

①9



Octrooiraad
Nederland

①1 **193474**

①2 **C OCTROOI**

②1 Aanvraag om octrooi: **8503516**

⑤1 Int.Cl.⁷
G06F9/38

②2 Ingediend: **19.12.85**

③0 Voorrang:
29.12.84 JP 0276835/84

④3 Ter inzage gelegd:
16.07.86 I.E. 86/14

④4 Openbaargemaakt:
01.07.99 I.E. 99/07

④7 Dagtekening:
02.11.99

④5 Uitgegeven:
03.01.2000 I.E. 2000/01

⑦3 Octrooihouder(s):
**Sony Corporation (Sony Kabushiki Kaisha) te
Tokio, Japan (JP).**

⑦4 Gemachtigde:
Ir. P.N. Hoorweg c.s. te 2517 GK Den Haag.

⑤4 **Werkwijze voor het bedrijven van een microprocessor.**

Werkwijze voor het bedrijven van een microprocessor

De onderhavige uitvinding heeft betrekking op een werkwijze voor het bedrijven van een microprocessor met pijpleidingsbouw, welke bedrijfswijze de mogelijkheid biedt om tijdens een aangevangen instructie-uitvoeringscyclus voor een latere instructie-uitvoeringscyclus bestemde instructie-informatie uit een geheugen te halen.

Uit de publicatie "theory and Design of Digital Computer Systems" van D. Lewin, 1981, par. 9.3, blz. 383 e.v. is een dergelijke werkwijze voor het bedrijven van de microprocessor bekend.

Moderne, monolitisch uitgevoerde microcomputers vertonen vaak een opbouw, welke aan een pijpleiding doet denken, en de mogelijkheid verschaft, dat ophaal- of haalcycli (fetch cycles) en instructie-uitvoeringscycli (instruction execution cycles) elkaar overlappen. Tijdens een haalcycli (fetch cycle) worden instructies uit een geheugen (op)gehaald. Iedere instructie bestaat uit een in een geheugen van het ROM-type, bijvoorbeeld een programmeerbaar geheugen van dit type of PROM-type, opgeslagen operatiecode (OP-code) en uit een eveneens in een geheugen van het ROM-type opgeslagen operand, welke kan wijzen naar een in een geheugen van het RAM-type opgeslagen waarde van een variabele. Bij een pijpleidingsbouw van gebruikelijk type worden het halen van de operatiecode en het halen van de operand uitgevoerd tijdens verschillende instructie-uitvoeringscycli, welke hierna als "instructiecycli" zullen worden aangeduid. Voor het halen van een 2-bytes instructie met een 1-byte operatiecode en een 1-byte operand zijn bijvoorbeeld in een 4-bits microcomputer twee instructiecycli nodig. In dat geval neemt derhalve iedere instructiecyclus drie machineklokcyclus in beslag. De efficiency van een dergelijke pijpleidingsopbouw van gebruikelijk type is derhalve niet zeer bevredigend.

In het algemeen is de haalcycli voor het halen van een operatiecode of van een operand korter dan de instructiecyclus, hetgeen wil zeggen, dat in de instructiecyclus een periode voorkomt, waarin geen halen plaatsvindt. Indien de volgende instructiecomponent tijdens een dergelijke periode zou kunnen worden gehaald, kan de looptijd worden verkort.

Een microcomputer werkt in het algemeen met een instructiecodeeenheid en een programmateller. De instructiecodeeenheid dient voor codering van gehaalde instructies en geeft een stuursignaal af, dat de uitvoering van de gehaalde instructie beveelt. Het door de instructiedecodeeenheid afgegeven stuursignaal bevat een haalcycli-stuurcomponent, welke aan het begin van iedere instructiecyclus een haalcycli voor het halen van de operand voor de op dat ogenblik uitgevoerde instructie en voor het in het midden van iedere instructiecyclus halen van de operatiecode voor de volgende instructiestart.

De onderhavige uitvinding stelt zich ten doel, een microcomputerbedrijfswijze met een verbeterde haalcycli ter verkorting van de instructiecyclus te verschaffen.

De onderhavige uitvinding verschaft hiertoe een werkwijze die wordt gekenmerkt door het voorprogrammeren van instructie-operatiecodes (OP-codes) voor instructies, die meer dan 1 byte instructie-informatie benodigen, met aanvullende informatie, welke informatie een verzoek bevat tot het onder toepassing van tijdbasiscompressie ophalen van instructie-informatie, en door het ophalen van meer dan 1 byte instructie-informatie uit het geheugen tijdens de instructiecyclus die onmiddellijk volgt op het verzoek tot het onder toepassing van tijdbasiscompressie ophalen.

Volgens een voorkeursvorm van de uitvinding bestaat de instructie-informatie uit één operatiecode-byte en één operand-byte en wordt de operatiecode onmiddellijk vóór het begin van een instructie-uitvoeringscyclus opgehaald en bestaat het ophalen uit het gelijktijdig met het begin van de instructiecyclus ophalen van de operand-byte, die behoort bij de opgehaalde operatiecode-byte.

Volgens een andere voorkeursvorm van de uitvinding bestaat de instructie-informatie daarentegen uit één operatiecode-byte en twee operand-bytes en wordt de operatiecode onmiddellijk voor het begin van een eerste instructiecyclus opgehaald en omvat de stap van het ophalen de stappen van het gelijktijdig met het begin van de eerste instructiecyclus ophalen van een eerste operand-byte, welke behoort bij de opgehaalde operatiecode-byte, en van het ophalen van een tweede operand-byte, die behoort bij de eerste operatiecode-byte, gedurende de eerste instructiecyclus die volgt op het ophalen van de eerste operand-byte.

Volgens een verdere voorkeursuitvoeringsvorm van de uitvinding wordt de instructie uitgevoerd, die gespecificeerd wordt door de operatiecode- en operand-byte tijdens de instructiecyclus die onmiddellijk voorafgaat aan het ophalen van de operatiecode-byte.

Volgens een verdere voorkeursuitvoeringsvorm van de uitvinding wordt de instructie uitgevoerd die gespecificeerd wordt door de operatiecode-byte en de eerste en tweede operand-byte tijdens een tweede instructiecyclus die onmiddellijk volgt op de eerste instructiecyclus.

De onderhavige uitvinding zal worden verduidelijkt in de nu volgende beschrijving aan de hand van de tekening van een voorkeursuitvoeringsvorm, waartoe de uitvinding zich echter niet beperkt. In de tekening tonen:

- figuren 1(A) en 1(B) tijdbasisschalen van een instructie-uitvoeringscyclus voor een 2-bytes instructie,
- 5 figuren 2(A) en 2(B) tijdbasisschalen voor een instructie-uitvoeringscyclus voor een 3-bytes instructie,
- figuur 3 een blokschema van een microcomputerstelsel,
- figuur 4 een vergelijking op tijdbasisschaal van een instructie-uitvoeringsprocedure, waarbij (B) de telwaarde van de programmateller tijdens uitvoering van een 1-byte instructie en (C) de desbetreffende telwaarde tijdens uitvoering van een 2-bytes instructie laat zien,
- 10 figuren 5(A) en 5(B) zowel voor uitvoering van een 2-bytes instructie als van een 3-bytes instructie de duur van de instructiecyclus bij informatiebewerking, en
- figuren 6(A) en 6(B) een instructiedecodeereenheid.

In de tekening, te beginnen bij figuur 1(A) en figuur 1(B), zal het principe en het effect van de hieronder
 15 beschreven microcomputerbedrijfswijze steeds in vergelijking met een dergelijke bedrijfswijze van gebruikelijk type worden toegelicht. Figuur 1(A) toont een typisch voorbeeld van een dergelijke bedrijfswijze van bekend type, terwijl figuur 1(B) een vergelijkbare verbeterde voorkeursbedrijfswijze laat zien. Beide tijdbasisschalen hebben betrekking op de uitvoering van een 2-bytes instructie met een 8-bits operatiecode, zoals LDA, en een 8-bits operand, welke in het onderhavige geval wordt gevormd door een adrescode,
 20 welke een te raadplegen (accessed) adres van een geheugen van het RAM-type identificeert. Bij het hier gegeven voorbeeld is de instructie erop gericht de op een specifiek dergelijk adres aanwezige informatie-inhoud naar de accumulator over te brengen, respectievelijk daarin in te voeren.

Bij de gebruikelijke bedrijfswijze volgens figuur 1(A) wordt de operatiecode LDA voor de tweede instructiecyclus vóór het begin van de eerste instructiecyclus gehaald. In de daaropvolgende haalcycli,
 25 samenvallend met het begin van de eerste instructiecyclus, wordt de op het door de operand geïdentificeerde rang-adres aanwezige informatie gehaald. In de tweede instructiecyclus worden de informatie, welke in de gelijktijdig met de eerste instructiecyclus beginnende haalcycli is gehaald, en de operatiecode LDA, welke vóór de eerste instructiecyclus is gehaald, gebruikt.

Bij een dergelijke bedrijfswijze van gebruikelijk type wordt de telwaarde van een programmateller aan het
 30 einde van iedere instructiecyclus met één verhoogd. Als gevolg daarvan heeft de programmateller aan het begin van de haalcycli van de operatiecode de telwaarde PC. Aan het begin van de eerste instructiecyclus wordt de telwaarde van de programmateller met één verhoogd, zodat de telwaarde dan PC+1 wordt. Op soortgelijke wijze wordt de telwaarde van de programmateller aan het begin van de tweede instructiecyclus met één verhoogd tot PC+2.

Bij de in figuur 1(B) weergegeven bedrijfswijze wordt de operatiecode voor de eerste instructiecyclus vóór
 35 het begin van de eerste instructiecyclus gehaald, zoals bij de gebruikelijke bedrijfswijze. Deze 2-bytes operatiecode bevat echter een verzoek tot compressie van de haalcycli. In de periode M_2-M_3 van de eerste instructiecyclus wordt op het door de operand RAM geïdentificeerde RAM-adres aanwezige informatie gehaald. De tijdens de periode M_2-M_3 van de eerste instructiecyclus gehaalde operand-informatie
 40 en de vóór de eerste instructiecyclus gehaalde operatiecode worden voor uitvoering van de eerste instructiecyclus gebruikt. Bij deze bedrijfswijze is de in de periode M_2-M_3 op het RAM-adres gehaalde informatie voor uitvoering van de instructie tijdens de eerste instructiecyclus niet nodig tot de daarop volgende periode M_0-M_1 .

Tijdens een dergelijke bedrijfswijze wordt de telwaarde van de programmateller aan het einde van iedere
 45 instructiecyclus met één verhoogd. Voorts wordt de telwaarde van de programmateller in reactie op iedere klokimpuls M_3 met één verhoogd. Dit wil zeggen, dat de telwaarde van de programmateller aan het einde van iedere haalcycli voor het halen van de operatiecode van PC naar PC+1 wordt verhoogd. In reactie op de klokimpuls M_3 tijdens de eerste instructiecyclus vindt weer verhoging met één plaats, zodat de telwaarde PC+2 wordt. Daarna wordt de telwaarde van de programmateller in reactie op een aan het eind van de
 50 eerste instructiecyclus verschijnende klokimpuls M_1 opnieuw met één verhoogd.

Tijdens de periode M_0-M_1 van de eerste instructiecyclus kan de operatiecode voor de volgende instructiecyclus worden gehaald.

Duidelijk zal derhalve zijn, dat voor uitvoering van een 2-bytes instructie bij de beschreven voorkeurs-
 bedrijfswijze slechts één instructiecyclus nodig is.

55 De figuren 2(A) en 2(B) tonen een ander voorbeeld, dat betrekking heeft op de uitvoering van een 3-bytes instructie. Het desbetreffende voorbeeld heeft betrekking op een subroutine volgens welke de programmabesturing overspringt naar een door twee bytes, dat wil zeggen een HOGHE adres-byte P_H en een

LAGE adres-byte P_L , gespecificeerd programma-adres. De instructie bevat derhalve een 1-byte (8-bits) operatiecode CALL en de twee operand-bytes P_H en P_L .

Bij de gebruikelijke bedrijfswijze volgens figuur 2(A) wordt de operatiecode voor de tijdens de derde instructiecyclus uit te voeren instructie vóór het begin van de eerste instructiecyclus gehaald. Tijdens de volgende haalcyclus, gelijktijdig met de eerste instructiecyclus, wordt de HOGЕ adres-byte P_H gehaald. Op soortgelijke wijze wordt tijdens de tweede instructiecyclus de LAGE adres-byte P_L gehaald. De HOGЕ en LAGE adres-bytes P_H en P_L , welke respectievelijk gelijktijdig met de eerste en de tweede instructiecyclus zijn gehaald, en de vóór de eerste instructiecyclus gehaalde operatiecode CALL worden gebruikt om de subroutine-CALL-instructie tijdens de derde instructiecyclus uit te voeren.

Bij deze bedrijfswijze van bekend type wordt de telwaarde van de programmateller aan het einde van iedere instructiecyclus met één verhoogd. Als gevolg daarvan wordt de telwaarde aan het einde van de haalcyclus voor de operatiecode van PC naar PC+1 verhoogd. Aan het einde van de eerste instructiecyclus wordt de telwaarde weer met één verhoogd, dat wil zeggen tot PC+2. Op soortgelijke wijze wordt de telwaarde van de programmateller aan het einde van de tweede en de derde instructiecyclus steeds met één verhoogd.

Bij de in figuur 2(B) weergegeven bedrijfswijze wordt de operatiecode CALL voor de tijdens de tweede instructiecyclus uit te voeren instructie vóór het begin van de eerste instructiecyclus gehaald. Deze operatiecode CALL bevat een verzoek voor tijdbasiscompressie van de daarop volgende haalcyclus. Tijdens de periode M_2-M_3 van de eerste instructiecyclus wordt de HOGЕ adresbyte P_H gehaald. Tijdens de periode M_0-M_1 van de eerste instructiecyclus wordt de LAGE adres-byte P_L gehaald. De tijdens de eerste instructiecyclus gehaalde adres-bytes P_H en P_L en de vóór de eerste instructiecyclus gehaalde operatiecode CALL worden gebruikt voor uitvoering van de subroutine-CALL tijdens de tweede instructiecyclus.

Bij deze bedrijfswijze wordt de telwaarde van de programmateller aan het einde van iedere instructiecyclus met één verhoogd. Bovendien wordt de telwaarde met één verhoogd in reactie op iedere klokimpuls M_3 , welke in het midden van iedere instructiecyclus valt. Aan het einde van de haalcyclus voor de operatiecode stijgt de telwaarde van PC naar PC+1. In reactie op de in de eerste instructiecyclus verschijnende klokimpuls M_3 stijgt de telwaarde weer met één en wordt PC+2. Daarna wordt de telwaarde van de programmateller in reactie op de aan het einde van de eerste instructiecyclus verschijnende klokimpuls M_1 opnieuw met één verhoogd, enz.

Aan het begin van de tweede instructiecyclus kan de operatiecode voor de volgende uit te voeren instructie worden gehaald.

Zoals hieruit naar voren komt, zijn derhalve bij de beschreven voorkeursbedrijfswijze slechts twee instructiecycli nodig voor uitvoering van een 3-bytes instructie.

Figuur 3 toont een microprocessorstelsel dat is ontworpen voor toepassing van de voorkeursbedrijfswijze.

De voorkeursuitvoeringsvorm van een dergelijk stelsel omvat een programmeerbaar geheugen 1 van het ROM-type, dat wil zeggen van het PROM-type, een informatiebuffer 2, een instructieregister 3, een bijvoorbeeld op een programmeerbare logische rangschikking PLA gebaseerde instructiedecodeereenheid 4, een programmateller 5 een RAM-adresregister 6, en veldeffecttransistoren 7-9.

De werking van een dergelijk microcomputerstelsel zal nu worden beschreven aan de hand van figuur 4. Daarin toont (A) de tijdens iedere instructiecyclus geproduceerde klokimpulsen M_2 , M_3 , M_0 en M_1 , terwijl (B) de telwaarde van de programmateller 5 tijdens uitvoering van opeenvolgende 1-byte instructies laat zien. (C) toont de variatie van de telwaarde van de programmateller 5 tijdens uitvoering van opeenvolgende 2-byte instructies.

De instructiedecodeereenheid levert een programmatelwaardeverhogingssignaal PCUP en enige stuursignalen C_1-C_3 , welke laatstgenoemden de informatiestroom door de microprocessorelementen 1-6 beheersen. De stuursignalen C_1 , C_2 en C_3 worden voor besturing van de veldeffecttransistoren 7, 8 en 9 aan EN-bewerking met respectievelijk de klokimpulsen M_3 , M_1 en M_3 onderworpen. De veldeffecttransistoren 7 en 8 maken in hun geleidende toestand vergrendeling door de informatiebuffer 2 of het instructieregister 3 mogelijk van informatie uit het programmeergeheugen 1 van het ROM-type; deze informatie heeft de vorm van een operand of van een operatiecode. De veldeffecttransistor 9 laat in geleidende toestand de operand in de informatiebuffer 2 tot de informatiebus toe, zodat de operand voor invoer, opslag of dergelijke bevelen in het RAM-adresregister 6 kan worden ingevoerd of voor een subroutine-CALL in de programmateller 5 kan worden ingevoerd.

Wanneer een operatiecode een verzoek tot het halen van een operand onder toepassing van tijdbasiscompressie bevat, geeft de instructiedecodeereenheid 4 zowel een programmatelwaardeverhogingssignaal PCUP als de signalen C_2 en C_3 af, waardoor de operand in reactie op de volgende klokimpuls M_1 in de informatiebuffer 2 kan worden ingevoerd en daaruit worden uitgelezen, en verhoogt de telwaarde van de

programmateller om de volgende operatiecode aan te wijzen.

Bij het voorbeeld volgens figuur 4(B) vindt uitvoering van een 1-byte instructie binnen een enkele instructiecyclus plaats. Het halen van de instructies uit het geheugen 1 van het PROM-type wordt gestart door de achterflank van de klokimpuls M_1 wanneer het uitgangssignaal C_1 van de instructiedecodeereenheid 4 aanwezig is. Op hetzelfde tijdstip wordt de telwaarde van de programmateller 5 in reactie op het reeds genoemde signaal PCUP met één verhoogd. De gehaalde instructie (operatiecode) wordt in het instructie-register 3 ingelezen en vervolgens aan de instructiedecodeereenheid 4 toegevoerd. Deze laatstgenoemde decodeert de gehaalde instructie en geeft de in aanmerking komende signalen C_1 , C_2 en C_3 af.

De achterflank van de klokimpuls M_1 signaleert het begin van de instructiecyclus, waarbinnen de tijdens de voorafgaande instructiecyclus gehaalde instructie wordt uitgevoerd. Gelijktijdig met de uitvoering van de in de voorafgaande cyclus gehaalde instructie wordt tijdens de op dat ogenblik aangevangen instructiecyclus de volgende 1-byte instructie gehaald. Het voor het halen van de instructie te benaderen adres van het geheugen 1 van het PROM-type wordt weergegeven door de telwaarde (PC, PC+1, PC+2), welke bij iedere achterflank van de klokimpuls M_1 door het genoemde signaal PCUP met één wordt verhoogd.

Tijdens de uitvoering van 1-byte instructies blijven, aangezien geen verzoek voor het halen van een operand onder toepassing van tijdbasiscompressie voorligt, de uitgangen C_2 en C_3 van de instructiedecodeereenheid 4 non-actief. Als gevolg daarvan blijven de veldeffecttransistoren 8 en 9 in hun niet-geleidende toestand. Aangezien deze keer niet in het midden van de instructiecyclus het signaal PCUP van de instructiedecodeereenheid verschijnt, wordt de telwaarde van de programmateller 5 slechts in reactie op de achterflank van de klokimpuls M_1 verhoogd.

Zoals figuur 4(C) laat zien, worden voor uitvoering van de 2-bytes instructies met een 1-byte operatiecode en een 1-byte operand de signalen C_2 en C_3 door de instructiedecodeereenheid 4 afgegeven na het halen van de operatiecode van de op de achterflank van de klokimpuls M_1 volgende instructie. De desbetreffende uitgangssignalen C_2 en C_3 brengen de veldeffecttransistoren 8 en 9 in hun geleidende toestand, waardoor de operand uit het geheugen 1 van het ROM-type wordt uitgelezen en in de informatiebuffer 2, de programmateller 5 en/of het RAM-adresregister 6 wordt ingevoerd.

Bij de achterflank van de klokimpuls M_3 neemt de telwaarde van de programmateller 5 in reactie op het van de instructiedecodeereenheid 4 afkomstige signaal PCUP met één toe. Tegelijkertijd wordt bijvoorbeeld de informatie in het RAM-adresregister 6 uitgelezen en voor uitvoering van de instructie in de periode

M_0-M_1 gebruikt.

Tijdens de periode M_0-M_1 wordt de volgende instructie uit het geheugen van het PROM-type gehaald. Aangezien de telwaarde van de programmateller 5 door het signaal PCUP in reactie op de achterflank van de klokimpuls M_3 wordt verhoogd, zal het door de telwaarde vertegenwoordigde PROM-adres overeenkomen met het adres van de operatiecode voor de volgende instructie. Duidelijk zal zijn, dat het ter verkrijging van tijdbasiscompressie bij een haalcycclus, zoals bij de voorkeursbedrijfswijze wordt voorgesteld, noodzakelijk is, dat de tweede of derde informatie-byte geen operatiecodecomponenten bevatten, doch slechts een operand, zoals numerieke informatie of een adres. Sinds de instructies bij de meeste 4-bits microcomputers slechts een 1-byte operatiecode en één of meer operand-bytes omvatten, kan de voorkeursbedrijfswijze bij de meeste microcomputers van dit type worden toegepast.

Zoals de figuren 5(A) en 5(B) laten zien, verschaft de beschreven bedrijfswijze de mogelijkheid om 2-bytes instructies, welke gewoonlijk 2 instructiecycli voor uitvoering vergen, bij een pijpleidingsopbouw binnen een enkele instructiecyclus uit te voeren. Op soortgelijke wijze kunnen 3-bytes instructies, welke gewoonlijk voor uitvoering 3 instructiecycli vergen, in 2 instructiecycli worden uitgevoerd. Dit laat duidelijk zien, dat de voorkeursbedrijfswijze tot een effectieve verkorting van de voor uitvoering van een instructie benodigde duur leidt.

Bovendien tonen de figuren 6(A) en 6(B) schematisch het gebied, dat respectievelijk voor een instructiedecodeereenheid van een microcomputer van beide types nodig is. In de figuren 6(A) en 6(B) vertegenwoordigen de verticale lijnen EN-elementkiesleidingen en vertegenwoordigen de horizontale lijnen OF-elementkiesleidingen. Het symbool m_1 vertegenwoordigt het kiessignaal voor de machinestatus 1, m_2 vertegenwoordigt het kiessignaal voor de machinestatus 2 en m_3 vertegenwoordigt het kiessignaal voor de machinestatus 3.

Zoals uit de figuren 6(A) en 6(B) naar voren komt, zijn bij een gebruikelijk microcomputerstelsel drie machinestatus nodig voor uitvoering van een 3-bytes instructie, hetgeen drie instructiecycli in beslag neemt. Bij de beschreven voorkeursbedrijfswijze zijn echter voor uitvoering van een 3-bytes instructie slechts twee instructiecycli nodig. Als gevolg daarvan zijn slechts twee machinestatus van de instructiedecodeereenheid voor uitvoering van 3-bytes instructies nodig. De OF-elementkiesleiding, welke als stuursignaalleiding voor de machinestatus 3 dient, is in een dergelijk geval derhalve overbodig. Dit resulteert in een aanzienlijke

lengtevermindering van de EN-elementkiesleidingen en levert derhalve een overeenkomstige vermindering van de afmetingen van de instructiedecodeereenheid.

5 Conclusies

1. Werkwijze voor het bedrijven van een microprocessor met een pijpleidingopbouw, welke bedrijfswijze de mogelijkheid biedt om tijdens een aangevangen instructie-uitvoeringscyclus voor een latere instructie-uitvoeringscyclus bestemde instructie-informatie uit een geheugen te halen, gekenmerkt door
10 het vóórprogrammeren van instructie-operatiecodes (OP-codes) voor instructies, die meer dan 1 byte instructie-informatie benodigen, met aanvullende informatie, welke informatie een verzoek bevat tot het onder toepassing van tijdbasiscompressie ophalen van instructie-informatie, en door het ophalen van meer dan 1 byte instructie-informatie uit het geheugen tijdens de instructiecyclus die onmiddellijk volgt op het verzoek tot het onder toepassing van tijdbasiscompressie ophalen.
- 15 2. Werkwijze volgens conclusie 1, met het kenmerk, dat de instructie-informatie uit één operatiecode-byte en één operand-byte bestaat en de operatiecode-byte onmiddellijk vóór het begin van een instructie-uitvoeringscyclus wordt opgehaald, en dat het ophalen bestaat uit het gelijktijdig met het begin van de instructiecyclus ophalen van de operand-byte, die behoort bij de opgehaalde operatiecode-byte.
- 20 3. Werkwijze volgens conclusie 1, met het kenmerk, dat instructie-informatie uit één operatiecode-byte en twee operand-bytes bestaat en de operatiecode-byte onmiddellijk vóór het begin van een eerste instructiecyclus wordt opgehaald, en dat de stap van het ophalen de stappen omvat van het gelijktijdig met het begin van de eerste instructiecyclus ophalen van een eerste operand-byte, welke behoort bij de opgehaalde operatiecode-byte, en van het ophalen van een tweede operand-byte, die behoort bij de eerste operatiecode-byte, gedurende de eerste instructiecyclus die volgt op het ophalen van de eerste operand-
25 byte.
4. Werkwijze volgens conclusie 2, gekenmerkt door het uitvoeren van de instructie die gespecificeerd is door de operatiecode- en operand-bytes tijdens de instructiecyclus die onmiddellijk voorafgaat aan het ophalen van de operatiecode-byte.
5. Werkwijze volgens conclusie 3, gekenmerkt door het uitvoeren van de instructie, die gespecificeerd is
30 door de operatiecode-byte en de eerste en de tweede operand-byte, tijdens een tweede instructiecyclus die onmiddellijk volgt op de eerste instructiecyclus.

Hierbij 4 bladen tekening

FIG. 1 (A)
(PRIOR ART)

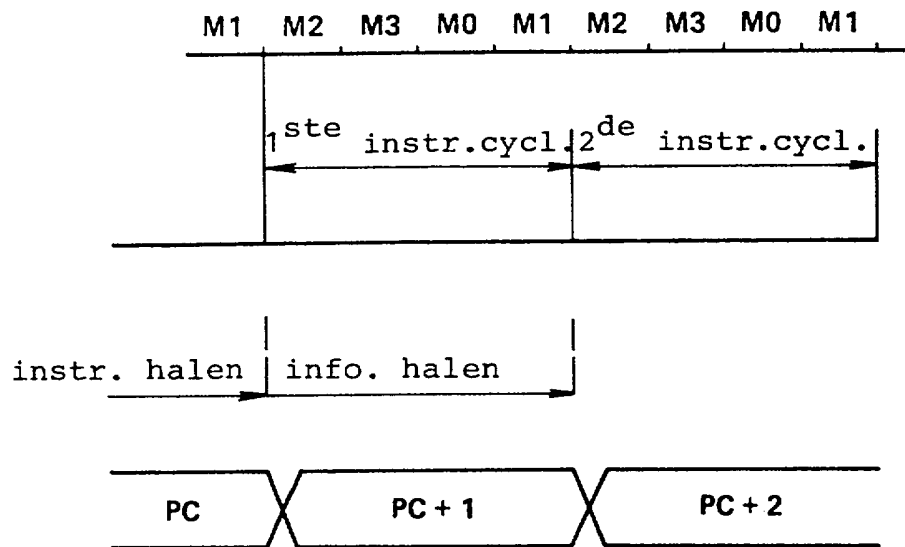


FIG. 1 (B)

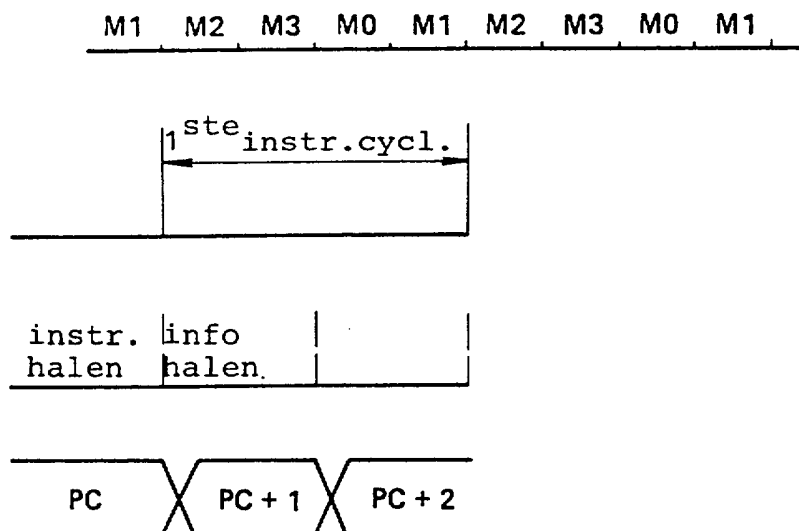


FIG. 2 (A)
(PRIOR ART)

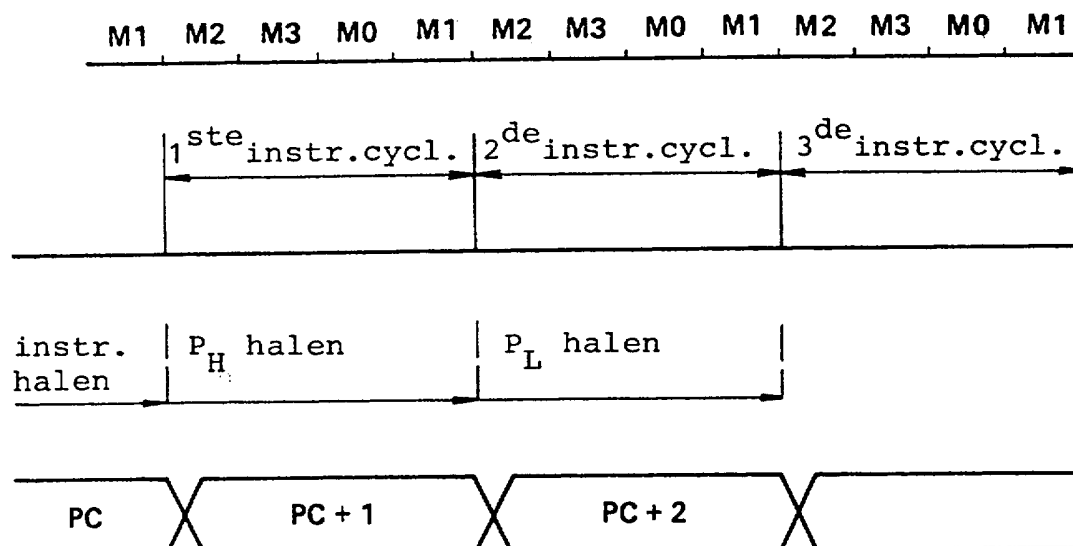


FIG. 2 (B)

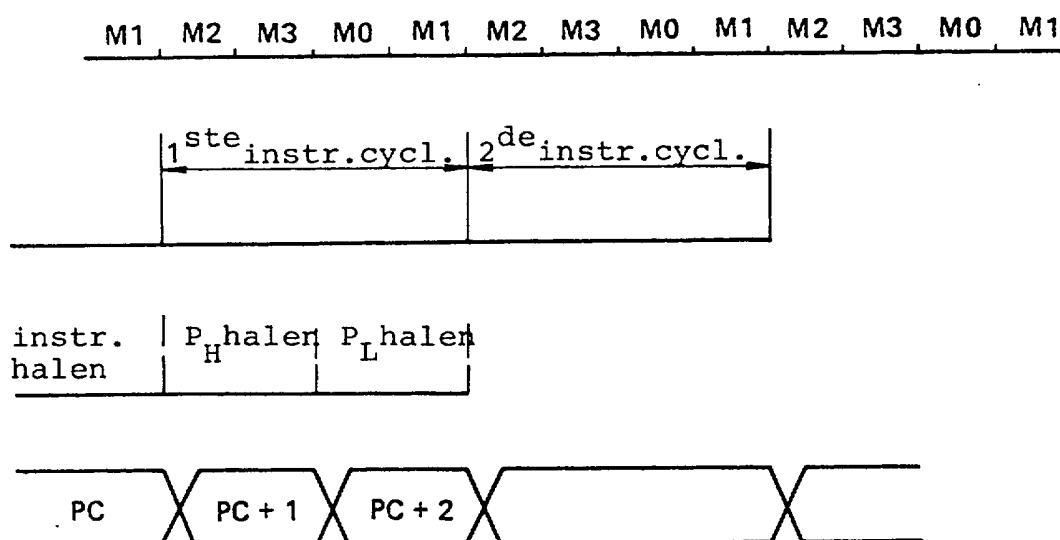


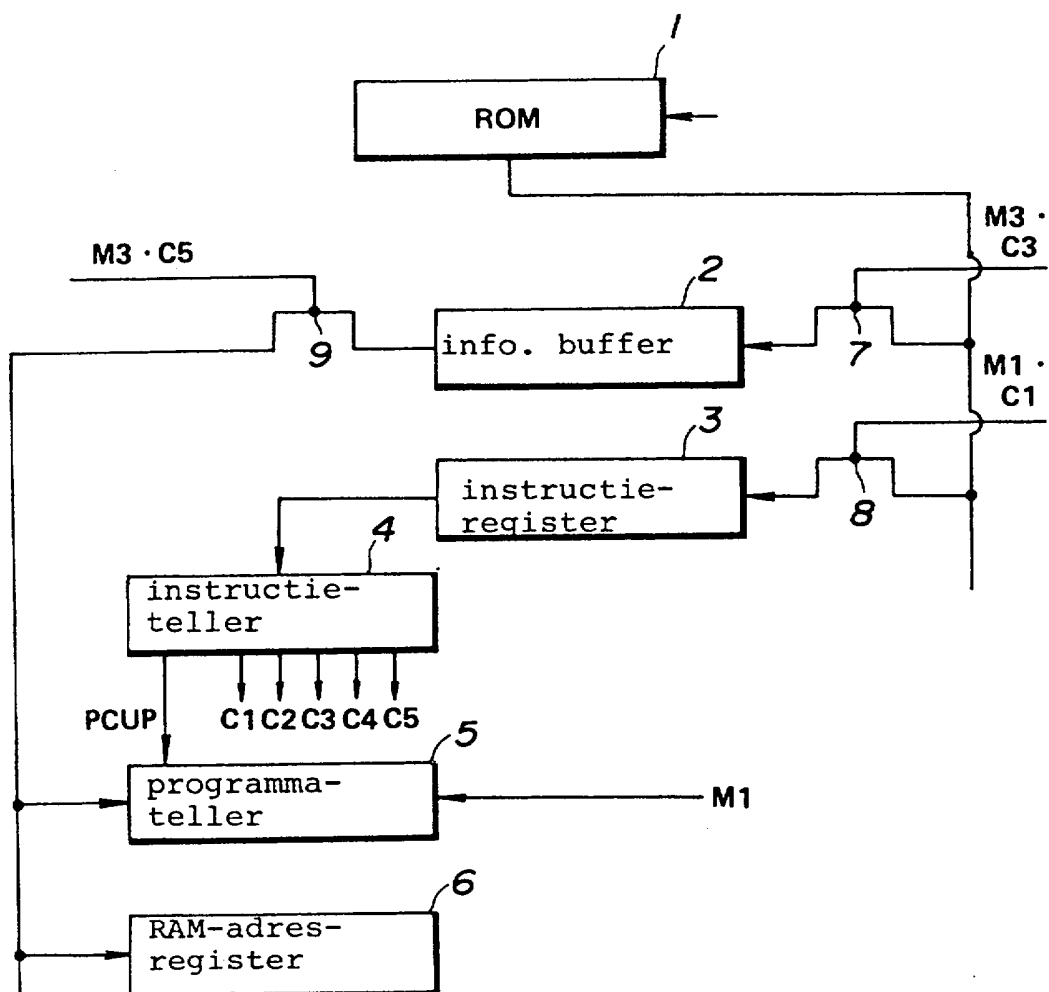
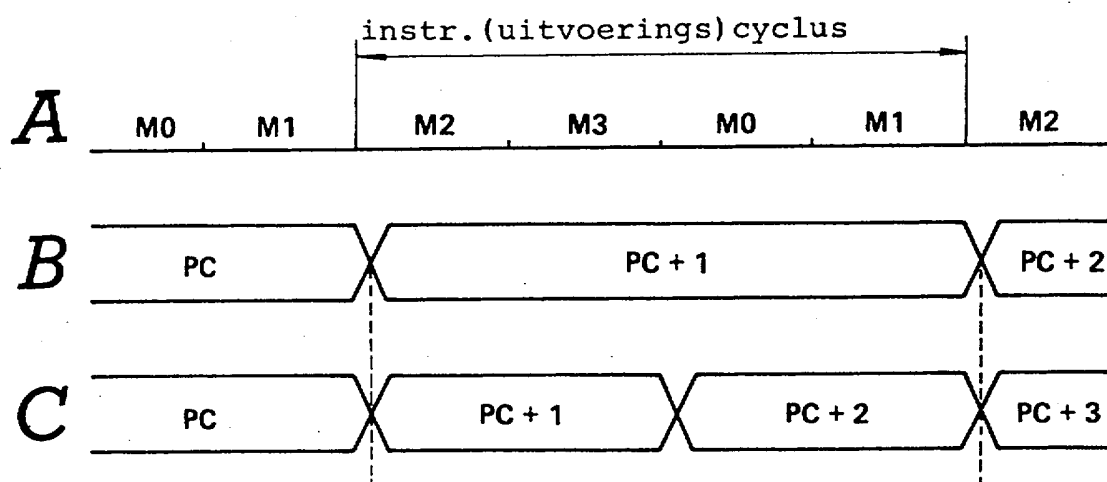
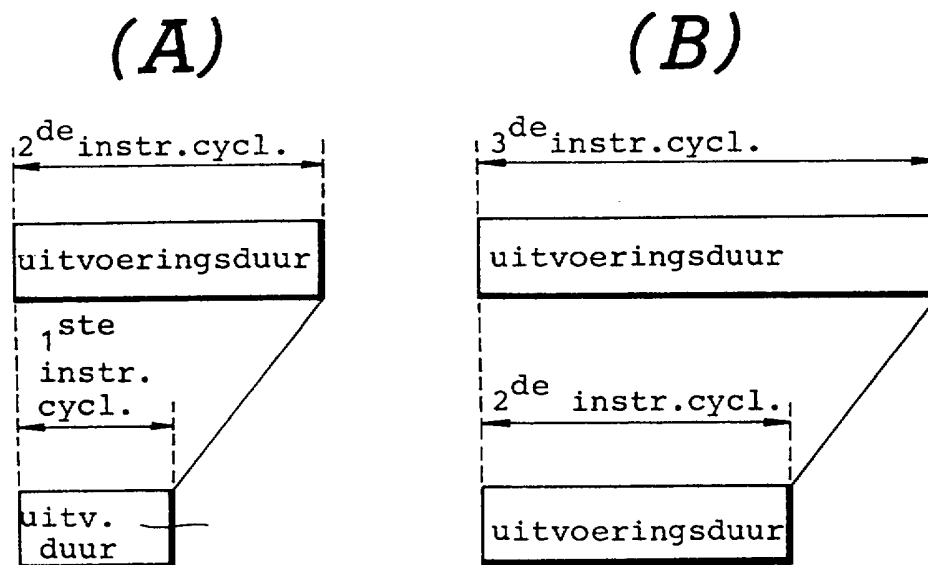
FIG.3**FIG.4**

FIG. 5*FIG. 6*