

## Hoofdstuk 7

---

# IRQ's, DMA's, bussen, stuurprogramma's enzovoorts

---

---

*In dit hoofdstuk*

- > Onder de motorkap
  - > Wat zijn conflicten bij interrupts en IRQ's?
  - > Wat is DMA?
  - > Welke bus is de beste: ISA, MCA of EISA?
  - > Stuurprogramma's ontcijferen
- 
- 

### Onder de motorkap

In de goede oude tijd wisten gebruikers niet hoe hun computer werkte. De software werd geïnstalleerd en gestart. Een technisch hoogstandje was het instellen van de dip switches op een nieuwe

interne kaart voor een unieke LPT- of COM-poort. Er is wel het een en ander veranderd.

Omdat de gebruiker steeds meer kracht, snelheid en capaciteit wil, wordt de computer tot het uiterste beproefd. Oude standaarden zijn verdwenen en nu is alles wat de klok slaat "heavy metal": het rechtstreeks adresseren van de hardware. De nieuwste software en hardware zijn volledig gericht op snelheid; de ROM-BIOS wordt overgeslagen en de computercomponenten worden direct benaderd. Het voordeel hiervan is dat de snelheid toeneemt en er meer mogelijkheden zijn, maar de andere kant van de medaille is dat het er allemaal niet eenvoudiger op wordt en er meer conflicten in de hardware optreden.

De meeste computers zijn meer dan een veredelde typemachine of rekenmachine. Zij kunnen worden uitgerust met faxkaarten, scanners, digitaliseertabletten, tape streamers, CD-ROM's, luidsprekers, geluidskaarten en netwerkkaarten. Met al deze uitbreidingsmogelijkheden moet er wel een kijkje onder de motorkap worden genomen. U zult moeten weten hoe een computer IRQ's gebruikt, en hoe DMA's en andere mechanismen ervoor zorgen,

dat er geen onderlinge conflicten tussen de produkten optreden.

## **Conflicten bij interrupts en IRQ's?**

Het hart van de computer is de *microprocessor*, of de *centrale verwerkingseenheid (CVE)*. Deze microprocessor verschilt van de andere chips op de systeemkaart, omdat er berekeningen mee kunnen worden uitgevoerd.

De vroegere microprocessors van Intel waren niet geschikt voor het zeer snel berekenen van getallen met een drijvende komma. Veel gebruikers van rekenintensieve software, zoals spreadsheets en CAD-programma's, installeerden een tweede processor, een mathematische coprocessor. Software die werd geschreven voor deze processor, gaf berekeningen met drijvende komma door aan deze coprocessor. Met de introductie van de 486 is daar een einde aan gekomen, omdat deze microprocessor een geïntegreerde mathematische coprocessor bevat.

Een ander feit waardoor deze chip zich onderscheidt van de andere chips, is dat de microprocessor het

kloppende hart is van het systeembord. De gegevensstroom van en naar andere apparaten wordt door de microprocessor geregeld.

Veel van de bijkomende berekeningen worden door ondersteunende besturingschips uitgevoerd, waardoor de microprocessor zich kan richten op belangrijker zaken, zoals beheer en belangrijke rekentaken. Sommige besturingen zijn voor de disktestations en het toetsenbord, andere I/O-besturingen regelen de invoer en uitvoer van gegevens naar externe apparaten. De organisatie lijkt op die van een operatiekamer, waarin de microprocessor de chirurg is, waarbij veel belangrijke (maar ondergeschikte) taken worden uitgevoerd door het team, zoals de anesthesist, de technicus en het verplegend personeel.

## **Onderbreking van de microprocessor**

De microprocessor moet worden geïnformeerd dat er iets gaat gebeuren als een apparaat aandacht vraagt. Wanneer u bijvoorbeeld Ctrl-Break indrukt, wordt dit onmiddellijk door de besturing van het toetsenbord doorgegeven aan de microprocessor. Hetzelfde gebeurt als er een fax binnenkomt en u

bent gegevens in een spreadsheet aan het berekenen. De microprocessor moet ervoor zorgen dat de binnenkomende gegevens naar schijf worden opgeslagen. Deze signalen die naar de microprocessor worden gestuurd, worden *interrupts* genoemd. Per seconde kunnen zich honderden interrupts voordoen.

De interrupts worden in twee categorieën onverdeeld: *hardware* en *software*. Een hardware-interrupt wordt gegenereerd door de microprocessor, een andere chip of een apparaat dat aandacht wil van de microprocessor. Zo worden hardware-interrupts ook gegenereerd door disktestations, het toetsenbord, enzovoorts. Een software-interrupt wordt, zoals de naam al aangeeft, gegenereerd door software; door interrupts kan software communiceren met DOS en ROM-BIOS.

## Interruptvectoren

Wanneer een microprocessor wordt onderbroken, moet de huidige taak worden stilgelegd, alle gegevens worden opgeslagen die in de registers staan en moet tenslotte worden gereageerd op de onderbreking. Na het reageren op de onderbreking

worden de opgeslagen gegevens teruggeplaatst in de registers en wordt de oorspronkelijke taak hervat.

De gegevens die worden bewaard voordat een hardware-interrupt wordt beantwoord, worden opgeslagen in een stack (stapel). Als zich een hardware-interrupt voordoet, wordt er een stack uit een voorraad gehaald. Gezien het grote aantal interrupts per seconde kan deze voorraad uitgeput raken, of kan een stack een te geringe capaciteit hebben. Dit genereert een *stack overflow*. Wanneer u deze problemen tegenkomt, moet u met de opdracht STACKS in het bestand CONFIG.SYS het aantal en de grootte van de interruptstacks vergroten. Zie deel 4 *Alle DOS-opdrachten* voor meer bijzonderheden over deze opdracht STACKS.

Elke interrupt, of deze nu een hardware- of een software-interrupt is, krijgt een uniek nummer toegewezen. De interrupt roept niet steeds "Volluk" om aandacht te krijgen van de microprocessor, maar roept zijn nummer, bijvoorbeeld "Interrupt 10". Wanneer de microprocessor wordt onderbroken, wordt de code op een bepaald adres (de *interruptvector*) uitgevoerd. In hoofdstuk 11 leest u hoe u de vorm van de cursor kunt veranderen door interrupt 10h (16 decimaal) aan te

roepen. De code die als reactie op een interrupt wordt aangeroepen, is de *interrupt service routine*, of ISR.

Interruptvectoren kunnen worden omgeleid naar andere vectoren. Deze techniek staat bekend als *het installeren van een interrupthandler*. Wanneer er een kritieke fout bij DOS wordt gesignaleerd, wordt bijvoorbeeld een interrupt 24 gegenereerd en roept DOS een code aan waardoor het bericht Abort, Retry, or Fail verschijnt. Door interrupt 24 om te leiden naar een bepaalde eigen code, kunt u dit vervelende bericht omzeilen en de fout op een meer galante manier oplossen. In hoofdstuk 11 wordt op deze wijze getest of er een diskette in een diskettestation zit zonder dat het bekende foutbericht verschijnt.



Elke interrupt heeft een corresponderende interruptvector die de *segment:offset* van de code is die wordt uitgevoerd als de interrupt zich voordoet. Het geheugengebied waarin de tabel met interruptadressen ligt opgeslagen, is de *interrupt vector table*. Wanneer de computer wordt gestart, wordt de lokatie van de interruptvectortabel aan het begin van RAM op adres 0000:0000 ingesteld. In de vectortabel staat voor elke interrupt een record,

te beginnen met interrupt 0 (CVE gedeeld door 0), gevolg door 1 (stapsgewijze uitvoering), enzovoorts. In elk record in de tabel staan twee woorden (een woord is hier een tekenloze integer die twee bytes geheugen in beslag neemt). Het eerste woord is de offset van de vector en het tweede woord is het segment. Wanneer zich een interrupt voordoet, wordt het corresponderende record in de interruptvectortabel benaderd, de offsetwaarde wordt geladen in het IP-register en de segmentwaarde in het CS-register.



Niet alle waarden in de interruptvectortabel wijzen naar een uitvoerbare code. Sommige vectoren wijzen naar gegevensstructuren. Interrupt 41h bijvoorbeeld wijst naar de parametertabel van de vaste schijf.

## Interrupt request lines (IRQ's)

Er zou chaos in de elektronica ontstaan als de microprocessor onmiddellijk op elke interrupt zou moeten reageren. Om dit te voorkomen is er een speciale chip die verantwoordelijk is voor de afhandeling van de hardware-interrupts. Deze chip, de *programmable interrupt controle (PIC)*, regelt de

stroom van binnenkomende interrupts en geeft ze één voor één door aan de microprocessor.

Elk apparaat dat de microprocessor onderbreekt, krijgt een soort volgnummer toegewezen, een *interrupt request line (IRQ)*. Deze lijn is een directe lijn van het apparaat naar de PIC. In de oorspronkelijke PC en XT kon de PIC maximaal 8 verschillende IRQ's afhandelen. Bij de 80286

Tabel 7-1 IRQ-toewijzingen voor 8086/8088-systemen	
IRQ	Omschrijving apparaat
0	IRQ-tijdklok
1	Toetsenbord
2	IRQ-cascade
3	COM2
4	COM1
5	Besturing vaste schijf
6	Besturing diskettestation
7	LPT1

werden twee PIC-chips gebruikt waardoor de capaciteit werd vergroot naar 15 IRQ's.

Het aantal van 8 of 15 IRQ's is niet echt veel. Behalve de standaardapparatuur, zoals de besturing van de vaste schijf, de mathematische coprocessor, het videosysteem, de I/O-poorten en de real-time klok, willen ook andere apparaten een IRQ, bijvoorbeeld een faxkaart, de muis, een geluidskaart en een scanner. De standaarden voor het toewijzen van IRQ's zijn de laatste jaren uitgebreid. In de tabellen 7-1 en 7-2 staan respectievelijk deze standaarden voor 8086- en 80286-systemen en hoger.

Tabel 7-2 IRQ-toewijzingen voor 80286-systemen en hoger	
IRQ	Omschrijving apparaat
0	IRQ-tijdklok
1	Toetsenbord
2	IRQ-cascade
3	COM2 en COM4, of busmuis

4	COM1 en COM3
5	LPT2
6	Besturing diskteststation
7	LPT1
8	Real-time klok
9	Overgebrachte IRQ van IRQ2
10	Beschikbaar
11	Beschikbaar
12	Muis (op IBM PS/2-systemen)
13	Mathematische coprocessor
14	Besturing vaste schijf
15	Beschikbaar

### Toewijzingen van IRQ's bekijken

In DOS 6 zit de onmisbare utility MSD (Microsoft System Diagnostics), waarmee u het huidige gebruik van de IRQ's kunt vaststellen. In afbeelding

7-1 staat het resultaat van een MSD-sessie waarin op de knop IRQ is gedrukt, of Q is geselecteerd.

IRQ	Address	Description	IRQ Status Detected	Handled By
0	004B:1875	Timer Clock	Yes	Block Device
1	004B:1923	Keyboard	Yes	Block Device
2	F008:EF5A	Second 8259A	Yes	BIOS
3	F008:EF5A	COM2: COM4:	COM2:	BIOS
4	0638:0096	COM1: COM3:	COM1: Serial Mouse	MSD.EXE
5	F008:EF5A	LPT2:	No	BIOS
6	F008:EF57	Floppy Disk	Yes	BIOS
7	0070:06F4	LPT1:	Yes	System Area
8	F008:E89B	Real-Time Clock	Yes	BIOS
9	F008:F850	Redirected IRQ2	Yes	BIOS
10	F008:EF5A	(Reserved)		BIOS
11	CB19:058E	(Reserved)		SCSI MGR\$
12	F008:EF5A	(Reserved)		BIOS
13	F008:FEAB	Math Coprocessor	Yes	BIOS
14	F008:FF33	Fixed Disk	Yes	BIOS
15	F008:EF5A	(Reserved)		BIOS

IRQ Status: Displays current usage of hardware interrupts.

**Figuur 7-1: MSD laat de configuratie van IRQ's zien**

In de tweede kolom staan de bijzonderheden over de interruptvector (de code die wordt uitgevoerd als er zich een interrupt voordoet). De derde kolom is een algemene beschrijving van de IRQ-poort en in de derde en vierde kolom staat of de IRQ-lijn in gebruik is en als dat zo is, welk programma of apparaat dat is.

Vrije lijnen hebben een waarde No of een lege waarde in de kolom Detected. Niet alle IRQ's met een lege kolom Detected zijn echter beschikbaar.

Om er zeker van te zijn dat een IRQ-lijn beschikbaar is, moet u de derde en vierde kolom bekijken. Als in de kolom Handled By geen BIOS of Default Handlers staat, is de lijn waarschijnlijk niet vrij. In het voorbeeld is het programma WIN386 geïdentificeerd als de handler voor IRQ1.



Zoals tabel 7-2 laat zien, delen bepaalde apparaten dezelfde IRQ-lijn. Zowel COM1 als COM3 zijn toegewezen aan IRQ4. De IRQ-lijnen mogen zich in een grote populariteit verheugen. De computertechnici gingen ervan uit dat niemand gelijktijdig COM1 en COM3 zou gebruiken, dus vonden zij het te verdedigen dezelfde lijn aan deze twee poorten toe te wijzen. Wees nu eerlijk, wie zou er met twee modems op één computer gaan werken. Toch ligt hier de kiem voor een conflict, als een poort door een muis wordt gebruikt en de zusterpoort door een modem. Een muis en een modem worden wel tegelijkertijd gebruikt. Als de beschreven situatie de uwe is, zorg er dan voor dat de muis en de modem worden aangesloten op COM-poorten op verschillende IRQ-lijnen.

Voor elk hardware-apparaat is een IRQ-lijn nodig. Wanneer u bijvoorbeeld een faxkaart installeert, moet u de kaart voor een bepaalde IRQ configureren.

Voordat u nieuwe apparatuur aanschaft, moet u wel weten of uw systeem deze apparatuur kan 'ontvangen'. Er moet worden gecontroleerd of er een IRQ-lijn beschikbaar is en of de nieuwe apparatuur van deze lijn gebruik kan maken. 8-bits kaarten, en zelfs 16-bits kaarten, kunnen alleen de eerste acht IRQ's benaderen. Vraag de leverancier voor welke IRQ's hun apparatuur kan worden geconfigureerd.

Het zou wel eens nodig kunnen zijn dat u een ongebruikte COM- op LPT-poort moet uitschakelen voor een nieuw apparaat, of zelfs bestaande apparatuur moet aanpassen voor andere IRQ's. Op deze manier is een minder flexibel apparaat toch op de juiste IRQ aan te sluiten.

## Wat is DMA?

De microprocessor, ongetwijfeld de drukst bezette chip in de hele computer, is de flessehals als het gaat om de verwerkingssnelheid. Hoe meer taken deze chip aan kan, des te beter is het.

Een veel voorkomende, maar tijdrovende handeling is het overbrengen van gegevens van en naar het geheugen. Niet alle handelingen die met het geheugen hebben te maken, dienen door de microprocessor te worden uitgevoerd. Wanneer bijvoorbeeld gegevens direct van de vaste schijf naar een diskette worden overgebracht, hoeven er geen berekeningen worden uitgevoerd. Het is dan ook niet nodig dat de microprocessor bij deze activiteit wordt betrokken. Een speciale chip, de *direct memory access (DMA)* besturing, kan gegevens direct van en naar het geheugen overbrengen. Alhoewel de microprocessor het eigenlijke werk niet uitvoert en daardoor minder wordt belast, heeft deze chip wel de 'eindverantwoording'.

De oorspronkelijke PC's en XT's maakten gebruik van een enkele besturingschip van Intel, 82327A-5 DMA, met vier DMA-kanalen. In theorie kan elk

kanaal tegelijkertijd worden gebruikt voor het verplaatsen van gegevens naar en vanuit het geheugen. Eén van deze kanalen is voor het opfrissen van het systeemgeheugen, zodat er drie kanalen overblijven voor de algemene activiteiten. Deze chip kan spijtig genoeg alleen 8-bits geheugenoverbrengingen afhandelen en dus maar 1MB geheugen adresseren.

Een tweede 8237A chip werd toegevoegd aan 286 en hogere systemen. Zoals de tweelinginterrupt besturingsconfiguratie die eerder in dit hoofdstuk is besproken, wordt de tweede DMA-besturing vastgemaakt aan één van de eerste besturingskanalen van DMA, zodat er in totaal zeven kanalen beschikbaar zijn. De tweede chip kan 16-bits geheugenoverbrengingen afhandelen en heeft zodoende volledige toegang tot het gehele geheugenpotentieel van het 286-systeem, hetgeen resulteert in hogere prestaties.

Over het algemeen wordt er maar één DMA-kanaal per keer gebruikt. De opvallendste uitzondering wordt gevonden in programma's voor het maken van backups, waarin vaak twee kanalen naast elkaar worden gebruikt. Het ene kanaal regelt de geheugenoverbrenging naar de vaste schijf en het



andere kanaal regelt dit naar de diskette. Het gelijktijdige gebruik van twee DMA-kanalen wordt *concurrent read-write* of *dual DMA* genoemd.



Bij sommige backup-programma's hebt u de keuze uit drie modi of snelheden:

- Snel. De DMA-besturingschip voert gelijktijdige lees- en schrijfbewerkingen uit.
- Gemiddeld. De DMA-chip wordt gebruikt voor geheugenoverbrenging, maar per keer wordt er slechts één kanaal gebruikt.
- Langzaam. De DMA-besturing wordt niet gebruikt. Het programma maakt gebruik van de microprocessor voor de standaard gegevensinvoer en -uitvoer.

De nieuwe backup-programma's van DOS 6, MSBACKUP en MWBACKUP, ondersteunen alleen de snelheidsopties Gemiddeld en Langzaam.

Bij het geheugenbeheersprogramma EMM386 zit de schakeloptie /D waarmee de grootte van de DMA-buffer kan worden bepaald. De grootte van deze buffer, standaard 16K, is in de meeste gevallen

voldoende. Wanneer u het bericht ziet dat de DMA-buffer te klein is, dient u de grootte met deze schakeloptie te wijzigen. Elke waarde (in kilobytes) tussen 16 en 256 is toegestaan.

## Welke bus is de beste: ISA, MCA of EISA?

Eén van de belangrijkste oorzaken van de populariteit van de IBM PC was dat de computer is ontworpen met een open architectuur. Het meest concrete aspect van die open architectuur is de toepassing van algemene uitbreidingsleuven, die door middel van de bus met de microprocessor en andere chips zijn verbonden. Andere producenten maakten uitbreidingskaarten, zoals interne modems, videokaarten en geheugenkaarten, die op de systeemkaart konden worden ingeplugd. Deze open architectuur leidde tot een hevige concurrentiestrijd waardoor IBM werd gedwongen een nieuw busontwerp te implementeren, de *Micro Channel Architecture*. IBM liet de open architectuur voor wat het was en probeerde grote royalty's op te strijken van potentiële naapers.

## De rol van de bus

De bus is de slagader van de computer die alle uitbreidingskaarten met de microprocessor verbindt. Deze bus bestaat uit een serie parallelle "draden", de *circuit pathways*. Net zoals op een autoweg met zes banen verplaatsen gegevens zich gelijktijdig over verschillende banen. Hoe meer banen, des te sneller gaat de stroom en des te beter de prestaties. Wanneer u een uitbreidingskaart aansluit op de systeemkaart, wordt het circuit van de uitbreidingskaart rechtstreeks aangesloten op de bus.

Het hoofddoel van de bus is om gegevens van en naar de microprocessor over te brengen, of tussen apparaten via de DMA-besturing. Omdat alle uitbreidingskaarten gebruik maken van dezelfde autoweg, moet de gegevensstroom worden gereguleerd. De bus bestaat uit vier grote onderdelen:

- Stroomlijnen
- Besturingslijnen
- Adreslijnen

## ■ Gegevenslijnen

De stroomlijnen leveren, zoals verwacht, de stroom voor de uitbreidingskaarten. Met de besturingslijnen worden tijdsignalen overgebracht van de systeemklok en naar de signaalinterrupts. Voordat er gegevens worden overgebracht, wordt het doeladres langs de gegevenslijnen verzonden waardoor de ontvanger wordt gewaarschuwd dat er gegevens worden gestuurd. De microprocessor controleert vervolgens of er een speciaal signaal is, dat de procedure laat starten (op de I/O Channel Ready lijn). Als alles in orde is, worden de gegevens langs de gegevenslijnen overgebracht.

Het aantal lijnen dat beschikbaar is voor de adresbus, is bepalend voor de totale hoeveelheid adresseerbaar geheugen. Twintig adreslijnen kunnen bijvoorbeeld 1MB geheugen adresseren. Het aantal buslijnen dat beschikbaar is voor de gegevensbus, is bepalend voor de doorvoercapaciteit van de bus. In de regel wordt er een optimale prestatie bereikt als het aantal gegevenslijnen overeenkomt met de gegevenslijnen in de microprocessor. Het aantal gegevenslijnen is de karakteristiek van de computerbus waarover het meest wordt gesproken. Een 16-bits bus

bijvoorbeeld betekent dat de bus gebruik maakt van 16 gegevenslijnen.

In de computerwereld zijn er drie soorten bussen: ISA, MCA en EISA. U kunt met MSD opvragen welke bus er op uw systeem wordt gebruikt door MSD te starten en op P te drukken voor de computerinstellingen (zie afbeelding 7-2).



Figuur 7-2: Computer in MSD laat het type systeembus zien

## De ISA-bus

Een naam geven aan busarchitecturen is zoiets als een naam verzinnen voor hoestpastilles of drankjes.

Oorspronkelijk waren er geen namen voor bussen die in PC's en XT's werden gebruikt. Er was maar één type bus, dus hoefde het beestje geen naam te hebben. Nadat er andere bustypen op de markt kwamen, werd het oorspronkelijke bustype *Industry-Standard Architecture* of ISA genoemd. De bus in de IBM PC AT (286) is de ISA-bus.

De bussen die in de eerste computers werden toegepast, hadden 62 lijnen: 20 daarvan waren adreslijnen en 8 lijnen waren gegevenslijnen. De bus kon slechts 1MB geheugen benaderen en gegevens werden overgebracht in enkel-byte pakketten (8 bits). De snelheid van de bus lag gelijk aan die van de microprocessor (4,77MHz).

Het originele ontwerp was duidelijk ouderwets. De 286 werd ontwikkeld en de bus werd uitgebreid en verbeterd. De meest in het oog springende wijziging in het ontwerp was de installatie van extra adressen en gegevenslijnen.

Om de bus compatibel te houden met de oorspronkelijke bus, zodat alle of bijna alle uitbreidingskaarten zouden werken op de nieuwe bus, werden de extra lijnen benaderd via een tweede connector op de systeemkaart. De originele bus

maakte gebruik van een connector van 3,2 inch met langs elke kant 31 contactpunten. De AT-bus ondersteunde een connector van 2 inch met langs elke kant nog eens 16 contactpunten.

De tweede connector leverde vier extra adreslijnen, wat het totaal op 24 adreslijnen bracht. Zodoende kon 16MB geheugen (gelijk aan de 286-processor) worden geadresseerd. Door de acht nieuwe gegevenslijnen was het mogelijk 16 gegevensbits per keer te verplaatsen.

De technici brachten ook een belangrijke verandering aan in het ontwerp van de busklok. Oorspronkelijk maakten zowel de microprocessor als de bus gebruik van dezelfde tijdchip en werkten daarom op dezelfde snelheid. De bus kon makkelijk op 6 of 8MHz draaien, maar 12 of 16MHz was te snel, de beoogde snelheid van de 286. Omdat de snelheid van de microprocessor steeds werd verhoogd, werd er een bus ontworpen die op een aparte klokchip kon draaien. Op deze manier waren de bus en de microprocessor niet meer afhankelijk van dezelfde chip en konden beide op eigen snelheid werken.

## De MCA-bus

In het midden van de tachtiger jaren leek het erop of de ISA-bus de flessehals van het systeem was. Geheugenchips en microprocessoren werden steeds sneller, maar de bus daarentegen niet. IBM besloot een nieuwe serie computers op de markt te brengen, de PS/2, met een volledig nieuwe busarchitectuur. Deze architectuur verbeterde de gegevensverwerking opzienbarend en was geschikt om de komende tien jaar gelijke tred te houden met de microprocessor. De nieuwe bus werd door IBM de *Micro Channel Architecture* (MCA) genoemd.

De eerste PC met deze nieuwe bus werd in 1987 op de markt gebracht. IBM ging ervan uit dat dankzij de verbeterde prestaties en de naam IBM klanten systemen zouden kopen die waren opgebouwd rond deze bus. Tot nu toe is dat niet zo. Twee factoren zijn debet aan de lauwe reacties van de gebruikers:

- Het nieuwe busontwerp was totaal anders dan de ISA-bus. Kaarten die waren ontworpen voor de ISA-bus, werkten niet op systemen met de MCA-bus en omgekeerd. De MCA-adapterkaarten waren kleiner en de connectoren waren anders.

- Vooruitlopend op het feit dat de concurrentie wel gedwongen was om de MCA-standaard in hun systemen te gaan toepassen, eiste IBM een aanzienlijke royalty van alle fabrikanten die hun systemen rond de MCA-bus produceerden.

Technisch gezien is de MCA veel beter dan de ISA. De grootste verbetering was een 32-bits gegevensbus en een 32-bits adresbus. Gegevens konden in pakketten van 4 bytes worden verplaatst en een gegevensblok van 4 gigabytes kon worden benaderd. Omdat IBM ook het ontwerp van de signaal- en stroomlijnen herzag om de kans op storingen te verkleinen, kon de bus sneller dan op 10MHz werken.

Daarbij konden onafhankelijke apparaten gezamenlijk de bus besturen. Het concept van het laten besturen van de bus door een ander apparaat dan de microprocessor wordt *bus mastering* genoemd. Zo kunnen twee of meer apparaten met elkaar communiceren zonder dat de microprocessor hierbij wordt betrokken. Bovendien konden MCA-systemen door middel van het bus-mastering ontwerp meerdere (parallele) microprocessoren ondersteunen (in een systeem kunnen vier 386-microprocessoren op een uitbreidingskaart zijn

geïnstalleerd). De toepassing van parallele microprocessoren bezorgde veel software-ontwikkelaars kopzorgen; er zijn geen grote programma's die gebruik maken van parallele processoren.

Een andere vernieuwing in de MCA-bus was de afschaffing van de dip switches voor de uitbreidingskaart. De techniek die werd toegepast, lijkt veel op de manier waarop de computer de configuratie van de vaste schijf en het geheugen onthoudt, namelijk door een CMOS-geheugen dat wordt gevoed door een batterij. Een MCA-systeem kan de configuratie van de bijbehorende uitbreidingskaart onthouden. Zo kunt u uitbreidingskaarten configureren met software en het toetsenbord in plaats van met een schroevendraaier.

Het staat buiten kijf dat de MCA-architectuur technisch vele malen beter is dan ISA.

## De EISA-bus

Terwijl IBM probeerde haar bus erdoor te drukken, besloten andere producenten, waaronder Compaq,

Hewlett Packard en AST Research, met een alternatieve bus op de markt te komen. Men hechtte veel waarde aan de negatieve reacties van de gebruiker over het gebrek aan compatibiliteit tussen de uitbreidingskaarten van ISA en MCA. De nieuwe bus zou volledig compatibel moeten zijn met de uitbreidingskaarten van ISA. Het resultaat was de *Extended Industry Standard Architecture (EISA)*.

Kort samengevat heeft de EISA-bus een aantal mogelijkheden die gelijk zijn aan de MCA-bus. De EISA heeft 32 adreslijnen en 32 gegevenslijnen. Dankzij verbeterde DMA-besturingen waren geheugenoverbrengingen door 32-bits DMA mogelijk. Net zoals bij de MCA wordt de bussnelheid bestuurd door een aparte busklok, waardoor de bus en de microprocessor qua snelheid onafhankelijk van elkaar kunnen werken. Door EISA wordt ook de zeer belangrijke bus-mastering ondersteund.

Het belangrijkste verschil tussen MCA en EISA is niet elektrisch maar fysiek. EISA heeft vanuit marketing oogpunt een beslissend voordeel boven MCA, omdat de uitbreidingskaarten van ISA wel

kunnen worden gebruikt in systemen met een EISA-bus maar niet in systemen met een MCA-bus.

De achterwaartse comptabiliteit wordt gerealiseerd door een slimme maar toch zo eenvoudig connector-ontwerp. De EISA-connector bestaat uit twee verschillende niveaus, met de onderste rij tanden als de nieuwe EISA-verbindingen. Wanneer er een ISA-kaart op een EISA-bus wordt aangesloten, maken de verbindingen van de ISA-kaart geen contact met de specifieke EISA-verbindingen, omdat de ISA-verbindingen hiervoor te kort zijn.

Alhoewel een EISA-kaart in een ISA-systeem past, moet u dit nooit en te nimmer doen. EISA-kaarten zijn niet compatibel met ISA-bussen. U kunt hiermee zowel de systeemkaart als de EISA-kaart beschadigen.

De kaarten van ISA en EISA zijn gelijk, waarbij de verbindingpunten van de EISA iets langer zijn. De MCA-kaart is totaal anders qua vorm, afmeting en ontwerp van de connector.

## De nieuwste ontwikkelingen

Alhoewel de producenten van MCA en EISA in een concurrentiestrijd zijn verwickeld, is de ISA-bus op enkele punten ingrijpend veranderd. ISA blijkt veerkrachtiger te zijn dan menige computertehnicus voor mogelijk had gehouden.

### De local bus

Wanneer u een enquête gehouden zou hebben onder gebruikers van computers met de vraag welk onderdeel sneller zou moeten, had het merendeel "scherm" geantwoord. Door de enorme vlucht van Windows en grafische programma's is het scherm de flessehals geworden van het systeem. Dat is niet te verwonderen met megabytes gegevens die de microprocessor naar het scherm stuurt. Zelfs de EISA- en de MCA-bus hebben moeite deze stroom gegevens te verwerken, omdat de snelheid van deze bussen nog steeds op ongeveer 10MHz ligt.

De oplossing is een *local bus*. Deze 32-bits bus is rechtstreeks aan de microprocessor gekoppeld en maakt gebruik van dezelfde klokchip. De local bus werkt dus op dezelfde snelheid als de microprocessor. Een enkele adapterkaart kon

bussnelheden van 33 of 50MHz aan, maar ook hiervoor werd een oplossing gevonden. Een groep producenten die zich voornamelijk richten op monitoren, ontwikkelden de VESA (Video Electronics Standards Association) VL-busstandaard.

Videokaarten die VESA compatibel zijn, kunnen ingeplugd worden op computers met een VL-bus. De resultaten zijn verbluffend en de kosten zijn een fractie van de videokaarten met video-coprocessorchips. Steeds meer computers met ISA worden geleverd met aansluitingen voor een local bus. Elke maand komen er nieuwe VESA-videokaarten op de markt.

Drie is het maximale aantal apparaten dat op een local bus kan worden aangesloten. De voornaamste kandidaten hiervoor zijn video-, netwerk en opslagapparatuur.

Het ontwerp van de local bus is een dermate succes dat ten tijde van het schrijven van dit boek, Intel met een eigen local bus op de markt kwam, de *Peripheral Component Interconnect (PCI)*.

### **SCSI, de niet-bus**

"Wanneer is een bus geen bus? Als het een SCSI is."  
Dit zou de vraag in een quiz kunnen zijn.

De *Small Computer System Interface (SCSI)*, uitgesproken als "skoezie" of "skuzzie", is een zeer snel systeem voor het aansluiten van apparatuur zoals vaste schijven, CD-ROM's en tape streamers. De meeste mensen denken dat de SCSI-besturing gewoon nog een besturing voor een vaste schijf is, maar SCSI is veel meer.

Verscheidene apparaten (tot zeven stuks) worden alle parallel gekoppeld door een enkele lintkabel die op zijn beurt weer is aangesloten op de SCSI-besturing. De SCSI-kabel is in feite een uitbreidingsbus voor het verbinden van snelle opslagsystemen en scanners. SCSI-apparaten kunnen communiceren en onderling gegevens uitwisselen, zonder dat de microprocessor of de standaard systeembus daarbij te pas komt. Stelt u voor dat u bestanden vanaf de vaste schijf naar een tape streamer kunt opslaan zonder dat het systeem trager wordt. Het SCSI-ontwerp kan met de standaard bustypen ISA, MCA en EISA werken.

Het oorspronkelijke SCSI-ontwerp maakte gebruik van een 50-bits (of lijn) bus en had 8 gegevensbits. De specificatie van SCSI-2 die in 1992 op de markt kwam, ondersteunde 16 en 32 gegevenslijnen waardoor een gegevensoverbrenging van meer dan 40MB per seconde mogelijk werd. Maar de herziene standaard is niet vrij van problemen. Niet alle SCSI-apparaten kunnen met elkaar samenwerken. Twee gedetailleerde programmerspecificaties strijden met elkaar om de gunst van de gebruiker: de *Common Access Method Committee (CAM)* en de *Advanced SCSI Programming Interface (ASPI)*.

Technici zijn bezig SCSI-besturingen te ontwerpen die kunnen worden gekoppeld aan een ISA local bus. Wie zei er ook al weer dat ISA dood is?

### **Wat nu?**

Met geheugen dat op een bus draait, randapparatuur die gebruik maakt van de standaardbus en video op een local bus zal geen enkele bus, ISA, MCA of EISA niet, de markt beheersen. In de toekomst zal een computer waarschijnlijk een groot aantal bussen hebben waarmee de desbetreffende apparatuur onderling



wordt verbonden en zo snel mogelijk zal werken. De tijd zal het leren.

## Stuurprogramma's ontcijferen

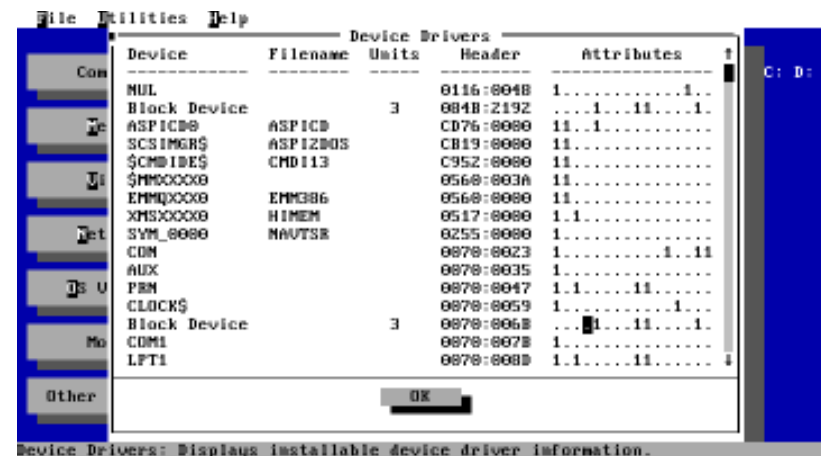
Het besturingssysteem is ontworpen om zich aan te passen aan standaard computerapparatuur, zoals videokaarten, printers, modems en diskettstations. Van het begin af aan was het duidelijk dat DOS al die apparatuur niet zou kunnen ondersteunen. In DOS 2 kon er softwarematig een speciale code worden geïnstalleerd voor het behandelen van niet-standaard apparatuur, zoals scanners en faxkaarten. Software waardoor DOS kan communiceren met een apparaat, staat bekend als een *stuurprogramma*, dat gewoonlijk in CONFIG.SYS wordt geladen.

Veel stuurprogramma's imiteren standaardapparaten. Een stuurprogramma voor een niet-ondersteunde vaste schijf antwoordt op de normale manier op de instructies van DOS, maar communiceert door middel van andere opdrachten met de vaste schijf. Zo laat het stuurprogramma van een joy stick het voorkomen alsof de joystick een

toetsenbord is. Het stuurprogramma gedraagt zich als een tolk, een tussenpersoon, waardoor een vreemd apparaat toch de taal van DOS begrijpt en omgekeerd.

Achter de schermen worden veel standaard DOS-apparaten ondersteund door stuurprogramma's die worden geladen vanuit het verborgen DOS-bestand IO.SYS.

Start MSD en druk op R voor een lijst met stuurprogramma's die op uw systeem zijn geïnstalleerd. Zie afbeelding 7-3 voor een dergelijke lijst.



**Figuur 7-3:** MSD geeft een overzicht van de stuurprogramma's

Routines van stuurprogramma's worden normaal gesproken aangeroepen door een software-interrupt. Wanneer een stuurprogramma is geïnstalleerd, wordt er een waarde in de interruptvectortabel geplaatst die naar de hoofdcode van het stuurprogramma wijst. In de vierde kolom van de lijst met stuurprogramma's (MSD) staan de beginadressen van de hoofdcode van het stuurprogramma. DOS onderhoudt een lijst (of ketting) met stuurprogramma's waarin elk stuurprogramma wijst naar het volgende in de lijst. Het eerste stuurprogramma in de lijst is NUL.

U kunt aanvullende informatie over het stuurprogramma, zoals de grootte van het stuurprogramma, oproepen met de opdracht MEM /D. Met het zoekfilter Find kunnen overbodige gegevens worden uitgefilterd:

```
mem /d | find "device" /i
```

Deze opdracht genereert een lijst die lijkt op de onderstaande:

		CON	System Device Driver
		AUX	System Device Driver
		PRN	System Device Driver
		CLOCK\$	System Device Driver
		A: - C:	System Device Driver
		COM1	System Device Driver
		LPT1	System Device Driver
		LPT2	System Device Driver
		LPT3	System Device Driver
		COM2	System Device Driver
		COM3	System Device Driver
		COM4	System Device Driver
1152	(1K)	XMSXXXX0	Installed Device=HIMEM
3104	(3K)	EMMQXXX0	Installed Device=EMM386
28912	(28K)	@CACHE-X	Installed Device=NCACHE2
768	(1K)	SETVERXX	Installed Device=SETVER
8096	(8K)	NAV&XXXX	Installed Device=NAV&
14912	(15K)	PC\$MOUSE	Installed Device=GMOUSE
44336	(43K)	DBLSSYS\$	Installed Device=DBLSPACE

Stuurprogramma's zijn onder te verdelen in twee basiscategorieën: tekenstuurprogramma's en blokstuurprogramma's. Tekenstuurprogramma's zijn één-byte georiënteerd, zoals printerpoorten. Deze stuurprogramma's omvatten onder andere de standaard DOS-apparatuur, zoals AUX, CON, LPT en NUL. Blokstuurprogramma's behandelen blokken gegevens en worden gewoonlijk toegepast bij stations, CD-ROM's en andere gegevensintensieve stuurprogramma's. Aan elk blokapparaat wordt een logische stationsletter toegekend. In de meest rechtse kolom Attributes in het stuurprogrammavenster van MSD staat meer informatie over het formaat van het

stuurprogramma. Zie tabel 7-3 voor de betekenissen van elk attribuut.

Stuurprogramma's hoeven niet naar een bepaald hardware-apparaat te verwijzen. ANSI.SYS is bijvoorbeeld een stuurprogramma dat de mogelijkheden van het toetsenbord en scherm vergroot. U kunt ook bepaalde stuurprogramma's als residente programma's laden in plaats van als standaardstuurprogramma's in CONFIG.SYS. In het meest bekende voorbeeld, het stuurprogramma van de Microsoft-muis, laadt u MOUSE.SYS in CONFIG.SYS of start u MOUSE.COM vanaf de opdrachtregel.

Om conventioneel geheugen uit te sparen kunt u met DEVICEHIGH en LOADHIGH respectievelijk stuurprogramma's en residente programma's in het hogere geheugen laden. Zie hoofdstuk 5 voor meer informatie over het HIGH laden van stuurprogramma's.

Tabel 7-3 Attributen stuurprogramma

Bitpositie	Attribuut	Betekenis
0	.....1	Standaard invoerapparaat
1	.....1.	Standaard uitvoer (teken) apparaat
2	.....1..	NUL
3	.....1...	Klokapparaat
4	.....1....	Interrupt 29 service stuurprogramma
5	.....1.....	Gereserveerd
6	.....1.....	Stuurprogramma ondersteunt IOCTL (I/O-besturing)
7	.....1.....	Gereserveerd
8	.....1.....	Gereserveerd
9	.....1.....	Gereserveerd
A	.....1.....	Gereserveerd

B	....1.....	Open/gesloten/verwisselbare gegevensdragers niet ondersteund
C	...1.....	Gereserveerd
D	..1.....	Of een tekenapparaat dat uitvoer-tot-bezig niet ondersteunt of een blokapparaat dat geen gebruik maakt van IBM-blokopmaak.
E	.1.....	IOCTL ondersteund
F	1.....	Tekenapparaat - blokapparaat bij leeg tekenapparaat

---



---

### *Samenvatting*

In dit hoofdstuk zijn veel factoren van de hardware onder de loep genomen die te maken hebben met de configuratie en toegevoegde apparatuur. De volgende belangrijke punten zijn besproken:

- > De taken van de microprocessor worden geregeld door interrupts; deze interrupts worden weer gekenmerkt als hardware en software. Wanneer een interrupt zich voordoet, wordt er door de microprocessor een interrupt-serviceroutine uitgevoerd waarvan het adres in de interruptvectortabel staat.
- > Alle hardware-interrupts worden beheerd door de programmeerbare interruptbesturing, die via IRQ-lijnen met de apparatuur is verbonden. Normaal gesproken moet elk apparaat een eigen IRQ-lijn hebben.
- > Het verplaatsen van gegevens uit het geheugen kan worden uitgevoerd door de besturingschip van de DMA (direct memory access). De gegevens kunnen langs meerdere kanalen worden verplaatst, maar gewoonlijk worden alleen door backup-programma's meerder kanalen tegelijk gebruikt.
- > De uitbreidingsbus verbindt apparaten met de microprocessor en aanverwante circuits. De voornaamste bustypen in de computer zijn ISA, MCA en EISA. ISA is

nieuw leven ingeblazen dankzij de technologie van de local bus.

- > Stuurprogramma's gedragen zich als tolken waardoor vreemde apparaten kunnen communiceren met DOS en de microprocessor. Met de opdrachten MSD en MEM roept u een overzicht op van de geïnstalleerde stuurprogramma's.

Met DOS 6 is er een aantal nieuwe opdrachten geïntroduceerd voor configuraties waarmee meerdere starts mogelijk zijn. In hoofdstuk 8 wordt beschreven hoe u met deze opdrachten startmenu's maakt en hoe de normale configuratie-opdrachten zijn te identificeren.

---

---

