerd: de wordline en dus de control gates wordt op 0V gehouden, terwijl een hoge spanning van 20V op de bitline ervoor zorgt dat de elektronen in de omgekeerde richting bewegen, van de floating gate terug naar de bitline en dus het silicium substraat.

Om een cel uit te lezen wordt de wordline op 0V gehouden, terwijl verschillende voltages worden toegepast op de bitline. Zodra het juiste voltage wordt toegepast, zal de cel gaan geleiden. Een onderdeel met de naam sense amplifier leest op dat moment de lading uit en converteert die naar de corresponderende bitwaarde, in het simpelste geval 0 of 1, maar ook dat kan complexer liggen.



Figuur 3: Heel eenvoudig geschetst is dit hoe de cellen in NAND-geheugen gestructureerd zijn.

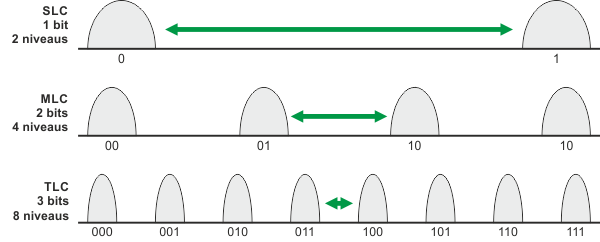
# 3D V-NAND: de hoogte in met flash-opslag

David van Dantzig 30 november 2014 05:59[14 reacties](https://nl.hardware.info/artikel/5765/4/3d-v-nand-de-hoogte-in-met-flash-opslag-uitdagingen#reacties)

[4 besproken producten](https://nl.hardware.info/artikel/5765/4/3d-v-nand-de-hoogte-in-met-flash-opslag-uitdagingen#besprokenproducten)

## Uitdagingen

Het principe van het programmeren van flashgeheugen is altijd hetzelfde, maar er is wel verschil in hoeveel bits er in een cel kunnen worden opgeslagen. Bij single level cell-geheugen (SLC) is dat één bit: 0 of 1. Bij MLC is dat twee bits, wat vier waardes kunnen zijn: 00, 01, 10 of 11. Bij TLC ten slotte zijn dat drie bits, voor deze mogelijke waardes: 000, 001, 010, 011, 100, 101, 110 en 111. Elk van die bitwaardes wordt vertegenwoordigd door een (negatieve) elektrische lading, ofwel een hoeveelheid elektronen. Bij SLC zijn dus twee ladingsniveaus nodig (waarvan één 0 kan zijn), bij MLC zijn dat er al vier en bij TLC zelfs acht. Dat heeft twee belangrijke consequenties, die zwaarder gaan wegen naarmate het productieprocedé kleiner wordt.



In TLC-geheugen moet een cel acht verschillende ladingen kunnen bevatten.

Hoe fysiek groter de transistor, hoe groter de floating gate en hoe meer elektronen die kan opslaan, hoe eenvoudiger het is om in die hoeveelheid te variëren om meerdere niveaus aan te brengen. Hoe kleiner de transistor, hoe kleiner de floating gate, hoe minder elektronen erin passen, hoe moeilijker het is om meerdere goed van elkaar te onderscheiden ladingniveaus aan te brengen. Ook zal duidelijk zijn dat je bij MLC (en nog meer bij TLC) minder speelruimte hebt voor voltagewisselingen. Immers, met dezelfde maximale hoeveelheid elektronen moet je bij TLC acht bitwaardes kunnen representeren, versus twee bij SLC.

Het probleem van de continue productieprocedéverkleining is zoals gezegd tweeledig. In de eerste plaats zorgt het proces waarmee een cel wordt voorzien van een lading voor slijtage. Het hoge voltage waarmee NAND geprogrammeerd moet worden, zorgt ervoor dat de isolatielagen (het ONO en het siliciumoxide) wegslijten, wat in de praktijk betekent dat ze hun isolerende eigenschappen verliezen. Daardoor kunnen elektronen uiteindelijk niet meer wegstromen uit de floating gate, of niet meer terugkeren naar het siliciumsubstraat. Dat zorgt voor een ongewenste verandering van het voltage in de cel en daarmee een foutief opgeslagen bitwaarde. Bij MLC en TLC, waarbij de voltages dichter op elkaar liggen, treedt dat effect significant sneller op – vandaar de lagere levensduur, cq. het kleinere aantal schrijfcycli dat deze typen flash aankunnen. Naarmate deze flashcellen kleiner worden, wordt het effect van slijtage groter: er is immers minder materiaal dat kan slijten.

Naast de slijtage vormt ook de miniaturisering zelf een uitdaging voor de werking. Het verkleinen van de omvang van een flash-cel zorgt ook voor een verkleining van de floating gate, die bij elke stap minder elektronen kan bevatten. Ter illustratie van de dimensies waarover we het hebben; het 15nm NAND-geheugen van Toshiba en Sandisk kan minder dan 20 elektronen per cel opslaan. Dat betekent in het geval van TLC, dat er minder dan 3 elektronen per voltageniveau beschikbaar zijn! Een handvol ontsnapte (weggelekte) elektronen betekent al dat de cel niet meer functioneert, ofwel verkeerde data bevat.

Daarnaast is ook de nabijheid van de cellen een factor. Zoals we al schreven is een essentieel onderdeel van de moderne mosfet in NAND-geheugen dat de control gate door middel van een capacitieve verbinding de floating gate kan beïnvloeden. Als de individuele transistors echter te dicht op elkaar komen te liggen, neemt het risico toe dat naburige cellen elkaar via zo’n capacitieve verbinding gaan storen. Die storing varieert met de spanning die er op een naburige cel staat, dus het is geen constante, waardoor het niet mogelijk is deze storing via een algoritme te compenseren. De enige oplossing is een hoger voltage gebruiken, om de capacitieve geleiding tussen control gate en floating gate tot stand te brengen, waardoor de elektronenstroom door het oxide gaat lopen.

Een simpele analogie is een ruimte waarin diverse mensen aan het spreken zijn. Zolang ze allemaal ver genoeg uit elkaar staan, is het niet lastig om te verstaan wat de persoon zegt bij wie je het dichtstbij staat. Hoe kleiner de ruimte wordt en hoe meer de sprekers dus op elkaar komen te staan, hoe lastiger het wordt om elk daarvan individueel te verstaan. Als ze dan ook nog eens steeds zachter gaan praten, gaat het verhaal helemaal verloren. Zo gaat het ook met NAND-flashgeheugen: hoe dichter de cellen op elkaar zitten, hoe meer ze elkaar beïnvloeden, dus hoe moeilijker het is om een cel uit te lezen of te beschrijven. Dat wordt verder bemoeilijkt door de afnemende omvang van de cel.

De halfgeleiderfabrikanten hebben al heel wat gedaan om deze effecten tegen te gaan, van zogenaamde luchtbarrières tussen de wordlines tegen interferentie tot high K metal gates in plaats van klassieke oxide-nitride-oxide diëlectricums (isolerende lagen), maar het einde is nu echt in zicht. Om de Wet van Moore toch te kunnen blijven volgen en dus de kosten van transistors te blijven reduceren, heeft Samsung een radicale innovatie gedaan: het is met NAND de diepte (of liever gezegd: de hoogte) in gegaan, met vertical-NAND. Hoe zit dat?

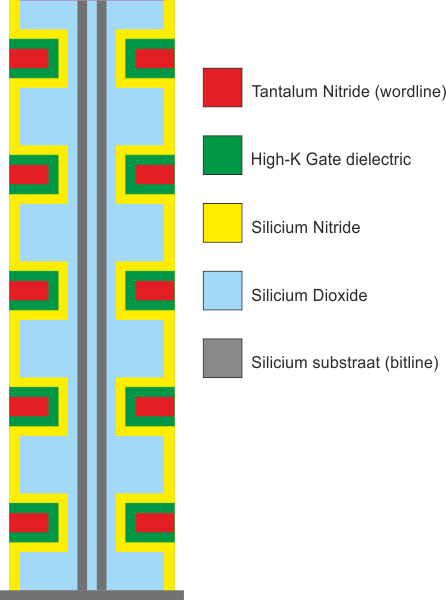
# 3D V-NAND: de hoogte in met flash-opslag

David van Dantzig 30 november 2014 05:59[14 reacties](https://nl.hardware.info/artikel/5765/5/3d-v-nand-de-hoogte-in-met-flash-opslag-3d-v-nand#reacties)

[4 besproken producten](https://nl.hardware.info/artikel/5765/5/3d-v-nand-de-hoogte-in-met-flash-opslag-3d-v-nand#besprokenproducten)

## 3D V-NAND

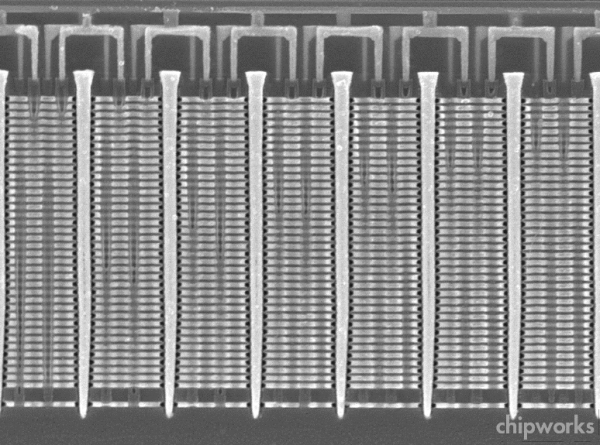
Samsung is niet de enige fabrikant die werkt aan 3D NAND, maar wel de eerste die het succesvol in de markt heeft geïntroduceerd. Sterker, in de 850 Pro series SSD’s gebruikt het al de tweede generatie van dit geheugen. De andere fabrikanten verkeren nog in een eerder stadium in het proces en daar hebben we dan ook nog geen producten van gezien. Hierdoor kunnen we alleen over Samsungs versie van dit type flashgeheugen wat meer vertellen. Veel details zijn er nog niet en voor dit artikel hebben we ons dan ook deels gebaseerd op voorwerk door andere publicaties, in het bijzonder AnandTech en het blog van Jim Handy, The Memory Guy.



Figuur 4: Samsungs 3D V-NAND met vijf niveaus (van de 32) schematisch weergegeven.

Wat Samsung heeft gedaan met haar 3D NAND is tegelijk eenvoudig en bijzonder ingenieus: het heeft feitelijk de bestaande mosfet-structuur een kwartslag gedraaid en gespiegeld in twee lagen tegen elkaar gezet, waarbij deze constructie verankerd is in een silicum substraat (dat, zoals je je zal herinneren, dient als bitline).

In figuur 4 zie je een schematische weergave van vijf lagen van 3D NAND, gebaseerd op wat Samsung haar TCAT (terabit cell array transistor) structuur noemt. Dit moet je zien als de dwarsdoorsnede van een cilinder, waarbij elke laag één, twee of drie bits kan bevatten, afhankelijk van of het gaat om SLC, MLC of TLC NAND. Samsungs huidige generatie verticaal NAND heeft overigens al 32 van deze lagen (zie figuur 5).



Figuur 5: een röntgenfoto van een dwarsdoorsnede van een aantal 3D cellen in Samsungs V-NAND (via Chipworks).

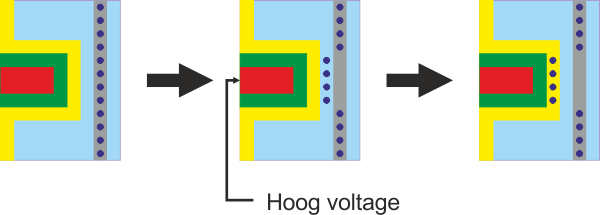
Wie het schema van de mosfet in figuur 1 goed bestudeerd heeft, ziet één groot verschil in de structuur van Samsungs 3D NAND. Waar in 2D NAND de lading wordt opgeslagen in een floating gate, is die afwezig in Samsungs verticale NAND. Samsung gebruikt in plaats daarvan een ontwerp dat het Charge Trap Flash noemt (CTF) – de lading wordt opgeslagen in een isolator, in dit geval siliciumnitride.

De reden hiervoor ligt in de aard van een floating gate. Deze kan je vergelijken met een emmer, waarin je water bewaart. Voor SLC geheugen heb je twee niveaus: vol en leeg; voor MLC vier niveaus, die elk eenvoudig van elkaar te onderscheiden zijn; voor SLC acht niveaus, die al meer op elkaar lijken. Als er een gat in de emmer zit, is 5/7 zo veranderd in 4/7, bijvoorbeeld. En vroeg of laat is het 0. Dat is ook het geval bij een floating gate: als er een lek zit in het isolerende materiaal (ONO en SiO2), zullen de elektronen ontsnappen. Zoals we beschreven hebben, neemt de kans daarop toe naarmate een flash-cel vaker wordt beschreven en daarmee slijt.

Een CTF heeft hier geen last van: een isolator geleidt nooit, dus wanneer er een ‘gat’ ontstaat, zullen niet alle elektronen ontsnappen – hooguit degene die vlakbij het gat in het isolerend materiaal zitten (de High-K Gate Dielectric in het 3D NAND schema). Doordat CTF niet vatbaar is voor slijtage, is de endurance ofwel slijtvastheid van 3D V-NAND geheugen hoger. Ook vereist CTF naar verluidt niet zulke hoge voltages om van een lading te worden voorzien. De floating gate van 2D NAND heeft zo’n hoog voltage nodig vanwege de relatief dikke laag isolerend materiaal.

Wat hetzelfde is gebleven, is het belang van de capacitieve geleiding tussen de control gate en de charge trap. Door toepassing van een high-K diëlektricum, ofwel een materiaal met een hoge dielektrische constante (K) kan Samsung de lagen binnen het 3D NAND dichter op elkaar stapelen, dan het zou kunnen doen met een traditioneel dïelektricum zoals ONO, dat dikker zou moeten zijn voor hetzelfde effect.

Verder is 3D V-NAND eveneens zeer vergelijkbaar met 2D NAND. Om een cel te programmeren wordt de bitline van die cel op 0V gehouden, terwijl de wordline van een hoger voltage wordt voorzien. Hierdoor springen de elektronen van de bitline over (‘tunnelen’) naar het siliciumnitride van de charge trap (CTF). Dit hebben we in figuur 6 geïllustreerd, waarbij de bolletjes de elektronen zijn.



Figuur 6: schematische weergave van het programmeren van een V-NAND cel. Een hoge spanning op de wordline creëert een tunneleffect waardoor elektronen van de bitline overspringen (tunnelen) naar de charge trap.

# 3D V-NAND: de hoogte in met flash-opslag

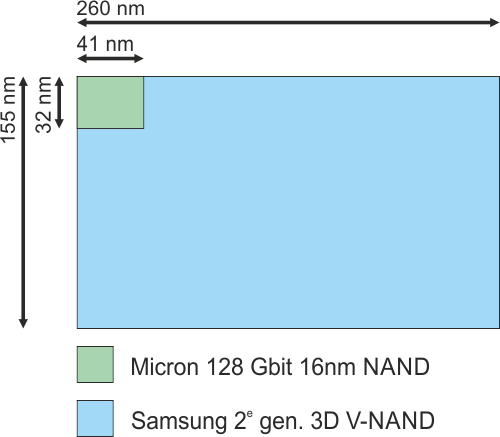
David van Dantzig 30 november 2014 05:59[14 reacties](https://nl.hardware.info/artikel/5765/6/3d-v-nand-de-hoogte-in-met-flash-opslag-consequenties#reacties)

[4 besproken producten](https://nl.hardware.info/artikel/5765/6/3d-v-nand-de-hoogte-in-met-flash-opslag-consequenties#besprokenproducten)

## Consequenties

Nu duidelijk is hoe 3D NAND werkt en hoe het zich verhoudt tot traditioneel 2D NAND, kunnen we nog kort ingaan op wat de consequenties zijn voor de toekomst. Zoals we nu weten, zijn de twee basisproblemen van 2D flashgeheugen de steeds kleinere celgrootte en de nabijheid ervan. In de praktijk betekent dat een fabricage-uitdaging: de vereiste lithografie om met groottes van luttele nanometers te werken, is extreem moeilijk en foutgevoelig. Daarnaast betekent het een levensduur- en werkingsuitdaging: hoe kleiner het procedé, hoe gevoeliger voor slijtage en storingen. 3D NAND vormt een oplossing hiervoor, maar ook 3D NAND moet kunnen blijven doorschalen. Daar helpt juist die derde dimensie bij.

Door meer lagen verticaal te stapelen, is het probleem van verdere verkleining in elk geval voorlopig uit de wereld: de cellen gaan ‘gewoon’ de hoogte in. Samsung heeft voor haar tweede generatie 3D V-NAND een “ouderwets” 40 nm procedé gebruikt. Daardoor is er meer ruimte tussen geheugencellen en zijn er veel minder problemen met storing. Bovendien kunnen er veel meer elektronen in een cel worden opgeslagen, wat de problemen van lekken reduceert, terwijl het gebruik van een CTF in plaats van een floating gate daar ook een handje bij helpt.



Figuur 7: qua oppervlakte is een cel in Samsungs tweede generatie 3D V-NAND een stuk groter dan bijvoorbeeld een 16nm cel in Microns state-of-the-art 2D NAND.

Hoewel hierdoor een Samsung 3D geheugencel veel groter is dan een 16nm cel van bijvoorbeeld Micron (zie figuur 7), zijn de kosten per cel hierdoor ongeveer gelijk. Het is namelijk veel duurder om een die ofwel een chip met geheugencellen groter te maken, dan om deze dikker te maken. Samsung is van plan om in 2017 al 1Tbit dies te maken; de huidige generatie is 86Gbit groot, dus dat is een verdubbeling elk jaar (256 Gbit in 2015, 512Gbit in 2016). Het interessante is dat Samsung dat wil gaan doen door simpelweg de hoogte in te gaan, en dus niet het procedé voor 3D NAND verder te verkleinen. Het Koreaanse bedrijf heeft al aangegeven dat het dit onder andere wil doen door TLC te gebruiken, dus drie bits per cel op te slaan. Daarnaast blijft natuurlijk de mogelijkheid open om ook 3D V-NAND op een kleiner procedé te maken, wat eenvoudiger zal worden naarmate de techniek volwassener wordt.