

DataRecover bvba

Burchtstraat 2
9300 Aalst
Belgium

Tel. : 0475 / 76.66.73

Fax : 053 / 41.75.84



Registratie nr. : 0826.283.018
BTW nr. : BE 0826.283.018

Bank : 001-6132677-25
Iban : BE70 0016 1326 7725
Bic : GEBABEBB

Email : info@datarecover.be
Website : www.datarecover.be

Harde schijven

Inhoudstabel

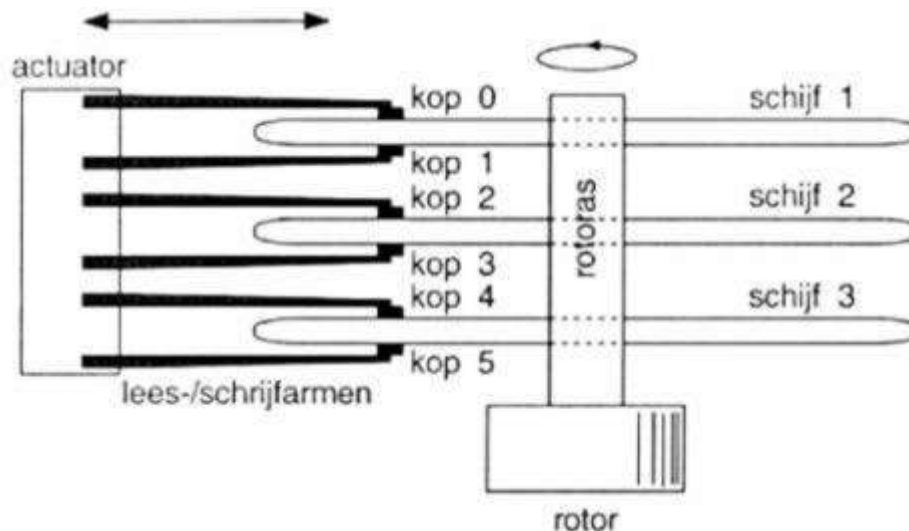
- **Inhoudstabel**
- **De opbouw van een harde schijf**
 - De opbouw van een harde schijf
 - Crash!
 - De fysische crash
 - De logische crash
- **Fysieke indeling van het magnetisch oppervlak**
 - koppen
 - Sporen
 - Sectoren
 - Logische sectoren
- **Spoor en sector layout**
 - Spoor en sector layout van een harde schijf
- **Magnetische coderingen: FM, MFM, RLL**
 - FM codering
 - MFM codering
 - RLL codering
- **CRC en ECC: Fouten detecteren en corrigeren**
 - Pariteit
 - Cyclic Redundancy Check en Error Correction Code
- **Logische indeling**
 - Wat is een logische indeling?
 - Partities
 - Bootsector
 - Cluster
 - Rootdir
 - Bestandstoewijzingstabel (FAT)
- **Dataoverdracht: de verschillende controllertypes**
 - ST412/506
 - ESDI
 - IDE, AT-bus, ATA-interface
 - SCSI
- **Dataoverdrachtsnelheid en toegangstijd**
- **Dissectie van een harde schijf**

De opbouw van een harde schijf

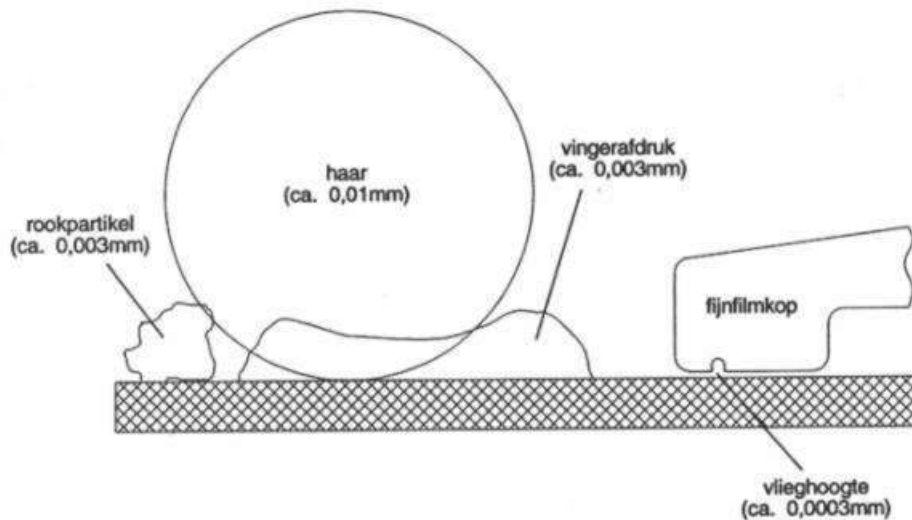
De opbouw van een harde schijf

De verschillen in werking tussen een floppy en een harde schijf zijn eigenlijk niet erg groot. De opmerkelijkste verschillen zijn de volgende:

- Zoals de naam het al zegt is een harde schijf hard, dus niet soepel en plooibaar.
- Er kunnen zich meerdere schijven met een gemagnetiseerde laag boven elkaar bevinden. Iedere schijf heeft dan zijn eigen paar lees- en schrijfkoppen. (één voor elke zijde.) Zie figuur.
- De schijf draait sneller rond, minstens 3600 omwentelingen per minuut. Veel gebruikte toerentallen voor hedendaagse schijven zijn: 5400 min⁻¹ en 7200 min⁻¹. De snellere harde schijven hebben omwentelingssnelheden van 9600 min⁻¹ en 10.000 min⁻¹.
- De schrijf- en leeskoppen raken het oppervlak niet. Zou het dat wel doen, dan zou het schijfoppervlak en / of de kop beschadigt raken wegens de hoge omwentelingssnelheid. Juist die hogere draaisnelheid zorgt voor een flinterdun luchtlaagje waarop de kop 'zweeft'. Een typische vlieghoogte is ergens tussen 0,0001 en 0,0003mm. (Zie tekening.)



Omdat harde schijven een hogere draaisnelheid hebben, moeten deze ook beter afgeschermd worden tegen onzuiverheden. Daarom zijn harde schijven doorgaans luchtdicht verpakt en zit er een filter in om deeltjes die toch afslijten op te vangen.



Crash!

Iedereen heeft al wel eens van een harde schijf crash gehoord. Maar wat is dat eigenlijk? Men onderscheidt twee soorten:

- Een fysische crash (ook wel een head-crash genoemd)
- Een logische crash

De fysische crash

Bij een fysische crash komt de lees- en schrijfkop (niet onzacht) in aanraking met het schijfoppervlak (bijvoorbeeld door een harde schok). Het resultaat is meestal dat de schrijf- en leeskop afbreekt en dan steeds meer schade gaat aanrichten omdat de schijf nog steeds blijft draaien. De schade die aangericht wordt bestaat voornamelijk uit het 'afschrapen' van de magnetische laag. (Deze afgeschraapte deeltjes gaan op hun beurt schade aanrichten, maar blijven meestal in de filter steken, indien aanwezig.)

Op de plaats waar de magnetische laag weg is, zijn natuurlijk ook de gegevens verdwenen. Het gebeurt evenwel dat de lees- en schrijfkop afbreekt na een harde schok zonder dat er gegevensverlies is, dit komt omdat op de plaats van de impact juist geen data staan. Het belangrijkste na een fysische crash is de schijf niet langer te laten draaien!

Er zijn gespecialiseerde bedrijven (www.datarecover.be) die gegevens kunnen terug halen. De gegevens die op niet beschadigde oppervlakken staan, natuurlijk!



De logische crash

Bij een logische crash (dit is de meest voorkomende crash) blijven de gegevens en het schijfoppervlak volledig intact, ze kunnen alleen niet meer toegankelijk gemaakt worden. Dit komt omdat het besturingssysteem niet meer weet waar een bepaald bestand staat. Dit kan bijvoorbeeld gebeuren door een verandering van de partitietabel (Zie verder) al dan niet veroorzaakt door een computervirus.

Soms is het mogelijk om deze gegevens weer te voorschijn te halen, soms ook niet. Het meest frustrerende bij een logische crash is dat de gegevens nog altijd fysisch aanwezig zijn.

Fysieke indeling van het magnetisch oppervlak

Koppen

Zoals gezegd kunnen er dus meerdere schrijf- en leeskoppen zijn. Deze koppen krijgen elk een nummer. Men begint bij kop 0 en telt zo verder.

Een diskette heeft dus een kop 0 en een kop 1.

De Engelse vertaling van kop is "head".

Sporen

Als de schrijf- en leeskop stilstaat en de schijf ronddraait, kan men alle gegevens lezen die zich op de cirkel bevinden met als middelpunt de as van de aandrijfmotor.

Zo'n cirkel noemt men een spoor. (Engels: track.)

Door de schrijf- en leeskop te verplaatsen wijzigt men in feite de straal van de cirkel en bekomt men grotere of kleinere cirkels. Dit zijn dan andere sporen.

Een standaard floppy bevat 80 sporen. (40 aan iedere zijde.)

Harde schijven hebben beduidend meer sporen.

De sporen worden genummerd van buiten naar binnen en het buitenste spoor krijgt als nummer 0. Bij de hierboven genoemde diskette is de indeling dus als volgt:

Kop 0, buitenste spoor:	spoor 0
Kop 0, op één na buitenste spoor:	spoor 1
Kop 0, ...:	spoor ...
Kop 0, op één na binnenste spoor:	spoor 38
Kop 0, binnenste spoor:	spoor 39
Kop 1, buitenste spoor:	spoor 40
Kop 1, op één na buitenste spoor:	spoor 41
Kop 1, ...:	spoor ...
Kop 1, op één na binnenste spoor:	spoor 78
Kop 1, binnenste spoor:	spoor 79

De sporen die boven elkaar liggen (hier bijvoorbeeld spoor 0 en spoor 40) noemt men een cilinder.

Bij oude diskettes waar er maar 1 zijde beschreven werd, is een spoor dus hetzelfde als een cilinder.

Bij harde schijven kan een cilinder dus uit meerdere sporen bestaan.

Sectoren

Een spoor is op zijn beurt weer onderverdeeld in sectoren. Het aantal sectoren per spoor zal voornamelijk bij harde schijven sterk verschillen, maar bij een standaard diskette zitten er 18 sectoren in één spoor.

Ook deze sectoren worden genummerd, maar hier begint men bij 1 te tellen.

Sector 1 ligt op een speciale plaats. De sectornummering begint op de plaats die wordt aangegeven door de uitsparing in het metalen schijfje van de diskette. Bij een harde schijf begint deze sector op een plaats die is aangegeven door een speciale magneetcode.

De fysische indeling in sporen en sectoren wordt gemaakt door middel van een low-level format. Als men een diskette formateert wordt er altijd een low-level format uitgevoerd. Bij een harde schijf kan via de BIOS meestal een low-level format uitgevoerd worden.

OPGEPAST: Bij een harde schijf is een low-level format reeds in de fabriek uitgevoerd en mag men in sommige gevallen zelf geen tweede low-level format meer uitvoeren!

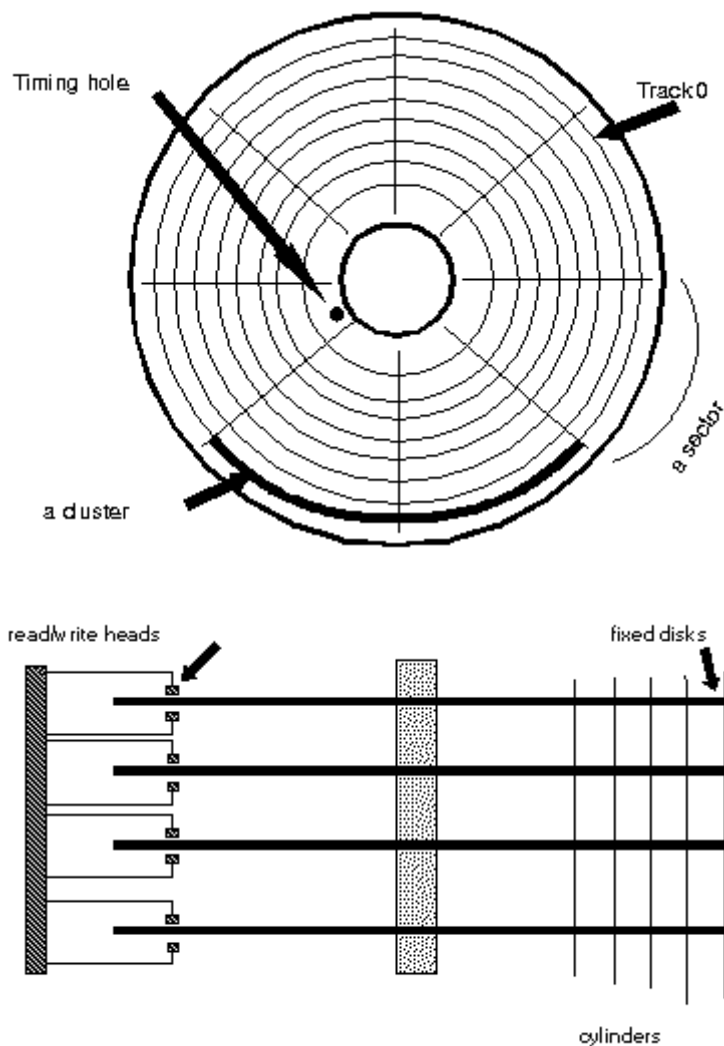
Logische sectoren.

Er zijn eigenlijk twee manieren om een sector te identificeren:

- Absoluut:
Men duidt telkens het spoornummer EN sectornummer aan.
vb: spoor 1, sector 1
- Relatief:
Men begint met sector 0 (ja, 0) te tellen en men telt verder. Sector 1 op spoor 1 wordt dan sector n+1 met n de laatste sector op spoor 0.
vb: sector 18

Beide aanduidingen vertegenwoordigen één en dezelfde sector.

Naast de gebruikelijke capaciteitsaanduiding bij opslagmedia, kun je soms ook nog de volgende vanzelfsprekende aanduidingen aantreffen: TPI (Tracks Per Inch) of BPI (Bits Per Inch). Voor een 1,44 MB floppy zijn deze waarden respectievelijk: 135 en 17.434



Spoor en sector layout

Men moet een verschil maken tussen de layout op een diskette en de layout op een harde schijf alhoewel ze grotendeels overeenkomen.

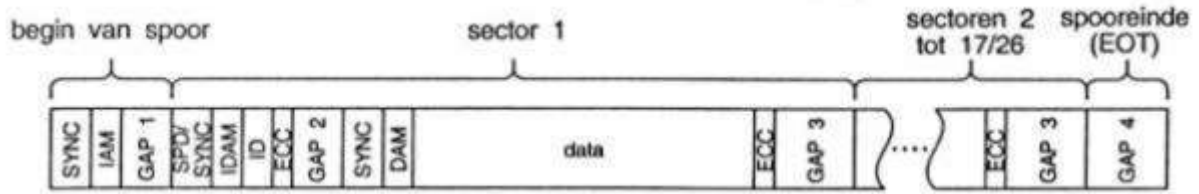
Met layout wordt bedoeld uit welke databits een spoor of een sector eigenlijk bestaat.

Bij een diskette wordt er om de data op te slaan de FM of MFM codering gebruikt.

Bij een harde schijf gebruikt men MFM of RLL codering.

Meer over deze verschillende coderingen kan je terugvinden in een later hoofdstuk.

Spoor en sector lay-out van een harde schijf



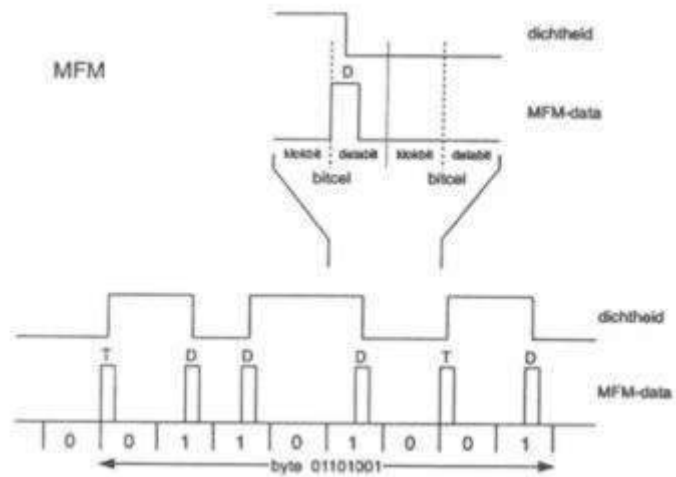
MFM formaat		RLL formaat	
Begin van het spoor:		Begin van het spoor:	
Naam:	Inhoud:	Naam:	Inhoud:
SYNC	10 bytes 00 ₁₆	SYNC	11 bytes 00 ₁₆
IAM	A1 ₁₆ FC ₁₆	IAM	A1 ₁₆ FC ₁₆
GAP 1	11 bytes 4E ₁₆	GAP 1	12 bytes FF ₁₆
Sector:		Sector:	
Naam:	Inhoud:	Naam:	Inhoud:
SPD	7 bytes 4E ₁₆		
SYNC	10 bytes 00 ₁₆	SYNC	10 bytes 00 ₁₆
IDAM	A1 ₁₆ FE ₁₆	IDAM	5E ₁₆ A1 ₁₆
ID	tr hd sc fl	ID	tr hd sc fl
ECC	4 bytes EXC waarde	ECC	4 bytes ECC waarde
GAP 2	5 bytes 00 ₁₆	GAP 2	5 bytes 00 ₁₆
SYNC	10 bytes 00 ₁₆	SYNC	11 bytes 00 ₁₆
DAM	A1 ₁₆ F8 ₁₆	DAM	5E ₁₆ A1 ₁₆
Data	512 databytes	Data	512 databytes
ECC	4 bytes ECC waarde	ECC	4 BYTES ECC waarde
GAP 3	15 bytes 00 ₁₆	GAP 3	3 bytes 16 17 bytes FF ₁₆
Einde van het spoor:		Einde van het spoor:	
Naam:	Inhoud:	Naam:	Inhoud:
GAP 4	+/- 56bytes 00 ₁₆	GAP 4B	+/- 93 bytes 00 ₁₆

MFM codering

MFM is gebaseerd op FM met de volgende verandering:

- Een klokbit wordt enkel en alleen geschreven als de vorige EN de huidige bitcel geen databit met waarde 1 hebben.

Dit geeft het volgende resultaat voor dezelfde datareeks:



Het verschil tussen FM en MFM is meteen duidelijk:

Met de MFM codering kan men twee maal zoveel data op dezelfde plaats stockeren als met de FM codering.

RLL codering

RLL is de afkorting van Run Length Limited. Daarmee wordt bedoeld dat het aantal "0"-bits dat elkaar kan opvolgen tussen twee "1"-bits beperkt is.

Er zijn verschillende versies van RLL, maar de meest gebruikte is degene waarmee er minstens 2 en maximaal 7 "0"-bits tussen twee "1"-bits kunnen liggen. Deze versie van RLL wordt ook wel aangeduid met RLL 2,7.

Andere versies zijn RLL 1,7 of RLL 3,9.

Als men nu echter een 16 "0"-bits na elkaar wil schrijven, zijn er dat dus te veel. Daarom wordt de te schrijven data eerst omgezet naar een RLL 2,7 code. Deze code is te vinden in de onderstaande tabel.

Databit RLL 2,7 code

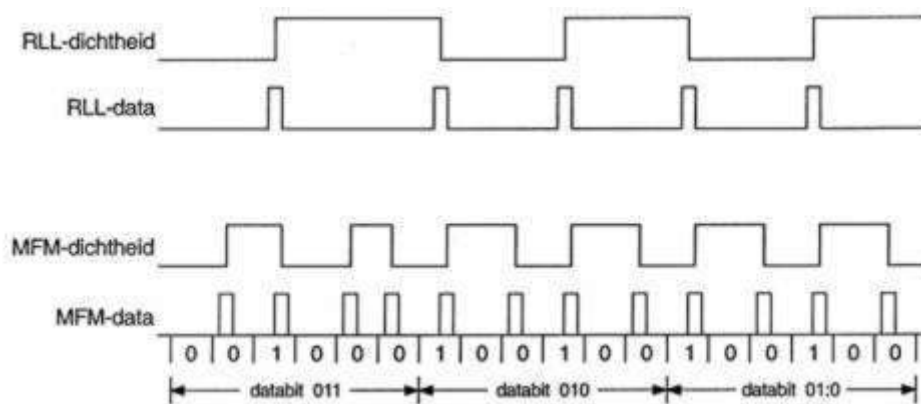
000	000100
10	0100
010	100100
0010	00100100
11	1000
011	001000
0011	00001000

De aandachtigen zullen nu al weten wat het grootste nadeel is van de RLL codering. Er zijn veel meer bits nodig (eigenlijk precies 2 maal zoveel) om de data te coderen. Hoe kan

men nu toch meer gegevens opslaan op dezelfde oppervlakte als men 2 maal meer gegevens moet opslaan?

Ik zal een antwoord proberen te geven aan de hand van de onderstaande figuur.

De databyte '011 010 01' in RLL 2,7 codering luidt dus '001000 100100 100100'. (Er van uitgaande dat de volgende databit met een '0' begint.)



Om een betere vergelijking te kunnen maken staat de RLL 2,7 bitreeks ook nog eens in MFM gecodeerd.

Bij de RLL codering worden er alleen '1' bits opgeslagen als de RLL code ook een '1' is. (Er zijn dus geen klok-bits.)

Het aantal '0' bits dat tussen twee '1' bits ligt wordt 'berekend' aan de hand van het tijdsverschil tussen twee opeenvolgende '1' bits. Om de kans op fouten te verkleinen wordt dan ook het aantal '0' bits dat elkaar kan opvolgen beperkt. (min. 2, max. 7) De kracht zit dus in de conversietabel om een databit om te zetten in een RLL code.

Een wiskundige analyse van de RLL codering en de MFM codering bewijst dat er bij MFM codering drie maal zoveel plaats in beslag genomen wordt als bij RLL codering.

We weten echter dat de RLL code twee maal zo lang is als de overeenkomstige MFM code, dus met RLL codering kan men 1,5 maal zo veel opslaan als met MFM codering op evenveel plaats.

RLL codering zorgt dus voor 50% meer opslagcapaciteit.

(Met de andere RLL coderingen, zoals RLL 3,9, kan de opslagcapaciteit ten opzichte van MFM zelfs tot 90% bedragen. Maar deze coderingen vergen betere en stabielere schijfelektronica.)

CRC en ECC: Fouten detecteren en corrigeren.

Om later te kunnen controleren (en eventueel te corrigeren) of bepaalde gegevens veranderd zijn (bijvoorbeeld verminkt tijdens de dataoverdracht), worden er extra bits aan de gegevens toegevoegd.

Pariteit

Pariteit is de eenvoudigste vorm van foutdetectie. Bij pariteit moet men op voorhand afspreken of het gaat om 'even' of om 'oneven' pariteit.

Stel, men heeft de volgende databits: 10010001 en men spreekt een even pariteit af.

Het aantal '1' bits hier is 3, hetgeen oneven is. Om het geheel even te maken, moet men dus een '1' bit toevoegen: 10010001 **1**

De databyte 00101101 beschikt al over een even aantal '1' bits, er moet dus een '0' bit toegevoegd worden: 00101101 **0**

Indien er dus ergens een bit veranderd (een '0' wordt een '1', of omgekeerd), kan men dus heel gemakkelijk bepalen of de databits correct zijn aangekomen of niet. Indien dit niet het geval is, moet de data opnieuw verstuurd worden.

Let wel: Indien er 2 dezelfde bits veranderen, is de pariteit correct. De fout wordt dus niet gedetecteerd: 11101101 **0**

Met pariteitsbits kan men ook op eenvoudige wijze een automatische correctie inbouwen.

```
1 0 0 1 0 0 0 1 1
0 0 1 0 1 1 0 1 0
0 1 0 1 1 1 0 0 0
1 0 0 0 0 0 0 0 1
1 1 1 1 1 1 0 1 1
0 0 1 0 1 0 0 1 1
0 1 1 0 1 1 1 0 1
1 1 1 0 1 0 1 0 1
0 0 1 1 0 0 0 0 0
```

De vette 'bits' zijn de pariteitsbits en deze worden zowel horizontaal als vertikaal berekend. Nu kan een fout op de horizontale as 'gezien' worden, maar ook op de verticale. Zodoende weet men exact de coördinaten van de bit die verkeerd is en kan men deze gewoon veranderen.

Cyclic Redundancy Check en Error Correction Code

CRC en ECC hebben dezelfde functie: het detecteren en mogelijk verbeteren van leesfouten, maar hier maakt men gebruik van krachtige wiskundige algoritmes. ECC is over het algemeen krachtiger dan CRC.

Het komt erop neer dat er bij het schrijven van de gegevens, ook nog een berekende controle waarde wordt opgeslagen.

Bij het lezen, wordt de controle waarde opnieuw berekend en vergeleken met de opgeslagen controlewaarde. (De controlewaarde noemt men de CRC of ECC waarde.)

Als je op je computer een melding krijgt dat er bij een floppy een leesfout is opgetreden, dan is dat omdat de CRC niet klopt.

Logische indeling

Wat is een logische indeling?

Onder logische indeling verstaat men de reeds bestaande fysische indeling klaar maken voor gebruik met een bepaald computersysteem. Er moet bijvoorbeeld bepaalde data aanwezig zijn om de computer te kunnen opstarten, om een directory structuur te kunnen creëren, etc

Deze indeling maakt men met een zogenoemde high-level format.
Telkens men een harde schijf formateert, wordt er een high-level format uitgevoerd.

Ieder besturingssysteem heeft zowat zijn eigen indeling:

EXT2 (Linux)
FAT 16 (DOS)
FAT 32 (Windows)
NTFS (Windows NT)

Partities

Voordat er echter een high level format kan uitgevoerd worden, moet de schijf verdeeld worden in partities. Een schijf wordt telkens ingedeeld in minstens één partitie. Indien er meerdere besturingssystemen op één computer aanwezig zijn, zal er waarschijnlijk voor ieder besturingssysteem een aparte partitie gemaakt moeten worden.

In de zogenaamde partitietabel wordt bijgehouden van welke sector tot welke sector een bepaalde partitie reikt.

De partitietabel staat altijd op de volgende locatie: Kop 0, Spoor 0, Sector 1, deze locatie wordt dan ook wel eens de partitiesector genoemd.

Indien deze locatie beschadigd raakt, is de schijf onbruikbaar. (Men kan niet meer vertellen waar een partitie begint, of waar deze ophoudt. Logische disk crash)

Diskettes hebben geen partitietabel!

Bootsector

De boot sector bevindt zich ofwel op de eerste sector van een schijf (floppy), ofwel in de eerste sector van een partitie (harde schijf).

In deze bootsector staan ook een aantal belangrijke gegevens: aantal sectoren per spoor, aantal koppen, aantal sectoren per cluster, aantal bytes per sector.

Verder staat er een klein programmaatje in om het besturingssysteem op te starten. (Indien dit programma de nodige bestanden niet kan vinden krijg je de melding 'ongeldige systeemdiskette' o.i.d.)

Cluster

Een cluster bestaat uit een aantal sectoren (meestal 1 to 4) en is de kleinste kwantiteit dat het besturingssysteem (DOS) kan beschrijven. Indien een sector 512 bytes groot is en een cluster uit vier sectoren bestaat, zal ieder bestand op de schijf minstens 2048 bytes in beslag nemen.

Ook een bestand van slechts 200 bytes!

Clusters worden ook gewoon opeenvolgend genummerd.

Rootdir

Voor het beheer van bestanden onder DOS is nog iets meer nodig. 'Een rootdir'. In deze rootdir staan alle bestandsnamen en directorynamen die in de root van de schijf voorkomen. De rootdir neemt maar een bepaalde hoeveelheid plaats op de schijf in beslag en daarom is het aantal bestanden en directories dat men in de rootdir kan maken, beperkt.

In de rootdir staat bijvoorbeeld de bestandsnaam met zijn extensie, de datum waarop het voor het laatst gewijzigd is, de attributen, de grootte van het bestand, of het gaat om een directory, enz

Het belangrijkste is de verwijzing naar de eerste cluster die het bestand in beslag neemt.

Bestandstoewijzingstabel (FAT)

(FAT staat voor File Allocation Table)

Als een bestand nu meerdere clusters in beslag gaat nemen, moet er ergens bijgehouden worden welke clusters het allemaal in beslag neemt. Dit wordt opgeslagen in de bestandstoewijzingstabel.

In die eerste cluster wordt er een verwijzing naar een tweede cluster gemaakt, enz. In de laatste cluster staat een zogenoemde EOF (End Of File) waarde.

Voorbeeld:

De rootdir:

NAAM	EXT	SIZE	START	ARC	R/O	SYS	HID	DIR	VOL
TESTDISK		0	0	*					*
DOC1		113	3	*					
DOC2		129	3	*					
TROPHY	WPG	3891	4	*					
MAILBAG	WPG	3353	13	*					
DOC4		104	9	*					
SCALE	WPG	3071	10	*					
DOC5		14028	8	*					
DIRECT		0	30					*	

De FAT

0	1	2	3	4	5	6	7
		<EOF>	<EOF>	5	6	7	<EOF>
8	9	10	11	12	13	14	15
17	<EOF>	11	12	<EOF>	14	15	16
16	17	18	19	20	21	22	23
<EOF>	18	19	20	21	22	23	24
24	25	26	27	28	29	30	31
25	26	27	28	29	<EOF>	<EOF>	<EOF>
32	33	34	35	36	37	38	39
0	0	0	0	0	0	0	0
40	41	42	43	44	45	46	47
0	0	0	0	0	0	0	0
48	49	50	51	52	53	54	55
0	0	0	0	0	0	0	0

Dataoverdracht: de verschillende controllertypes

De gegevens die op de harde schijf staan moeten op de één of andere manier overgebracht kunnen worden naar de computer. Dit gebeurt via een interface die verbonden is met een controller. Door de jaren heen zijn er verschillende types ontwikkeld die allemaal zo hun eigen voor- en nadelen hebben.

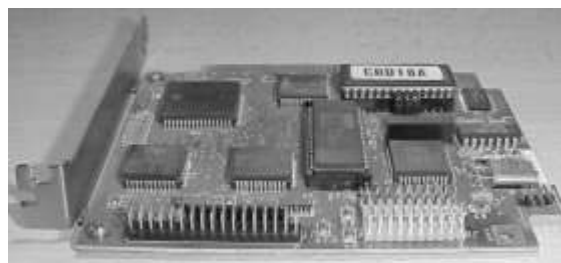
ST412/506

De eerste controllers waren van het type ST412/506. Op zo'n controller kunnen maximaal twee harde schijven worden aangesloten zodanig dat ze beiden over een gemeenschappelijke stuurkabel beschikken, maar over een aparte datakabel. Er worden dus twee flatcables per schijf aangesloten.

De snelheid waarbij deze controller gegevens kon overdragen was voor MFM codering 5Mbit/s en voor RLL codering 7,5Mbit/s. Dit was niet de maximum snelheid, maar de vereiste snelheid. Niet meer, en ook niet minder.

In deze snelheid zaten behalve de gegevens ook nog de adresmarkeringen en alle andere data. Deze data moest ook overgedragen worden omdat de controller verantwoordelijk was om eventuele foutcorrecties e.d. te doen.

In de praktijk lag de 'echte' overdracht snelheid van bruikbare gegevens veel lager vanwege de hoge storingsgevoeligheid van de datakabel. (Er moest veel gegevens opnieuw verzonden worden.)



ESDI

ESDI is eigenlijk een verbetering van de ST412/506 interface. Ze zijn daarom ook nauw verwant met elkaar. Ook de ESDI interface maakt gebruik van twee verschillende flatcables. Eén voor stuursignalen en één voor datatransport. Er kunnen echter tot maximaal 7 ESDI-loopwerken worden aangesloten.

(Voor ieder loopwerk is er op de controller dus een aparte data-aansluiting voorzien!)

Een belangrijke verbetering was de volgende:

Bij de ST412/506 interface, moest de datakabel zeer kort worden gehouden omdat men anders te veel signaalverlies leed. (hoge storingsgevoeligheid.)

Bij een ESDI interface is dit opgelost, want hier worden niet de eigenlijke gegevens zoals ze op de harde schijf staan getransporteerd, maar worden ze eerst geconverteerd naar het NRZ formaat. (Non Return to Zero)

Dit formaat is veel minder gevoelig voor storingen en kan door de controller weer makkelijk terug geconverteerd worden naar de eigenlijke data.

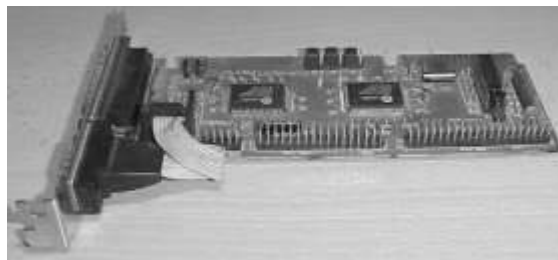
IDE, AT-bus, ATA-interface

Deze drie duiden eigenlijk één en dezelfde interface aan. IDE staat voor Intelligent Drive Electronics, hetgeen erop duidt dat de harde schijf intelligenter is dan bij de vorige interfaces het geval was.

Met intelligent wordt verstaan dat de schijf zelf de juiste maatregelen neemt bij bv. een leesfout. Vroeger moest dit door de controller afgehandeld worden.

Bij een IDE interface kunnen maximaal twee apparaten per controller aangesloten worden. Meestal zijn er twee controllers aanwezig zodat er dus maximaal 4 apparaten kunnen aangesloten worden.

Per controller moet dan één apparaat als 'master' fungeren en het (eventuele) andere als 'slave'. Voor iedere lees of schrijf operatie is een tussenkomst van de processor vereist.



SCSI

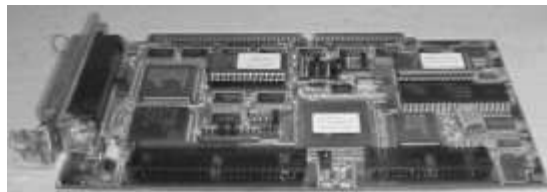
SCSI staat voor Small Computer Systems Interface. Daarmee is meteen gezegd dat SCSI bedoeld is voor PC's en andere kleine systemen.

SCSI lijkt een beetje op ESDI, er kunnen via een flatcable maximaal 8 apparaten aangesloten worden.

Let wel: De controller is zelf al een apparaat, er kunnen dus nog maar 7 andere apparaten worden aangesloten.

Ook hier zijn de loopwerken intelligent.

Het grote voordeel van SCSI tegenover IDE is dat bij communicatie tussen twee apparaten die aangesloten zijn op dezelfde SCSI bus, geen tussenkomst van de processor vereist is.



Dataoverdrachtsnelheid en toegangstijd

De snelheid van de dataoverdracht is afhankelijk van:

- De snelheid van het overdrachtsmedium (SCSI, IDE, ...).
- De snelheid waarmee de harde schijf de data kan aanbieden, die op zijn beurt afhankelijk is van:
 - De toegangstijden
 - De rotatiesnelheid van de harde schijf
 - De hoeveelheid cachegeheugen

De snelheid van het overdrachtsmedium

De snelheid van het gebruikte medium ligt zo goed als vast. Men moet in het begin dus een juiste keuze maken.

Meestal is de snelheid van het medium vele malen hoger dan hetgeen de snelste harde schijf aankan. De keuze wordt dus belangrijker als er meerdere schijven op dezelfde controler worden aangesloten en er gelijktijdig dataoverdracht moet plaatsvinden.

De keuze kan ook bepaald worden door de beschikbaarheid, de prijs, de uitbreidingsmogelijkheden, ...

De snelheid waarmee de harde schijf de data kan aanbieden

De toegangstijden

Vooraleer er data kan overgedragen worden, moet de harde schijf zijn schrijf- en leeskoppen naar de juiste plaats bewegen en wachten tot de juiste data onder de schrijf- en leeskoppen beschikbaar wordt.

Deze totale tijd is belangrijk en kan nog verder worden opgesplitst in:

- Response time
- Seek time
- Settle time
- Rotation time

Response time

De response time (reactietijd) is de tijd die de harde schijf nodig heeft om te reageren op een commando van de computer.

Deze tijd is meestal zeer klein en wordt daarom niet expliciet vermeld.

Seek time

De seek time (zoektijd) is de tijd die de harde schijf nodig heeft om zich naar het gewenste spoor te bewegen.

Hoe korter het gewenste spoor zich bij het huidige spoor bevindt, hoe kleiner deze tijd zal zijn. (De afstand is namelijk korter.)

Sommige fabrikanten geven daarom de tijd op die de schijf nodig heeft om zich één spoor verder te bewegen waardoor het lijkt alsof het een zeer snelle schijf betreft!

Kijk bij aanschaf van een nieuwe schijf altijd uit of het gaat om de gemiddelde zoektijd (Average seek time).

Settle time

Omdat de schrijf- en leeskop met een zeer hoge versnelling bewogen wordt (Ze start en stopt zeer abrupt), wordt er na het stoppen even gewacht om de schrijf- en leeskop te laten uittrillen. Deze tijd kan overlappen met de Rotation time en is zeer kort.

Rotation time

Als de harde schijf eindelijk klaar is om data te ontvangen, moet de de juiste data nog onder de lees- en schrijfkoppen voortbewegen.

Gemiddeld zal deze tijd gelijk zijn aan de tijd die nodig is om de schijf een halve rotatie te laten maken. Deze tijd hangt dus nauw samen met de rotatiesnelheid.

De rotatiesnelheid

De rotatiesnelheid speelt een rol bij de rotation time en bepaald in grote mate de dataoverdrachtsnelheid van de harde schijf. Hoe sneller de schijf ronddraait, hoe sneller de data onder de schrijf- en leeskoppen doorbeweegt en hoe sneller ze beschikbaar is.

Maar, een hogere snelheid is niet altijd noodzakelijk en kan andere nadelen met zich meebrengen zoals: Meer warmteontwikkeling, meer lawaai, meer trillingen, ...

De hoeveelheid cachegeheugen

Als de computer opdracht geeft tot het lezen van een bepaalde sector op de harde schijf, wordt meestal een volledig spoor ingelezen. Dit spoor wordt dan opgeslagen in een speciaal geheugen op de harde schijf, het cache geheugen.

Men gaat ervan uit dat de kans reëel is dat er nadien nog een sector wordt gelezen die zich op hetzelfde spoor bevindt. Als dit het geval is, hoeft er geen trage mechanische actie meer ondernomen te worden, en kan de gewenste sector vanuit het cache geheugen naar de computer overgedragen worden.

Het cache geheugen is meestal niet uitbreidbaar.

Indien er veel cache geheugen voorhanden is, kunnen er meerdere sporen ingelezen en tijdelijk bewaard worden.

Er kan zich ook iets soortgelijks, maar dan omgekeerd, voortdoen bij het wegschrijven naar de harde schijf.

Het wordt eerst in het cache geheugen gezet, en daarna (als de schijf tijd heeft) wordt het daadwerkelijk weggeschreven. Dit is dus o.a. een reden waarom men een computer eerst moet afsluiten alvorens hem uit te schakelen. Er kunnen zich immers nog niet weggeschreven gegevens in het cache geheugen bevinden!

Dissectie van een harde schijf

WAARSCHUWING:

DOE DIT NIET MET EEN HARDE SCHIJF DIE JE NOG WIL GEBRUIKEN!!!

Een harde schijf is luchtdicht en de binnenzijde is stofvrij. Na opening is ze niet meer stofvrij en zal ze niet lang meer werken (Zie schaaltekening in hoofdstuk 1.).

Indien je harde schijf kapot gaat en nog in garantie is, doe dit dan ook niet: Je bent je garantie kwijt!

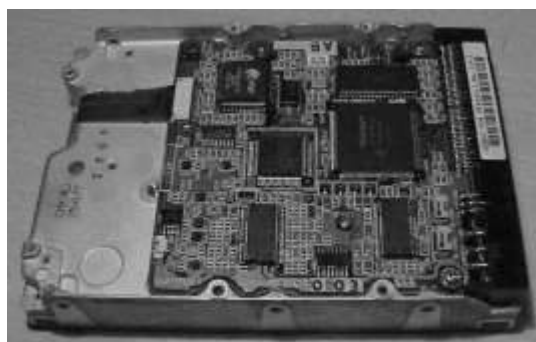
De harde schijf die gedissecteed werd, was reeds defect.

De dissectie

De harde schijf die gedissecteed werd, was een van het type 'Quantum ProDrive LPS', 402MB.

Als je ook zo'n schijf hebt, kan het interne iets afwijken, want er waren immers meerdere versies van deze schijf op de markt.

De boven- en onderzijde van de volledige schijf:



En zo ziet een harde schijf er van binnen uit.

Merk op dat het schijfoppervlak zo glad is dat het foto toestel en mijn vingers weerspiegelt.



Linksbovenaan zit een luchtfiltertje dat kleine deeltjes (stof, afgeschaapt materiaal, ...) opvangt.

Als je de tekening van de gecrashte harde schijf uit hoofdstuk 1 bekijkt, zie je dat daar de stoffilter helemaal vol zit.



Vanuit een ander standpunt, kan je duidelijk zien dat deze harde schijf bestaat uit 2 magnetische schijven. Beide schijven zijn zowel aan de boven- als aan de onderzijde beschrijfbaar. Er moeten dus 4 lees- en schrijfkoppen zijn.



Hier zie je de lees- en schrijfkop in detail evenals de actuator. (De arm waarop de koppen zitten.)



Het zwarte ding dat linksonder bij de magnetische schijf begint en dat via een zwenkpunt naar omhoog loopt, is een soort mechanische beveiliging om het magnetisch oppervlak en de koppen te beschermen.

Het werkt als volgt:

Om de schrijf- en leeskoppen te laten zweven, moet de schijf een bepaalde draaisnelheid hebben. Heeft de schijf deze draaisnelheid nog niet bereikt, en bewegen de koppen toch, dan brengt dit onherstelbare schade aan het magnetisch oppervlak en/of de koppen met zich mee. Doordat de schijf ronddraait, wordt er ook aan de aanwezige lucht een snelheid gegeven. Deze 'wind' drukt tegen deze beveiliging en pas als de windsnelheid groot genoeg is, worden de koppen ontgrendeld. Op die manier wordt er voorkomen dat de koppen reeds bewegen als de schijf nog op snelheid aan het komen is.
