

SPELEN MET GELUID

Dat er met een computer meer mogelijk is dan het oplossen van rekensommen, is inmiddels wel duidelijk. Waar nog niet zo lang geleden tekstverwerkers, rekenbladen en presentatie software het grootste deel van de harde schijf in beslag namen, vinden we nu steeds meer programma's, die de computer geschikt maken om als vermaakscentrum te dienen. We spreken van een 'multi media computer', om aan te geven, dat er andere dingen mee kunnen worden gedaan dan met een 'office computer'. De introductie van het Windows XP operating systeem heeft er sterk toe bijgedragen, dat steeds meer computergebruikers de geluids en beeldmogelijkheden van hun 'doos' gaan exploreren.

Niet alleen vinden we andere software op de multimediacomputer, ook de hardware verschilt van de officecomputer, al valt de grens tussen de twee soorten vaak moeilijk vast te stellen. Een multimedia omputer beschikt over een geavanceerde grafische kaart, die een realistische beeldweergave mogelijk maakt. Geluidskaarten voorzien de computer van surround stereo, en DVD-spelers completeren het 'thuis theater'. Naarmate onze beeld- en geluidsbestanden beter en dus groter worden, groeien de harde schijven mee. Capaciteiten van 80 Gigabyte zijn allang geen uitzondering meer. Vergeleken met de harde schijf van mijn eerste 286-machine is dat vierduizend maal groter. Vermenigvuldig ter vergelijking de oppervlakte van je huiskamer maar eens met vierduizend!

Als ons bevattingvermogen net zo snel groeide als de harde schijven, was er geen vuiltje aan de lucht, maar helaas is dat niet zo. Bevindt het internet zich al in de schemer, bij digitale beeld- en geluidstechniek tasten de meesten nog in het duister. Om begrijpelijke redenen houd ik het hoofdstuk over beeldbewerking maar voor gezien, sorry, ik kan het gebruik van beeldspraak maar niet afleren. De volgende links begeleiden je stap voor stap op weg naar het leren omgaan met de vele toepassingen van de computer op geluidsgebied.

Multimediatoepassingen vereisen een modern computersysteem. Een Pentium4, minstens 256 Mb RAM en 40 Gigabyte harddisk zijn een vereiste, om de geluidsmogelijkheden van je computer uit de verf te laten komen. Net als bij andere hobbies, is een redelijk doorzettingsvermogen, alsmede een bescheiden aanslag op je bankrekening noodzakelijk, om tot bevredigende resultaten te komen. Als het je er om te doen is, om in een uurtje een gratis professionele digitale opname studio in elkaar te knutselen, haak dan maar af. Dit is een hoofdstuk voor liefhebbers.

MUZIKALE HARDWARE

Geluidskaart

Als je dit zit te lezen, kunnen we er van uit gaan, dat je in het bezit bent van een computer en dat daar een werkend moederbord in zit. Er zijn verschillende merken moederborden, bekende namen zijn onder andere ASUS en A Open. Deze beide borden maken gebruik van de SiS chipset. Op andere borden vind je soms de VIA chipset, en zo zijn er nog wel meer. Wat heeft dat met Beatles of Beethoven te maken? Wel, tegenwoordig zijn er complete audiomodules in chipsets geïntegreerd, die de aanwezigheid van een aparte geluidskaart overbodig maken. Om er achter te komen, of jouw computer zo'n "on board" geluidschip heeft, ga je naar Control Panel en zoek je naar de geïnstalleerde geluidshardware. Hier staat dat onder "Sound and Multi Media, Eigenschappen, Hardware". Staat daar zoiets als SiS 7012 of AC'97, dan heb je beet. De drivers, die je operating systeem nodig heeft, om met de geluidsafdeling te communiceren, zijn dan meestal al geïnstalleerd. Op de achterkant van de computer vind je 3 of meer 3,5 mm gaatjes, waar je de luidsprekers, een microfoon en een externe geluidsbron op kan aansluiten. Of je met zo'n on board geluidschip gelukkig moet zijn, valt nog te bezien. De serieuze geluids-

technicus wil een Multi Channel of meerkanaals geluidskaart hebben. Om vast te stellen, of de on board chip daaraan voldoet, kunnen we een eenvoudig testje uitvoeren. Met je spraakondersteuning aan, start je WinAmp en selecteer je je favoriete MP3. Als tijdens het spelen je spraakondersteuning hoorbaar blijft, is de kans groot, dat je een multi channel chip hebt. Helemaal zeker is dit echter niet, aangezien sommige spraaksynthesizers als systeemservice draaien en dan ook via een single channel chip hoorbaar blijven. Zekerheid krijg je pas, als je een tweede instantie van WinAmp hebt gestart en nu twee MP3's door elkaar kunt horen.

Wat voor geluidchip je ook hebt, de mooiste set up is die met een extra geluidskaart. Een heel behoorlijke multi channel kaart heb je al voor zo rond de € 50,-. De Creative Sound Blaster Live Player 5.1 is zo'n kaart. Naast een prima geluidskwaliteit haal je daarmee meteen een heuse geluidsstudio in huis, met alle bekende en onbekende geluidseffecten aan boord. Met de 5.1 kan je een vrouwenstem veranderen in het geluid van de verschrikkelijke sneeuwman, of je eigen stem laten klinken als een robot, die in een concertzaal staat te jodelen. Let er bij de aanschaf op, dat je de originele CD met de Creative drivers en software erbij krijgt. Anders mag je 25 Mb gaan downloaden van de Creative website. Nadat de kaart in de computer is aangebracht, is het installeren van de drivers en software goed te doen voor een VIP.

Luidsprekers

Uitgaand van een single channel on board chip en de SB Live geïnstalleerd, heb je de optie, om twee luidspreker systemen aan te sluiten. Op de on board chip sluiten we een klein actief systeem aan, waarover we de systeemgeluiden en de spraaksynthesizer laten horen. Op de SB Live wordt een systeem met sub woofer van Altec Lansing of Creative aangesloten. Deze 'surround stereo' systemen bestaan uit een sub woofer, die de lage tonen voor z'n rekening neemt. Deze speaker kan uit het gezicht, bijvoorbeeld ergens op de grond, worden geplaatst. Lage tonen bevatten geen stereo informatie, dus de precieze plaatsing is niet kritisch. De twee of vier satellite speakers worden in een links rechts opstelling geplaatst, waarbij het handig is, een geluidsbestandje bij de hand te hebben, waarop het verschil tussen links en rechts te horen is. Bij de SB Live software wordt dit bestandje meegeleverd, maar je kan het ook van BlinfoTec downloaden (zie hieronder). Als de originele snoeren worden gebruikt, dan is de fase altijd correct. Als de speakers in een stereo opstelling 'uit fase' zijn aangesloten, is dat te horen door op de ideale plaats tussen en voor de speakers plaats te nemen, en te luisteren, of het makkelijk is, om de richting van bijvoorbeeld stemmen en instrumenten te kunnen bepalen. Als een enkele stem of instrument uit alle richtingen lijkt te komen, dan staan de speakers 'uit fase'. De kwaliteit van het speakersysteem is bepalend voor de totale geluidskwaliteit. Hier staat een systeem van rond de € 100,- en ik ben vrij kritisch over geluidskwaliteit.

CD-speler en -brander

Om je huisstudio compleet te maken, heb je een CD/DVD-speler en CD/DVD-brander nodig. Bij oudere computers moest je die nog apart kopen en in (laten) bouwen. Tegenwoordig worden de meeste computers inclusief speler en brander geleverd. Op het moment van dit schrijven is het nog steeds niet duidelijk, welke standaard uiteindelijk de 'globale' standaard zal worden. Je kan er echter van uit gaan, dat de meeste spelers en branders die je nu koopt, bruikbaar zijn voor geluid. Ik heb verschillende merken geprobeerd, maar heb nooit uit kunnen maken, welk merk het beste was. Van twee dezelfde spelers gaf de ene het na zes maanden op en draait de andere nu, twee jaar later, nog steeds zonder problemen. Geen peil op te trekken. Vraag bij aankoop, of de CD brander het programma 'CloneCD' ondersteunt. Als dat zo is, dan zal je nooit problemen ondervinden bij het maken van een back up van een CD.

Je operating systeem wijst een drive letter toe aan de beide CD stations, dus als je twee harde schijven hebt, C en D, dan worden E en F de CD stations. Let op dat er voldoende koeling aan-

wezig is, want CD stations zijn, zeker tijdens intensief gebruik, echte kacheltjes!

Als je nog een verzameling LP's en of cassettes hebt, en je zou die opnames ook graag netjes geordend op de harde schijf willen hebben staan, dan moet er nog een ding gebeuren. Van een voorversterker, waarop een platenspeler, een cassette speler, een CD-speler en een tuner kan worden aangesloten, moet de lineuitgang aangesloten worden op de lineingang van de geluidskaart. Hierbij kan de platenspeler soms voor problemen zorgen, omdat het uiterst zwakke signaal van het MD-element nogal gevoelig is voor storing van buitenaf. De door de computer uitgestraalde velden kunnen dan een probleem vormen. Experimenteren met de plaatsing van de componenten, en de aanwezigheid van goede aarding kan vaak voor de oplossing zorgen. Deze problemen heb je niet met een USB-platenspeler.

DE THEORIE BIJ DIGITALE GELUIDSBEWERKING

Voor wie nog meer wil weten over de theorie van digitaal geluid, bits en bytes, compressie, mp3, platenspelers en snoeren met pluggen. Dit verhaal gaat over geluid. Digitaal geluid, in al of niet gecomprimeerde vorm. Aangezien digitaal geluid met getallen te maken heeft, zullen we geregeld kleine rekenvoorbeeldjes moeten gebruiken om te begrijpen hoe het in elkaar steekt. Het is natuurlijk wel de bedoeling, dat we alles allemaal netjes kunnen blijven volgen. Wiskundig zal het dus niet worden en alles komt stap voor stap.

We gaan ons een beeld vormen van geluid, van het begin tot aan het einde. Dat wil zeggen: vanaf het moment dat het geluid gemaakt wordt en dus ontstaat, tot het moment dat het op je harde schijf is opgeslagen. De weg terug bekijken we natuurlijk ook. Onderweg komen we zijdelings op machtsverheffen en het leek me zinvol om die kreet ook even uit te leggen, bij geval je er niet bekend mee bent. Daarnaast gaan we in op het idee achter audio compressie. Waarom is het en wat is het globale principe erachter. Daarna kijken we naar de mp3 file, omdat die enorm veel gebruikt wordt en op internet enorm populair is. We zien dan dat een mp3 meer bevat dan geluid alleen. Vervolgens bespreken we hoe je geluid kunt opnemen. We gaan in op de Windows mixer en bevelen een paar audioprogramma's aan, met wat algemene procedures.

Bits en bytes: de basis

Eerst iets over bits en bytes. Een bit is een hardwarematig elementje in het geheugen van je computer, dat spanning of geen spanning heeft. Spanning leggen we uit als: "het bit heeft waarde 1", geen spanning geeft bitwaarde 0. Acht bitjes bij elkaar heet een byte. (Vroeger schreef je overigens niet "byte", maar "bite", voor "BInary Term".)

In een byte kun je, met de acht bitjes die er daar zijn, van alles opslaan. Een paar voorbeelden om dat uit te leggen. Volgens een bepaalde rekenmethode kun je een gewoon getal, bijvoorbeeld 115, omzetten in een combinatie eentjes en nulletjes, zodat je ze in een geheugenbyte kunt opslaan. Natuurlijk is daar een bovengrens aan, omdat je in een byte maar acht bitjes ter beschikking hebt. Neem je niet 1 maar 2 of meer bytes geheugenruimte, dan kun je daar ook hele grote getallen in opslaan.

Zou je een gewone letter een bepaald nummer geven, dan kun je via het opslaan van nummers ook teksten opslaan in een serie bytes. De meest gangbare codering waarbij letters van het alfabet, leestekens en cijfers een nummertje krijgen om opgeslagen te worden heet ASCII. Dit wordt uitgesproken als askie. Sommigen noemen het ask twee, anderen zeggen eskie, maar de juiste naam is askie, met twee i's op het eind. Voor dit verhaal is dat verder niet zo interessant, maar bij geval je benieuwd bent: ASCII staat voor American Standard Code for Information Interchange. Ook kun je er, volgens een bepaalde indeling en rekenkundige manier van opslag, een getal in zetten met een drijvende komma. Een getal als 3.141592654 (toevallig de waarde van het getal pi) kun je zodoende ook al kwijt in een paar bytes bij elkaar. Gehele getallen, getallen met drij-

vende komma en tekst, het kan allemaal worden opgeslagen in bytes; Elementjes in het geheugen van je pc die 8 kleine condensatortjes bevatten waar al dan niet spanning in gezet is.

Beeld kun je ook opslaan in bytes. Beeld is opgebouwd uit horizontale lijnen, waarvan er nogal wat op je beeldscherm passen. Elke lijn is weer ingedeeld in kleine stukjes. 1 Zo'n stukje is een beeldpuntje op het scherm, een pixel genoemd. Een pixel is er wel of hij is er niet. Pixels die aan staan vormen samen het zichtbare beeld op het scherm. Als je dus een georganiseerde manier kunt bedenken om pixels die aan staan als eentjes, en pixels die uit staan als nulletjes netjes bij elkaar in een serie bytes te stoppen, kun je ook beeld opslaan. Als je kleuren een nummer geeft kun je ook die in digitale vorm bewaren.

Geluid leent zich er eveneens voor om opgeslagen te worden in bytes. We zullen later zien hoe dat werkt, want ook geluid laat zich vertalen in eentjes en nulletjes.

Je ziet dat een bit, zo'n klein dingetje waar spanning of geen spanning in zit, een erg belangrijk dingetje in je computer is. Acht bitjes bij elkaar heet een byte zoals gezegd, en als je maar genoeg bytes bij elkaar hebt, kun je bijna alles opslaan wat je maar kunt bedenken: teksten, foto's, tekeningen, muziek, geluid, spraak, rekentabellen en ga zo maar door. In je computer zit een vreselijk grote hoeveelheid bytes. Tegenwoordig wel een miljard of nog meer. Omdat we niet graag in honderden miljoenen spreken, zijn er naast bytes ook kilobytes, megabytes, gigabytes, terabytes en exabytes bedacht.

Een kilobyte, afgekort KB, is van oudsher niet 1000, maar 1024 bytes. Wij mensen kennen het decimale (dus tientallig) stelsel met cijfers 0 t/m 9, maar de computers kennen enkel het tweetalig stelsel met 2 cijfers: de 0 en de 1. Een enkel bitje dat 0 of 1 is is niet zo erg zinvol. Het combineren van grotere groepen nullen en enen maakt pas dat je er zinvolle dingen mee kunt doen. Om de computer met ons tientallig stelsel te laten rekenen, is heel wat werk nodig.

De processor zelf werkt namelijk met machten van 2, en dat is voor het optellen van bijvoorbeeld 9 plus 10 niet zo erg handig om mee te moeten werken. Toch gebeurt dat, omdat de processor het binaire systeem gebruikt. Anders zou die, qua wiskundige hardware, te complex moeten worden. De omzetslag van een simpele optelsom voor ons vanuit het binaire stelsel is een hoofdstuk apart, wat we hier maar fijn laten zitten. Het blijft gaan om machten. Omdat er maar twee cijfers zijn waaruit het talstelsel waarmee de processor werkt is opgebouwd, gaan alle interne berekeningen in de processor met machten van 2. Daar zit veel meer achter, maar onthoud dit voorlopig gewoon als feit. Machten van twee liggen aan de basis van alle rekenkundige bewerkingen van de processor.

Machtsverheffen

Machtsverheffen is het op een rij zetten van steeds hetzelfde getal, met vermenigvuldigingstekens ertussen. Een getal kun je tot een bepaalde macht verheffen en daar komt dan iets uit. 2 tot de macht 3 schrijf je uit als drie keer het getal twee achter elkaar, met vermenigvuldigingstekens ertussen. Dus: 2 keer 2 keer 2. En dat is acht. Dus: twee tot de macht drie is acht. Twee tot de macht vier, ook wel twee tot de vierde genoemd, is 16. Twee Maal 2 maal 2 maal 2. En zo kun je door gaan. Twee tot de vijfde zijn 5 tweeën op een rij met vermenigvuldigingstekens ertussen en dat reken je dan uit. Maar, er zijn een paar rare dingen met machtsverheffen. Een getal tot de macht 1, hoe moet dat dan? Je hebt toch minimaal twee getallen nodig om te kunnen vermenigvuldigen. Maar bij iets tot de macht 1 loop je al vast. Hoe nu verder? Wel: er is een simpele afspraak gemaakt. Die zegt: elk getal tot de macht 1 is gelijk aan zichzelf. Dus: 4 tot de macht 1 is 4, 6 tot de macht 1 is gewoon 6. Je kunt een getal ook tot de macht 0 verheffen. Logischerwijs zou dat helemaal niet kunnen, maar de wiskunde kent de afspraak dat een getal tot de macht 0, altijd 1 is. 10 Tot de nulde of 2 tot de nulde, het is gewoon 1. Samenvattend: een getal tot de macht 0 is altijd 1. Een getal tot de macht 1 is bij afspraak altijd gelijk aan zichzelf. Een getal tot de macht twee of hoger is een kwestie van blijven vermenigvuldigen.

Kilobytes en megabytes

2 tot de macht 10 is 1024. Vandaar dat het logisch is gebleken, om 1 kilobyte niet gelijk te stellen aan 1000 bytes, maar wel aan 1024 bytes. Intern voor de processor maakt dat het rekenleven een stuk eenvoudiger. Daarom is 1 megabyte ook niet precies 1 miljoen bytes, maar iets meer. Omdat 1 kilobyte gelijk is aan 1024 bytes, is 1 megabyte weer gelijk aan 1024 maal 1 kilobyte. Dus: 1024 maal 1024 is 1048576 bytes. De letters k, m en b hebben, afhankelijk van of je ze als hoofd- of kleine letter schrijft, een andere betekenis. Kleine b staat voor bit, hoofdletter B staat voor byte. Kleine k weet ik niet meer, grote K is Kilo. Kleine m is mili, grote M is mega. Mb is dus megabit, MB is megabyte. Zo verder redenerend is 1024 megabyte weer gelijk aan 1 gigabyte, en 1024 gigabyte is vervolgens een terabyte. Daarboven is nog de exabyte en dan houdt de praktijk wel zo'n beetje op. Omdat computers tot op heden nog altijd met machten van 2 werken, is aan heel die boel nog altijd niets veranderd. Dit wetend kunnen we een stap verder.

Digitaal geluid: de basis

Hoe sla je dan audio op? Wel. Daar zijn vele manieren voor bedacht, maar de aller oudste en meest simpele manier is pcm, pulse code modulation. In Windows kretologie heet de lichtelijk opgefluchte vorm daarvan een wav file. De letters W A V staan voor "wave", geluidsgolf. Geluid maak je door lucht in trilling te brengen. Roep maar eens met je hand op 3 centimeter van je mond. Dan voel je de lucht trillen. Je oor is echter veel en veel gevoeliger dan je hand, en kan trillingen in de lucht op kilometers afstand waarnemen. Geluid is dus lucht, waar zich een trillingsgolf doorheen beweegt. Die vibreert tegen je trommelvlies en je oor vertaalt dat door een ingewikkeld systeem in je hoofd in iets, dat je hersenen begrijpen als geluid.

Nu is een speaker in feite niks meer dan een groot oppervlak dat in trilling gebracht wordt, om zodoende de audio informatie via de trillende lucht in je oor te brengen. Midden achter de conus van de speaker zit een magneet die de conus naar zich toe trekt. De conus kan vrij bewegen ten opzichte van de magneet. Hij wordt bewogen door spanning op de spoel achter de magneet te zetten. Hoe meer spanning, hoe verder de conus naar buiten komt, in weerwil van de aantrekking van de magneet. Haal je de spanning echter weg, dan gaat de conus weer naar ruststand, doordat de magneet de conus naar zijn centrum trekt. Geluid uit een speaker maak je dus, door de spanning op de spoel heel snel te variëren. Beetje meer, beetje minder, nog minder, even maximaal, even geen spanning, halverwege, en zo voort in een ogenschijnlijk willekeurig patroon. Daardoor beweegt de conus van de speaker, die brengt de lucht in trilling, de golf plant zich voort tot in je oor en je hoort het geluid. Dat is ruw weg het idee.

Digitale audio werkt nu als volgt. Bij het opnemen van het geluid van bijvoorbeeld een microfoon, wordt de conus van de microfoon bewogen door het in de ruimte aanwezige geluid. Daardoor ontstaat er spanning in het spoeltje, dat bij de magneet van de conus in het microfoontje zit. Die spanning gaat door het draadje van de microfoon naar, laten we zeggen, je minidisk recorder of je geluidskaart. En daar komt de truc. Ongeveer 44 duizend keer per seconde staat daar een chipje te meten, wat op elk moment de spanning is die uit je microfoon komt. Elk zo'n meting noemen we een monstertje, een sample. De chip die de metingen verricht heet de A-D converter (analoog -> digitaal).

Die spanning uit je microfoon heeft natuurlijk een minimum, helemaal niks, en hij heeft een maximum, namelijk zo veel als er maar opgewekt kan worden in het spoeltje. Omdat de A-D converter een analoog signaal in zich krijgt maar een digitaal signaal zal moeten uitspugen dat uit nullen en enen bestaat, moet er een systeem worden bedacht om een willekeurige spanning ergens tussen minimum en maximum, uit te drukken in een reeks nullen en enen. Met 8 bitjes die allemaal onafhankelijk van elkaar 0 of 1 kunnen zijn, kun je 256 verschillende combinaties maken. De binaire combinatie "00000000" is gelijk aan ons gewone getal 0. De binaire combinatie "11111111", (acht binaire eentjes op een rij), is voor ons gelijk aan het normale (decimale) getal

255. Daarom zijn er 256 mogelijkheden met acht bitjes, namelijk mogelijkheid 1 t/m 255, en alles op 0, waarmee het totaal aan mogelijkheden op 256 komt. Acht bitjes is zoals gezegd 1 byte. In 1 byte past dus een getal, tussen 0 en 255 inclusief.

Nu bepalen we dat er tussen minimum en maximum, 256 stapjes bestaan. Helemaal geen spanning (binair "00000000"), een heel klein beetje spanning ("00000001"), nog wat meer spanning ("00000010") en ten slotte maximale spanning, "11111111", en dat is dan combinatie nummer 255 in ons normale stelsel.

44 Duizend keer per seconde wordt er door de A-D converter spanning van het audiosignaal gemeten, en 44 duizend keer per seconde wordt er dus een groepje van acht nullen en enen (1 byte) door de A-D converter aan je computer gegeven. Elke byte heet een sample. Elke sample, waar we er zo'n 44 duizend per seconde van krijgen, is een getal tussen de 0 en 255, maar dan verhaspeld in een binaire notatievorm, acht nulletjes en eentjes. Met andere woorden: elke digitale sample, opgebouwd uit nulletjes en eentjes, representeert een bepaalde analoge spanning. Je pc slaat alles vervolgens netjes op in je harde schijf. Dat is de hele weg van geluid tot file op je hard disk. Terug is nu eenvoudig. De pc stuurt 44 duizend keer per seconde een byte (8 bitjes) naar je geluidskaart. Daar zit een tweede chipje, de D-A converter, die de digitale data stroom moet omzetten naar analoog, en dus hoorbaar te maken geluid. Bij elke byte die wordt gelezen, wordt het bij die toevallige binaire combinatie horend spanningskijje naar je speaker gestuurd. Krijgt de D-A converter acht eentjes, dan zal ie maximale spanning naar je speaker spoel toe sturen waardoor de conus maximaal naar je toe gehaald wordt. Krijgt de D-A converter acht nullen, dan stuurt ie helemaal geen spanning naar de speaker, waardoor de magneet de kans krijgt de conus weer in ruststand terug te trekken. Alle binaire combinaties tussen acht nullen en acht enen in, levert een spoelspanning op die ergens tussen min en max in ligt. Zo maakt de pc het digitale opgenomen geluid weer hoorbaar door je speaker.

Digitaal geluid: de werkelijkheid

Nu moeten we de zaak wat nuanceren, om tot de werkelijkheid te komen. De sampling frequentie is niet precies 44 duizend, maar wel 44 duizend 1 honderd maal per seconde. Waarom nou precies die 1 honderd nodig was kan ik ook niet verklaren maar het is nu eenmaal de wereldwijd aanvaarde standaard geworden. Hoe vaker per seconde je een sample van een op te nemen signaal trekt, hoe nauwkeuriger je het spanningsverloop vast legt en hoe getrouwer de uiteindelijke weergave zal klinken. Hoe lager de sample frequentie, hoe slechter de audio kwaliteit. 44100 is de door Philips bedachte standaard, voor cd kwaliteit.

Dan: met 1 byte heb je maar 256 combinatiemogelijkheden om spanningen op te slaan tussen minimum (0) en maximum (255). Een beetje weinig voor zoiets ragfijns als audio spanningen. Neem je niet 1 maar 2 bytes per sample, dan heb je geen 8 maar wel 16 bitjes ter beschikking. Nu zou je denken dat daarmee het aantal mogelijkheden precies verdubbelt, omdat je niet 1 maar 2 bytes ter beschikking hebt. Maar dat is niet waar. Omdat we met machten van 2 werken, is het veel meer dan dat.

Met 8 bitjes had je 256 mogelijkheden, maar met 9 bitjes heb je er al 2 maal zo veel, dus 512.

Met 10 bitjes heb je dan al 1024 mogelijkheden en als we wat stapjes overslaan heb je met 16 bits 65536 mogelijkheden. Dat is geen toeval, want 2 tot de macht 16 is immers 65536.

Het spectrum tussen geen spanning (0) en maximum spanning (65535) is nu heel wat groter. De nuances in de spanningswisselingen kun je dus veel getrouwer opslaan als je meer digitale stapjes creëert, tussen spanning min en max. Daarom samplet men op cd kwaliteit niet met 8 bits, maar met 16 bits per sample. Dat zijn 2 bytes per sample. Het aantal bits per sample wordt in de muziekindustrie de audio resolutie genoemd. Daarbij gaat het dus om het aantal bits per sample, niet om het aantal samples per seconde.

En dan: stereogeluid bestaat niet uit 1, maar uit 2 kanalen. Links gesampled en rechts ook. Dat

betekent, dat er 2 A-D en ook 2 D-A converters moeten zijn, die tegelijkertijd hun werk doen. En dat betekent weer, dat er per seconde 2 keer zoveel data uit komt. Een stroom voor het linker kanaal, en eentje voor het rechter.

Samenvattend hebben we nu 44100 samples per seconde per audiokanaal, elk van 2 bytes breed. Dat is 44100 maal 2 bytes aan data, dat maal 2 kanalen, is 176.400 bytes, (zo'n 172 kilobyte) aan wav data per seconde. Dat geldt dus als je samplet op 44.1 kilohertz, met een resolutie van 16 bits per sample en stereo signaal. In de uiteindelijke stereo wav file wordt om beurten een sampletje voor het linker kanaal en dan eentje voor het rechter kanaal opgeslagen. Dat beurtelings opslaan van data voor het linker en dan voor het rechter kanaal heet interleaved audio.

Per minuut is dat slordig gerekend zo'n 10 megabyte data voor cd kwaliteit audio. 10 meg per minuut wordt al gauw heel groot.

Audiocompressie

Comprimeren, hoe gaat dat? Laten we eerst eens vaststellen wat comprimeren eigenlijk is. Comprimeren is het kleiner maken van een bestand met data, op zo'n manier dat het gecomprimeerde bestand later terug gerekend kan worden naar het originele, grote bestand.

Audio in normale, ongecomprimeerde vorm op harde schijf, bestaat op Windows pc's meestal in de vorm van .wav bestanden. Gecomprimeerde bestanden hebben echter allerlei verschillende extensies. Een ervan is mp3. Een wav bestand van 1 minuut zou 10 megabyte groot zijn. Dat hebben we eerder gezien. Maar als we het gaan comprimeren naar mp3, dan kan hetzelfde stukje geluid tot een tiende worden teruggebracht. Dus geen 10, maar 1 megabyte. En zelfs nog veel kleiner.

Wav bestanden kun je gewoon met de Windows media player laten afspelen. Mp3 bestanden werken op dezelfde manier. Een programma als winamp kun je installeren om die taken over te nemen van media player, bijvoorbeeld omdat winamp een stuk makkelijker te bedienen is. Maar hoe werkt het dan, dat comprimeren? Wel. Het is in werkelijkheid een enorm complex wetenschappelijk verhaal, dat we gelukkig kunnen terug brengen tot een heel simpel basis principe. Stel je eens een heel simpel audiosignaal voor. Een toon van 4000 hertz. Daar doorheen horen we een toon van 4003 hertz. Alleen is het volume van die vlakbij liggende toon toevallig zo zacht, dat wij het niet horen. Of we nu de 4 kilohertz toon horen met of zonder zachte bijtoon, ons oor merkt geen verschil. Het opslaan van 2 tonen in een gecomprimeerd bestand kost echter meer schijfruimte dan het opslaan van 1 toon. Daarom hoeft de zachte toon van 4003 hertz niet te worden opgeslagen, want of die er nou wel of niet is hoort niemand.

Een praktijkvoorbeeld om dit principe te illustreren. Stel, ergens buiten hangt een wind gong. Dat is een ding dat mensen voor de lol wel eens buiten hangen. Het zijn dunne, metalen staafjes, die zachtjes tegen elkaar aan tinkelen als de wind er langs blaast. Behalve het serene geluidje heeft een wind gong geen nuttige functie.

Als het stil is, hoor je de windgong met de ramen dicht binnen nog heel zachtjes tinkelen. Maar als de stofzuiger aan gaat, hoor je de windgong niet meer. Hij weerklinkt nog wel, maar je hoort hem niet meer omdat de stofzuiger hem overstemt. Dit principe heet "sound masking", gemaskeerd geluid. Het ene overstemt het andere.

Zo werkt het nu ook met opgenomen digitale audio, die je comprimeert. Het ene signaal overstemt het andere, waardoor het andere dus in feite kan worden weggegooid omdat niemand het ooit horen zal en dat scheelt ruimte in je gecomprimeerde bestand.

Daarom is een studie gemaakt van de perceptie van onze oren, zodat er kon worden bepaald wat wij wel en wat we niet meer horen. Op basis daarvan werkt compressie. Frequenties die wel in de audio zitten en dus ruimte innemen, worden op volume en op omliggende geluiden getest om te zien of ze weg mogen. Hoe minder frequenties je gaat op slaan, hoe kleiner je files gaan worden.

Hoe meer je weg gooit, hoe minder je hoeft op te slaan. Maar op een zeker moment ga je zo veel weg gooien om kleinere files te krijgen, dat je de aantasting van de audio weldegelijk gaat horen.

Rekenen aan compressie

Een bitrate is bij veel gecomprimeerde audioformaten een maatstaf voor compressie. 32 kilobit betekent: de gecomprimeerde audio file zal 32 duizend bitjes per seconde nodig hebben om af te spelen. Een rekenvoorbeeldje maar weer. 32 kilobit is, even voor het gemak, 32000 bits per seconde. 32000 bitjes gedeeld door 8 bits in een byte levert 4 duizend bytes voor een seconde gecomprimeerde audio, op een bitrate van 32 kilobit. Per minuut is dat: 4000 bytes maal 60 seconden is 240.000 bytes per minuut. Dat gedeeld door 1024 is ongeveer 234 echte kilobytes data per minuut. Samengevat: 32 kilobit mp3 geluid komt neer op ongeveer 234 kilobyte data per minuut. Je snapt dan wel, dat je voor een minuut muziek met 128 kilobit iets meer data per minuut gaat krijgen. We kunnen dat eenvoudig uitrekenen:

128000 bitjes per seconde maal 60 seconden voor een minuut, is 7.680.000 bitjes nodig per minuut. Dit grote aantal bits per minuut moet je delen door 8 om op het aantal bytes per minuut te komen. De uitkomst is dan 960.000 bytes per minuut. Om daar echte kilobytes van te maken deel je dat weer door 1024, is ongeveer 938 kilobyte. Samenvattend: geluid op 128 kilobit kost geen 234, maar ongeveer 938 kilobyte per minuut. Dat is bijna 1 MB. 128 Kilobit comprimeert dus een factor 10. Immers: een wav bestand op normale grootte kost je 10 megabyte schijfruimte per minuut, terwijl 128 kilobit mp3 nog geen 1 MB nodig heeft. Tien keer zo weinig.

En dat klopt dan allemaal weer prachtig met elkaar. Want 32 kilobit en 128 kilobit scheelt een factor 4. Er gaat per slot van rekening 4 keer 32 in 128. En dus is het aantal benodigde bits per seconde ook vier keer zo groot voor 128 kilobit, ten opzichte van 32 kilobit.

Hoe lager de bitrate, hoe meer geluid er wordt weggegooid en hoe kleiner de resulterende mp3 gaat worden. Dat hebben we al gezien. Moet je in een mp3 file op 128 kilobit nu slechts 1 kanaal kwijt omdat je een mono opname naar mp3 aan het omzetten bent, dan is dat prachtig. Want je kunt elke seconde de volledige 128 duizend bits gebruiken om de audio van dat ene kanaal op te slaan en je hoeft dan dus niet enorm strak te comprimeren, niet al te veel frequenties weg te gooien en dat levert uiteindelijk een prima klinkend gecomprimeerd eindresultaat.

Maar moet je binnen die 128 duizend bitjes voor 1 seconde audio 2 kanalen kwijt, dan betekent dat dat er 64000 bitjes voor links en 64 duizend bitjes voor rechts moeten worden gebruikt om de audio in 128 kilobit per seconde te laten passen.

Conclusie: 128 mono klinkt prima, 128 stereo is in feite 2 x 64 kilobit mono tegelijkertijd afgespeeld en klinkt dus hoorbaar minder goed. Simpelweg omdat er bij stereo twee keer zoveel audio moet worden opgeslagen, in de ruimte die bij mono volledig voor 1 kanaal mocht worden gebruikt.

Opslagruimte en bandbreedte

Op dit moment van schrijven leven we bijna aan het begin van het jaar 2005. Een paar jaar geleden was het nog nodig om mp3 bestanden zo klein mogelijk te maken. Iedereen had nog een gewoon telefoon modem en dus was het duurder om grotere bestanden te versturen. Daarbij waren de harde schijven nog niet zo groot als nu, en je kon zelf nog geen cd's branden.

Tegenwoordig echter hebben heel veel mensen breedband internet, grote schijven en een cd of dvd brander. Het is dus echt niet meer nodig mp3 bestanden echt klein te maken en dat is maar goed ook. Want hoe lager de bitrate van mp3, hoe slechter de resterende audio kwaliteit is. Moge het oude misverstand hiermee eindelijk uit de weg worden geruimd: comprimeren is niet alleen een kwestie van kleine bestanden willen hebben. Het kiezen van de juiste bitrate moet altijd een afweging zijn tussen hoorbaar kwaliteitsverlies aan de ene kant, en omvang van het bestand aan de andere kant. Daarbij weegt kwaliteit mijns inziens veel zwaarder dan bestandsomvang.

Mijn advies is dan ook als volgt. Voor mono materiaal, zou je het best een bitrate van 128 kunnen aanhouden. Voor stereo materiaal 256. Ik weet het, op internet zwerven genoeg mp3tjes rond op 128 kilobit die stereo zijn en die klinken toch ook goed? Ja en nee. Ze klinken acceptabel, maar in vergelijking met het origineel voordat de muziek in mp3 gecodeerd werd, is 128 kilobit stereo beslist hoorbaar beneden cd kwaliteit.

Voor de duidelijkheid: een wav-bestand kan mono of stereo zijn en dan heb je niets te maken met bitrates. De kwaliteit van de wav hangt af van de sampling frequentie. 44100 samples per seconde klinkt goed. Daarnaast bepaalt de resolutie de audiokwaliteit van de wav. 8 Bit is telefoonkwaliteit, 16 bit cd kwaliteit en tegenwoordig is het niet bijzonder meer om in studio's 24 en 32 bit opnames te maken. Voor ons is 16 bit prima. Tijdens het maken van de wav moet je bepalen of die mono (1 kanaal) of stereo (2 kanalen) moet worden. Als je geluidsbron mono is, is het onzin om in stereo op te nemen. In het beste geval ontvangt je computer dan hetzelfde signaal zowel links als rechts en daarbij is het onnodig twee keer hetzelfde op te slaan. Is je bron waarvan je opneemt echter stereo, dan is het jammer om de stereo-effecten uit de audio verloren te laten gaan door de wav mono op te nemen. Luister dus naar het origineel dat je in mp3 gaat omzetten en beoordeel aan de hand daarvan of de wav mono of stereo moet worden. Is die keuze eenmaal gemaakt, dan is het een kwestie van een eenvoudige instelling in je digitale opname programma. Daarentegen bepaalt bij een mp3 bestand enkel nog de bitrate de kwaliteit van het uiteindelijke geluid. Een mp3 is immers een wav-bestand geweest dat naar mp3 is geconverteerd. De sampling frequentie, het aantal kanalen en de resolutie liggen dus al vast omdat de wav al klaar is als je gaat mp3en.

Opbouw van een mp3-bestand

Ziezo. Nu we weten hoe een mp3-file in elkaar zit waar het de audio compressie betreft, kunnen we weer een stapje verder. Want een mp3-file bevat meer dan audio alleen. Er zit een tag in, en er zijn diverse dingen die nuttig zijn om te weten als je met mp3 werkt.

Een tag is in audioverband een brokje extra informatie die aan het mp3-bestand wordt toegevoegd. Geen geluid, maar andere randinformatie zoals naam van de uitvoerend artiest, naam van het liedje, jaar van productie, naam van het programma waarmee het mp3tje gemaakt werd en zo voort. De tag is bedacht omdat mensen nu eenmaal graag grote verzamelingen mp3tjes aanleggen. En als je een grote verzameling hebt, zelfs al geef je je bestanden zinvolle namen, dan is het nog steeds bijna ondoenlijk om al die extra informatie in de bestandsnaam zelf te zetten. Allerlei mp3 afspeelprogramma's lezen de tags uit de mp3 bestandjes en kunnen daaruit een catalogus opbouwen, waar je vervolgens weer doorheen kunt bladeren om je verzameling te sorteren op jaar, genre of wat je maar wil. Voorwaarde is natuurlijk wel, dat al die informatie in de mp3 tags is opgenomen. Op het moment dat je met je encoder het mp3tje maakt, bepaal je wat er in de tag moet komen te staan. Natuurlijk zijn er ook eenvoudige programmaatjes waarmee je achteraf, nadat het mp3tje klaar is, de informatie uit de tag nog kunt veranderen. Naar ik hoop is daarmee weer een raadsel rond mp3 opgelost. Een tag is niets moeilijks; Gewoon wat men noemt non-audio data in het mp3 bestandje.

Andere gecomprimeerde audioformaten

Natuurlijk is mp3 niet de enige speler in de audio compressie markt. Veel voorkomende alternatieven zijn o.a. .ogg, .wma, .vqf en tegenwoordig ook .flac. Je kunt je afvragen welk formaat je het beste kunt gebruiken. In principe is die keuze helemaal aan jezelf. Er is amper iets zinnigs over te zeggen. Wel is mp3 het meest gebruikte formaat en je kunt er zeker van zijn dat mp3 afspeelbaar is op veel moderne cd walkmans, dvd spelers en andere audio hardware. Ogg comprimeert weer beter dan mp3, maar het formaat is lang niet zo bekend en dus niet zo uitwisselbaar als mp3. Uit eigen ondervinding heb ik gemerkt dat ogg verbazingwekkend goed klinkt op lage

bitrates als je dat vergelijkt met mp3. Een waarschuwing is op dit moment wel aan de orde. Onthoud goed dat mp3 verlies in kwaliteit betekent, evenals ogg, wma en vqf. Het mp3 formaat kiest bepaalde frequenties om uit de audio weg te gooien, terwijl ogg, wma en vqf weer andere dingen uit het audio frequentiespectrum weg laten om de wav volgens hun eigen methode te comprimeren. Ga je een bestand dus omzetten vanuit het ene naar het andere gecomprimeerde formaat, dan wordt het verlies in audio kwaliteit, ten gevolge van de compressie, de optelsom van de verliezen voortvloeiend uit beide formaten. Heb je dus een wav waar je een mp3 van maakt, om die mp3 een tijdje later om te zetten naar bijvoorbeeld ogg, dan ontbreekt er in de uiteindelijke ogg het frequentiedeel dat door de mp3 encoder is weggelaten, en ook het deel dat door de ogg encoder is gekozen. Dat betekent dat het eindresultaat in ogg lang niet zo goed klinkt, als wanneer je de wav direct in ogg had omgezet. Het converteren tussen compressieformaten onderling is dus beslist niet aan te raden, al struikel je op internet over programma's die deze functionaliteit aanbieden. Als het resultaat vervolgens slecht klinkt, legt de eindgebruiker de schuld vaak bij het uiteindelijke audio formaat, terwijl degene die heeft zitten rommelen met compressie formaten in feite de persoon is die het kwaliteitsverlies op zijn geweten heeft.

Verliesloze audiocompressie

De voor- en nadelen van audio compressie zullen nu duidelijk zijn. De bestanden worden beduidend kleiner, met alle positieve gevolgen van dien. Daar tegenover staat, dat het gecomprimeerde geluid bij formaten als mp3 en zijn concurrenten altijd door de compressie zal worden aangetast. Afhankelijk van de mate van compressie al of niet hoorbaar.

Maar het kan ook anders. Er bestaan ook gangbare audio bestandsformaten , waarbij het gecomprimeerde resultaat beduidend kleiner is dan het originele wav bestand, terwijl de kwaliteit volledig behouden blijft. Het behoeft geen betoog dat het altijd uit de lengte of uit de breedte zal moeten gaan, dus ook aan verliesloze audio compressie zit weer een duidelijk nadeel, en dat is dat de resulterende, gecomprimeerde file weliswaar kleiner wordt dan de wav, maar lang niet zo klein zal worden als het geval is met verliesgevende compressie als mp3.

Toch loont het beslist om er je tijd aan te spenderen. Een formaat als flac is vandaag de dag erg in opkomst. Er zijn wel concurrenten, maar die zijn lang zo goed niet waar het de compressiefactor betreft. Flac wordt ontwikkeld door dezelfde lui die ogg hebben gemaakt. Ogg is wel "lossy", verliesgevend, maar flac is "lossles", verliesloos. Flac comprimeert ongeveer tot de helft. Heb je een wav-file van 700 MB, de duur van een audio-cd, dan maakt flac daar een flac bestand van dat tussen de 300 en 350 MB groot is. Geen wonder klein bestand, maar een factor 2 compressie bij een kwaliteitsverlies van nul is bijzonder knap in elkaar gezet.

Aanvankelijk bestond flac alleen voor unix-achtige systemen, maar er is nu ook een Windows variant gemaakt en sindsdien gaat het hard met de opmars ervan. In eerste instantie was er alleen een commandoregel gestuurde versie, die je vanuit een soort van dosbox (cmd.exe) onder Windows moest gebruiken. Maar zeker Windows gebruikers trekken een erg zuur gezicht als ze eens wat moeten typen in plaats van klikken, dus heeft iemand zich boos gemaakt en een brug voor de muisgebruiker geslagen. Er is nu een versie van flac die voorzien is van een front-end. Dat is gewoon een grafische interface, waarmee de gebruiker op de vertrouwde Windows manier bestanden kan aanwijzen die in flac moeten worden omgezet. Flac files maken is nu zo eenvoudig als 1 2 3 en net zo simpel als het maken van een mp3 bestandje.

Maar men heeft zich nog meer moeite getroost voor ons, Windows liefhebbers. Als je wav en mp3 bestanden kunt afspelen met winamp, dan moet flac ook gewoon afspeelbaar worden met winamp, toch? En dat is ook al voor elkaar gebokst. Voortaan ondersteunt winamp net zo gemakkelijk flac als dat het met mp3 kan omgaan. De full version van winamp kan al vanaf winamp 2.91 flac af spelen. Heb je echter voor winamp lite op je pc gekozen, dan kun je een eenvoudig te installeren plug-in downloaden zodat winamp lite eveneens flac afspeelt.

Maar het is nog spannender. Als je programma's gebruikt om geluid te bewerken, bijvoorbeeld Adobe Audition, dan zul je op internet import en export filters vinden waarmee je ook in Audition een flac-file kunt laden of bewaren, als waren het gewone wav-bestanden. En dat heeft weer tot voordeel, dat je vanaf nu in Audition zelfs nooit meer wav-files hoeft te gebruiken. Als je iets hebt opgenomen dat je normaal als wav bewaarde, kun je dat nu probleemloos opslaan als flac. Dat scheelt een hoop schijfruimte, zonder kwaliteit van je audio te hoeven inleveren. Kortom: als je er zin in hebt, is flac zeker je moeite lonend en dubbel en dwars waard. Zelf gebruik ik het om cd's die echt bijzonder goed zijn opgenomen en gemastered, via internet als flac naar vrienden te sturen. Die hoeven minder lang op het binnenrollen van de data te wachten en ze krijgen de haarscherpe, originele versie van de cd zonder enig kwaliteitsverlies. Het hangt er dus maar vanaf waar je de compressie voor nodig hebt. Moet het klein en is het niet erg om aan kwaliteit in te boeten, dan is mp3 en alles wat daarop lijkt prima te gebruiken. Accepteer je echter wat grotere bestanden omdat je het belangrijk genoeg vindt geen kwaliteit te verliezen, dan is flac het cadeautje van de meest recente audio techniek

Audio- en datacd's

Een ouderwetse audio-cd waar 74 minuten geluid op past, is 750 megabyte aan data. Dat klinkt vreemd, want op een recordablecd staat geschreven dat er maar 650 MB aan data op past en geen 750 MB. Toch is het waar. Als je een audio-cd maakt, dan mag er meer data op dan wanneer je er een data-cd van maakt. Die 650 MB geldt voor data, maar er mag 750 MB aan audio op. Dat komt omdat een cd-speler in het algemeen genomen erg foutgevoelig is. Als er geen foutcontrole zou zijn ingebouwd, dan zou je in een audio-cd voortdurend allerlei digitale tikjes en overslagjes horen. Een data-cd zou echter niet eens als zodanig gebruikt kunnen worden. Zonder foutcorrectie op de cd, zou er namelijk op de honderd bytes minstens 1 keertje iets fout gaan, waardoor het originele gebrande bestand gemiddeld elke 100 bytes wel een foutje zou bevatten. En dat kan natuurlijk nooit de bedoeling van een opslag medium zijn. Om dat toch goed te krijgen, gebruiken ze op de cd iets dat lijkt op crc: Cyclic Redundancy Check. Hoe het werkt voert nu te ver, maar het gaat erom, dat je op 100 procent aan data, ongeveer 5 procent extra data toe voegt. Die extra data is ervoor, om de originele gegevens terug te kunnen rekenen bij geval er een paar bytes achter elkaar onleesbaar zijn op de cd. Die foutcorrectie is er niet voor niets, want zeker met audio-cd's is de cd-speler voortdurend aan het gissen, terugrekenen en interpoleren, om te zorgen dat het originele signaal zo onbeschadigd mogelijk uit de D-A converter komt, ondanks de vele leesfouten uit de cd. Dankzij het feit dat ons oor niet tot op de sample nauwkeurig kan horen, is het dus mogelijk een "ongeveer" signaal terug te geven in plaats van exact. En dat gebeurt ook. De audio-cd-speler laat in theorie dus steeds een ander signaal horen als je hetzelfde liedje af speelt. Dat signaal is telkens anders, omdat de leesfouten op de cd gezien worden door de speler. De foutcorrectie daarvan is zo gemaakt, dat de speler wel ongeveer kan gissen wat het origineel geweest is, maar niet precies. Zou je dat wel precies willen hebben, dan zou er meer crc data aan de audio moeten worden toegevoegd. en dan zou er op een audio-cd geen 750, maar 650 megabyte aan gegevens passen; Net zo veel als op een data-cd. Philips heeft ervoor gekozen dit niet te doen, omdat het interpolatie- en foutcorrectiegedrag van een cd-speler voor iedereen onhoorbaar is.

Bij data zoals computerprogramma's echter, mag op een cd nooit een beetje afwijken. Gebeurt dat toch, dan krijgt de processor foute instructies en loopt je computer vast. Daarom is er meer extra crc data nodig op de data-cd, om te zorgen dat de gegevens die je erop zet, ook tot op het bit nauwkeurig terug gelezen zullen worden. Die ruimte gaat domweg van je opslagcapaciteit van de cd af. Een datacd bevat in feite dus even veel gegevens als een audio-cd, alleen ben je bij datacd's 100 megabyte kwijt aan foutcorrectie.

Tips voor het opnemen

Als je voor het eerst begint met het opnemen van geluid met je pc, dan loop je tegen een aantal dingen aan die op het eerste gezicht niet meteen helder zullen zijn. Daarom de volgende paragrafen om je op weg te helpen.

Aansluiten

Om te beginnen is er geluid dat je graag met je computer wilt opnemen. Dat kan geluid uit je microfoon zijn, maar het kan ook uit je cassettespeler, mini disk, DAT recorder, platenspeler of je radio komen. Alle audio apparaten die een uitgang hebben, kun je in principe op je computer aansluiten. Het apparaat waarvan je opneemt, noemen we de geluidsbron.

Een microfoontje kun je vaak rechtstreeks op de geluidskaart van je computer aansluiten. Er zijn genoeg goedkope microfoontjes in de handel die daarvoor gemaakt zijn. Ze hebben dan een snoer met aan het eind een jack plugje van 3.5 millimeter dik. Zo'n stekertje dat ook vaak aan het hoofdtelefoon snoertje zit van je walkman. Dit type plugje heet een 3.5 millimeter mini-jack. Heeft je microfoon een normale, dikke jack plug, dan heet het gewoon een jack, zonder mini. Als er in je geluidskaart geen ingang is waar je zo'n dikke jack plug in kunt drukken terwijl je microfoon er wel zo eentje heeft, dan kun je een verloopstukje kopen en dat brengt ons op de volgende kreet: mannetje of vrouwtje.

Een mannetjes plug is er een met een of meerdere contact pinnen. De vrouwtjes plug is de tegenhanger: het contactpunt waar je de mannetjes plug in moet drukken. De plug met de gaatjes. Meestal spreken we echter van male en female, liever dan van mannetjes en vrouwtjes pluggen. Meestal heeft je microfoon een 3.5 millimeter male mini-jack, en in je geluidskaart zit dan de 3.5 millimeter female mini-jack waar het microfoonplugje in kan. Het genoemde verloop plugje voor een microfoon met een dikke jack heeft aan de ene kant een female regular jack, en aan de andere kant een male 3.5 millimeter mini-jack. Daarmee kun je je microfoon met dikke plug toch gewoon op een geluidskaart met kleine jacks aansluiten.

Een enkele keer krijg ik opnames van mensen te horen, die hun microfoontje hebben gebruikt om het luidspreker geluid van hun cassette recorder op te nemen. Dat kun je natuurlijk wel doen, maar in feite klinkt het resultaat dan nergens naar. Je hoort de akoestiek van de kamer waarin het bandje is afgedraaid, je hoort het loopwerk van de cassette speler en je hoort alle geluiden die de aanwezigen in de kamer maakten terwijl de opname liep. Als je toch graag je cassette bandje of bandrecorder spoel om wilt zetten naar je pc, zorg er dan zo mogelijk voor dat je een afspeelapparaat bij iemand leent dat voorzien is van een audio uitgang, zodat het signaal via een kabeltje kan reizen in plaats van door een speaker, door de lucht en weer door een microfoon.

Soms krijg ik ook wel eens deze vraag: de apparaten die ik bekeken heb, beschikken over pluggen waar soms in en soms out bij staat. Moet je nu altijd de in pluggen met de in pluggen verbinden en moeten de out pluggen ook verbonden worden, of moet de in aan de out of andersom? Die vraag kun je door logisch denken beantwoorden.

Als een apparaat geluid opwekt, bijvoorbeeld doordat je je cassetterecorder op afspelen hebt gezet, dan wil dat apparaat zijn geluid aan de buitenwereld kwijt. Het is dus logisch dat je het geluid naar buiten haalt door de uitgang (out) van het apparaat. Dat is niet de luidspreker, al hoewel dat strikt genomen ook een uitgang is. Het gaat om een speciaal voor dit doel op het apparaat aangebrachte plug, of set van plugjes, die het mogelijk maakt dat apparaat via een kabel op iets anders aan te sluiten.

De audio reist dan door het kabeltje heen naar het andere apparaat en daar moet het weer naar binnen, omdat het opgenomen moet worden of hoorbaar gemaakt. Als je dus een signaal in een apparaat gaat stoppen, doe je dat via de ingang. Met andere woorden: de uitgang van je geluidsbron sluit je aan op de ingang van je opname apparaat, bijvoorbeeld je pc. Uit gaat naar in.

Nu bestaan er, elektronisch gezien, drie signaalsoorten waar wij mee te maken kunnen krijgen,

alle drie met een verschillende uitgangsspanning. Dat is voor ons van belang wanneer we onze pc op een geluidsbron gaan aansluiten. Het zijn: microfoonsignaal, platenspelersignaal en lijnsignaal. Het zal duidelijk zijn bij welke apparaten het microfoon signaal en het platenspeler signaal vandaan komen, en het lijn signaal wordt over het algemeen voor al het andere gebruikt. Daarover zo meteen meer.

Het is niet de bedoeling een microfoon signaal aan te sluiten op een ingang die gemaakt is om een platenspelersignaal te ontvangen. Evenmin kun je een platenspeler aansluiten op een ingang die een lijnsignaal verwacht. Je zult dus op voorhand moeten weten welk soort signaal je geluidsbron afgeeft en welke ingangen er beschikbaar zijn op je geluidskaart.

Dat lijkt ingewikkeld, maar in de praktijk is het heel simpel. Je geluidskaart heeft namelijk vaak maar twee soorten ingangen: microfoon in en line in. In principe kun je zo goed als alle apparaten op line-in van de geluidskaart aansluiten, tenzij het een microfoon of platenspeler is. Voor een microfoon gebruik je de daarvoor speciaal gemaakte ingang, en je platenspeler daar hebben we het straks wel over. Voor de rest geven alle andere apparaten een lijnsignaal, of het nou een tuner, bandrecorder of wat dan ook betreft, dus die gaan gewoon op line-in.

Natuurlijk moet je zelf zorgen voor de juiste aansluitsnoertjes. Om in de winkel naar het juiste snoertje met de goeie plugjes aan weers kanten te kunnen vragen, moet je nog weten hoe de naam is van de meest voorkomende aansluitpluggen. Zo'n twintig jaar geleden had je kabels met aan het einde een vrij dikke, ronde plug. Het is een metalen cirkeltje van een goeie centimeter doorsnee, met langs de binnenrand in de rondte vijf metalen pinnetjes op een rij. Zo'n ding heet een din plug. Din staat voor Deutsche Industrie Norm. Din is er in male en female. Heeft je apparaat een female din uitgang, dan heb je aan je kabel aan de ene kant een male din plug nodig. Maar tenzij je een oude bandrecorder o.i.d. wilt aansluiten, zul je waarschijnlijk niet vaak meer in aanraking komen met din pluggen.

Wel zijn er vandaag de dag veel stereo apparaten, die per ingang of uitgang twee losse stekertjes gebruiken. Eentje voor links, eentje voor rechts. Het apparaat bevat de female connector, dus je kabel moet male zijn om aan te kunnen sluiten. Ik heb het nu over een type plug met vele namen. De officiële naam is Kinch, maar hij wordt ook wel Japanse steker of tulp plug genoemd. Het vrouwtje ziet eruit als een appelmoes blik in het piep klein. Misschien amper een centimeter hoog en minder dan een centimeter doorsnee. De male tegenhanger, het ding dat erin past, is een kort metalen pinnetje van misschien drie millimeter dik en een halve centimeter lang. Zoals gezegd heb je met kinch altijd twee plugjes bij elkaar: links en rechts. Nu ken je de jack, de mini-jack, de din plug en de kinch, male en female.

Aansluiten van een platenspeler

Tenslotte nog iets over het aansluiten van een platenspeler. De moeilijkheid daarbij is, dat een platenspeler niet zomaar op je geluidskaart kan worden aangesloten. Immers: daarop zitten alleen een microfoon ingang en een lijn ingang. Een ingang voor een platenspelersignaal is er bijna nooit bij. Daarmee heb je dan een probleem, want rechtstreeks kun je de pick-up dan niet aansluiten. Toch is er een oplossing. Bij de wat modernere stereo installaties, en dan bedoel ik alles dat moderner is dan een elektrische koffergrammfoon of pick-up met ingebouwde speaker, gebruik je een losse platenspeler, die zijn geluid via een kabeltje uitstuurt naar de stereo versterker. Dat ding staat meestal in een kast in de woonkamer. Daar weer mee verbonden, zijn twee of meer luidsprekers die het uiteindelijke geluid uit de versterker in de kamer hoorbaar maken. Een afgespeelde plaat draait dus op de platenspeler, en het signaal loopt via de versterker naar de luidsprekers. Nu is het vaak mogelijk, om bijvoorbeeld op een cassettebandje een opname te maken van de plaat die wordt afgedraaid. Daarvoor gebruikt men veelal een losse cassette recorder. Een apparaat zonder eigen luidsprekers, alleen maar voorzien van een ingang en een uitgang. Dat apparaat, het cassette deck, wordt net als de pick-up op de versterker aangesloten. Als je een bandje

op het deck afspeelt, gaat het signaal vanuit de uitgang van het deck naar een ingang op je versterker, om van daaruit via andere kabels naar je boksen gestuurd te worden. Maar als je de cassette niet gaat afspelen maar er juist iets in wilt opnemen, dan moet er dus geluid ergens vandaan naar het deck gestuurd worden. Nu is de versterker altijd het centrale apparaat waarop alle geluidsbronnen binnen komen. Of het nu gaat om de pick-up, de radio of het cassette deck, als ze iets willen afspelen gaat hun signaal naar een ingang op de versterker. Apparaten als een cassette deck geven een lijnsignaal aan de versterker af, en als je er iets mee wilt opnemen dan verwachten ze ook een signaal van lijnsterkte op hun ingang. Nu hebben veel versterkers meestal een uitgang, waar het geluid op lijn-signaal sterkte wordt heen gestuurd dat op dat ogenblik wordt weergegeven. Als je dus een grammofoonplaat op een bandje wilt opnemen, dan legt het signaal de volgende weg af. Je pick-up produceert een signaal van het type platenspelersignaal. Dat reist door een snoertje naar je versterker, waar het een ingang binnen gaat die gevoelig genoeg is om het zachte signaaltje dat platenspelers nu eenmaal afgeven, te kunnen oppikken. De versterker heet niet voor niets zo, want hij geeft het signaal via een van zijn eigen uitgangen weer door aan de ingang van het cassette deck, alwaar het op het bandje zal worden opgenomen. Maar het cassette deck verwacht een signaal op volledige lijnsterkte. De versterker moet dus het zwakke platenspelersignaal oppompen, versterken, totdat het op lijnsterkte is zodat de cassette recorder er iets mee kan. En daarin zit 'm nou de kneep. De pick-up geeft een te zwak signaal af om met je geluidskaart op te kunnen nemen. Het moet versterkt worden. Wat kun je dus doen? Zowel je platenspeler als je versterker zet je bij je pc. Je pick-up sluit je zoals dat hoort aan op je versterker en de tape out uitgang van de versterker die vroeger naar de ingang van je cassette deck liep, sluit je nu aan op je geluidskaart. Niets bijzonders, alleen een apparaat extra nodig, je versterker. Dit is niet meer nodig met een USB-platenspeler.

De opname

Nu je de geluidsbron, wat het ook zijn mag, op je computer hebt aangesloten, moeten we nog daadwerkelijk op de pc gaan opnemen. Nu zit er heel veel programmatuur standaard bij Windows, maar er is niets bij waarmee we gewoon een uur achter elkaar geluid mee kunnen opnemen. We zullen het internet op moeten om zo'n programma te halen. Zonder er tien te gaan noemen: kijk eens op www.freecorder.com voor een gratis soundrecorder met alleen de basisfuncties. Wil je kunnen knippen en plakken, wat voor goede eindresultaten erg is aan te raden, werp dan eens een blik op Audition. Er zijn honderden van zulke programma's en ik noem er maar wat. Goldwave of sound forge behoren ook tot de keuzes. Haal maar eens wat demonstratieversies op en maak je keuze. Zoek eens uit hoe je vanuit zo'n programma een bestand kunt afspelen. Hoe je iets kunt opnemen en hoe je dat op schijf bewaart. Experimenteren kan geen kwaad en het verkennen van deze gereedschappen is een erg leuke bezigheid. Nu je een paar proefopnames hebt kunnen maken, wordt het tijd voor wat tips om uiteindelijk tot een mooi opgenomen wav te komen.

1. Mono of stereo? Onderzoek eerst of het geluid uit je geluidsbron, mono of stereo klinkt. Als je de wav gaat opnemen, dan kun je een monobron het best ook als mono wav-bestand opnemen. Daarbij gaat het er niet om of de cassettespeler of welk afspeelapparaat je ook gebruikt stereo is, maar het gaat erom of je tijdens het afspelen van de bron door beide speakers hetzelfde hoort of niet. Hoor je geen verschil tussen links en rechts, dan kun je de wav veilig in mono opnemen. Hoor je wel verschil, of zelfs mooie stereo effecten, dan is het jammer om die niet eveneens op te nemen. Is je bron dus stereo, maak de wav dan ook stereo, al kost het je tijdelijk wat meer schijf ruimte.
2. Begin en einde goed. Zet eerst je pc op opname en start dan pas je geluidsbron. Zo mis je in je pc-opname niets van het origineel. Als de opname is afgelopen, stop dan eerst je ge-

- luidsbron en dan pas je pc, om dezelfde reden. Sla de wav nu op naar schijf. Als je er strakjes iets aan verknoeit met bewerken, heb je het ongeschonden origineel nog.
3. Knip begin en eind. Aan het begin van je opname zit nu een stuk stilte, omdat je bezig was je geluidsbron te starten. Zet je audioprogramma op afspelen en kijk op je scherm goed naar de tijd teller. Als je hoort dat het geluid begint, stop je het afspelen onmiddellijk en kijkt hoe ver de tijd is opgelopen. Dit wordt het beginpunt van je uiteindelijke wav-file, want alles wat zich hiervoor bevindt gaan we weg knippen. Stilte is immers niet interessant om te beluisteren. Houd wel in gedachten dat je zelf even tijd nodig had om direct na het horen van het begin van de werkelijke audio, het afspelen te stoppen. Een kort voorbeeld. Stel je start de zojuist opgenomen wav-file vanaf het begin. Je hoort nu enkel stilte. Zo gauw je de eerste tonen van de opgenomen muziek hoort, stop je het afspelen. Stel dat er volgens de tijdteller in ons audioprogramma dan 12 seconden verstreken zijn. Als je nu de eerste 12 seconden van de wav-file weg knipt, ben je de stilte aan het begin kwijt zoals de bedoeling is. Maar je mist ook de eerste paar tiende seconden van het begin van de muziek. Om dat te voorkomen, trek je ongeveer een seconde van de begintijd af. Het punt waar de muziek zo'n beetje begint is dus $12 - 1 = 11$ seconden vanaf het begin van je wav-file. Knip de eerste elf seconden van je opname af. Als je de opname nu afspeelt, zul je horen dat de muziek vrijwel direct begint, zonder een lange stilte vooraf en dat is wel zo netjes. Aan het einde van je opname doe je hetzelfde: luister waar de audio ophoudt en knip alles weg vanaf dat punt tot het einde van je bestand. Controleer het begin en einde van je opname en als dat in orde is, bewaar dan de opname opnieuw.
 4. Normaliseren. Nu de opname strak is bijgeknipt, moet je nog zorgen dat het niveau van de opname sterkte precies goed komt te staan. Ik weet het, dat klinkt raar. De opname sterkte regelen achteraf. Toch is dat bij de pc de veiligste werkwijze. Maar waarom bestaat dat, die opname sterkte? Omdat niet elke bron een even hard signaal af geeft, zelfs al hebben ze allemaal een lijnsignaal. De ene plaat is harder opgenomen dan de andere, en op de radio klinkt de ene zender luider dan de andere. Het is de bedoeling dat je je geluidsopnames zo secuur mogelijk afregelt waar het de opname sterkte betreft. Is de opname sterkte te zacht, dan verzuipt de opname straks in de ruis die audio apparaten nu eenmaal allemaal genereren en dat klinkt lelijk en het is storend bij het uitluisteren van het opgenomen eindresultaat. Regel je echter te hard in, dan kan de geluidskaart het felle signaal niet aan en treedt er vervorming op, wat evenmin erg prettig luistert. Je kunt je geluid daarom het beste volgens de vu-meter van je computer programma, met opzet een tikje te zacht opnemen. Mocht er dan een harde piek in het geluid voor komen, dan zal het geluid daar in elk geval niet gaan vervormen. Je moet dus de balans vinden tussen te hard en te zacht. Dat is een kwestie van proberen en naar de vu-meter van je audioprogramma kijken. Gaat de boven de 0 lijn of in het rood, dan gaat hetingangssignaal te hard je geluidskaart in. Is het signaal te zacht, dan beweegt de vu-meter niet of nauwelijks. Nu je de opname met opzet net iets te zacht hebt opgenomen om eventuele harde pieken te kunnen opvangen, kan het zijn dat die pieken helemaal niet zijn gekomen, of dat de opname ondanks de pieken toch een tikje harder had gemogen. Je kunt nu achteraf de sterkte van je opgenomen wav bestand nog veranderen met je audioprogramma, zodat de computer bekijkt wat het exact juiste geluidsniveau is. Dat doe je met een programmafunctie die in de meeste software "normalize" is genoemd. Lees het helpschermje bij de functie en volg de aanwijzingen die daarin staan. Controleer het resultaat en bewaar je wav opnieuw.
 5. Eventueel mp3en. Hier zijn talloze programma's voor. Cdex kan het, de xing mp3 encoder kan het ook, maar waarschijnlijk kan je audioprogramma je wav ook al als mp3 opslaan. Doe dat nu, verlaat je audioprogramma door het af te sluiten en controleer het resultaat van de mp3 file.