

MATHEMATISCH CENTRUM

2e BOERHAAVESTRAAT 49

AMSTERDAM

REKENAFDELING

Het standaard-typprogramma van ARMAC

door

E.W. Dijkstra

MR 24

1956

VOORWOORD

Rapport MR 23 is te beschouwen als beschrijving der conventies bij de invoer; dit rapport, dat de uitvoer behandelt is dan ook op te vatten als het direct vervolg op MR 23. Ook dit is geschreven voor de goede verstaander, die aan een half woord genoeg heeft en uit behoefte aan snelle documentatie. Als, naar wij hopen, dit communicatieprogramma de toets der praktijk heeft kunnen doorstaan, zal een gedetailleerdere beschrijving volgen.

I N H O U D

	pag.
Het standaardprogramma	1
Het opstellen van het codewoord.	2
Het inbrengen van het codewoord.	4
Pagina-layout.	5
Regelindeling	5
Blokindeling.	5
Paginaindeling.	6
Telling van het aantal TWNR-signalen	7
De controle.	8
De automatische plaatsing der tabulatorstoppen	9
Résumé der verwijzingen naar het geheugen.	9
Résumé der autostarts.	10

Het standaard-typprogramma van ARMAC

Het meest versatiele standaardprogramma van ARMAC is de typ-routine, die de inhoud van S typt. Een van de functies hiervan is het uit de gegeven binaire representatie uitrekenen van de opeenvolgende decimale cijfers; deze zijn afhankelijk van de plaats, waar de binale komma in het woord (S) wordt geïnterpreteerd. De typroutine kent hiervoor twee mogelijkheden: of achter het laagste cijfer (d.w.z. typ (S) als geheel getal) of onmiddellijk achter het tekencijfer (d.w.z. typ (S) als breuk). De onderscheiding in "geheel getal" en "breuk" slaat op de interpretatie van het binaire woord in S, niet op het beeld op de pagina! Het is namelijk aan de programmeur om te bepalen, of, en zo ja voor welk cijfer, de punt getypt wordt. Voorts kan of de absolute waarde, of het getal met teken getypt worden; ook het aantal cijfers staat de programmeur ter beschikking. Deze en nog enige andere vrijheden worden voor elk geval vastgelegd in een speciaal codewoord. Als de zg. centrale typroutine wordt aangeroepen, moet het codewoord ingevuld zijn op adres 0X0. Dit invullen geschiedt als regel door een van de tien zg. speciale typaanroepen. Op het ogenblik bevat het standaard typ-programma ruimte voor tien codewoorden, genummerd van 0 t/m 9. In het geheugen zijn hiervoor gereserveerd (in volgorde) de adressen 4080 t/m 4089. De speciale aanroepen (22 0X28, 22 1X28 22 9X28) plaatsen het overeenkomende codewoord in adres 0X0 en geven dan de besturing over aan de centrale typroutine.

Zo typt b.v. de aanroep

```
10 13 B2
22 5 X28=) "typcode 5"
```

de inhoud van 13 B2 volgens het codewoord 5 in adres 4085.

Als tien verschillende codewoorden genoeg is, worden zij bij de invoer in adressen 4080 t/m 4089 ingevuld, waarmee de functie der speciale aanroepen vast ligt. Als men aan tien verschillende codewoorden niet genoeg heeft, kan b.v. het hoofdprogramma de betekenis van "typcode 9" wijzigen, door het nieuwe codewoord op adres 4089 in te vullen.

Opm.1: Het is essentieel, dat de 10-opdracht onmiddellijk gevolgd wordt door de sprong naar de typroutine, zoals in bovenstaand voorbeeld. Deze restrictie is een gevolg van de manier, waarop het typen gecontroleerd wordt. (zie later).

Opm.2: Als men aan tien vaste typcodes niet genoeg heeft, kan men ook het hoofdprogramma het gewenste codewoord in 0X0 laten zetten, en direct de centrale typroutine aanroepen. Aangezien

dit op een nog al speciale manier zou moeten gebeuren, wordt dit nu niet beschreven.

Opm.3: Alle typroutines voor getallen werken in de veronderstelling, dat de schrijfmachine staat op "kleine letter"; zij laten de schrijfmachine in deze stand achter.

Het opstellen van het codewoord.

Als het teken getypt moet worden, begint het codewoord met T; als het teken onderdrukt moet worden - de typroutine typt dan de absolute waarde - laat men deze T weg.

Hierna schrijft men S, als na (al of niet) het teken een spatie ingelast moet worden; ontbreekt de S in het codewoord, dan wordt er geen spatie ingelast.

Hierna komt Gn of Bn. In het geval Gn ($1 \leq n \leq 10$) wordt (S) opgevat als geheel getal van n decimale cijfers; het n^{de} cijfer komt overeen met het eenhedencijfer. Als $n \leq 9$, moet [S] in absolute waarde kleiner zijn dan 10^n . In het geval Bn ($2 \leq n \leq 10$) wordt (S) opgevat als breuk van n decimale cijfers; het 1^{ste} cijfer komt overeen met het eenhedencijfer, de breuk wordt exact op het laatste cijfer (dus de $n-1^{\text{ste}}$ decimaal achter de komma) afgerond. Door deze afronding kan het eerste cijfer = 1 worden; als regel is het natuurlijk = 0. Omdat $n \geq 2$, komt dus minstens één cijfer achter de komma.

Elk dezer n cijfers kan op drie wijzen verwerkt worden.

- a) "Loos": als van te voren bekend is, dat een of meer van deze cijfers = 0 zijn en wij ze niet op het papier wensen te zien, kunnen ze geheel onderdrukt worden. Als een geheel getal b.v. een exact 1000-voud is, maar we in deze nullen niet geïnteresseerd zijn (omdat ze b.v. een gevolg zijn van een om programma-technische redenen ingelaste factor 1000) eindigt het typproces met drie slagen in de zg. "loze cyclus".
- b) "Imperatief", d.w.z. het verwerkte cijfer wordt (zonder meer, zie ad c) getypt.
- c) "Facultatief", d.w.z. als het verwerkte cijfer $\neq 0$ is, wordt het, benevens alle eventuele volgende cijfers, als bij "imperatief verwerken" behandeld; als het verwerkte cijfer = 0 is, en de "overgang naar imperatief" heeft nog niet plaats gevonden, dan wordt het door een spatie vervangen.

Opm.1: De controle typroutine heeft twee standen: de imperatieve en de facultatieve; in de laatstgenoemde worden nullen door spaties vervangen. Tenzij anders gespecificeerd begint het typprogramma in de facultatieve stand; het kan naar de imperatieve

stand overgaan of doordat een cijfer $\neq 0$ verwerkt wordt, of door expliciete aanduiding in het codewoord. De overgang terug naar weer facultatief is in het zelfde getal niet meer mogelijk.

Opmerking 2: Men realiseer zich de verschillen tussen loos en facultatief: bij facultatief wordt het cijfer, dat $=0$ blijkt te zijn, door een spatie vervangen; bij loze verwerking wordt er geen enkel signaal naar de typemachine gezonden. Verder betreft de loze verwerking cijfers, die gegarandeerd $= 0$ zijn, de facultatieve echter cijfers, die misschien $= 0$ zijn.

Bovendien kan men desgewenst een punt of een spatie inlassen. De punt bewerkstelligt, als nog facultatief getypt wordt, de overgang naar imperatief voor de volgende cijfers; de spatie laat (evenals een aantal loze verwerkingen) deze stand onveranderd.

Preciezer: na Bn of Gn kunnen wij kiezen uit de volgende specificaties (het in het codewoord beschikbare aantal cijfers laat maximaal zes achtereenvolgende keuzen toe).

F_n ($n \geq 1$) typ n facultatieve cijfers

S_n ($n \geq 1$) typ spatie, gevolgd door n facultatieve cijfers

I_n ($n \geq 1$) typ n imperatieve cijfers

P_n ($n \geq 1$) typ punt, gevolgd door n imperatieve cijfers

L_n ($n \geq 1$) verwerk n loze cijfers.

Omdat steeds $n \geq 1$ is, is het niet mogelijk, om b.v. vlak voor of na een punt een spatie in te lassen, twee punten of spaties onmiddellijk achter elkaar in te lassen, etc.

Na afloop wordt het codewoord afgesloten door:

XK, als hier het getal klaar is

XS, als het door een spatie gevolgd moet worden

XT, als het door een tabulatie gevolgd moet worden.

Enige voorbeelden mogen dit toelichten:

T	G10	S9	J1	XK	} typ geheel getal van 10 cijfers na teken en spatie; de eerste negen cijfers facultatief de laatste imperatief.
T S	G10	F9	J1	XK	
(T S	G10	F4 F5	J1	XK)	

Het onderste voorbeeld illustreert hoe het ook kan, maar niet moet.

T B10 F1 P3 S3 S3 XT: typ teken, gevolgd door breuk met 1 cijfer facultatief voor de komma - dit is bijna altijd een spatie - gevolgd door punt en negen cijfers, in groepjes van 3 door spaties gescheiden. Na afloop Tab.

Typ een bedrag, in centen gegeven, uit in guldens; het bedrag wordt zonder teken getypt, is minder dan f 1.000.000, de zes cijfers voor het guldenaantal worden door een spatie in twee maal drie verdeeld. Als het getal door een spatie afgesloten moet worden, luidt het antwoord: G8 F3 S2 J1 P2 XS. Tussen gulden- en centenaantal wordt dus een punt getypt, het laatste cijfer voor de punt wordt altijd getypt. Als we weten, dat het bedrag een geheel aantal guldens is en de punt, gevolgd door twee nullen niet getypt moet worden, zou het codewoord G8 F3 S2 J1 L2 XS luiden.

Als we van een bedrag b.v. het duizendvoud uit willen typen, maar weten, dat dit aantal kleiner is dan 10, en nog drie cijfers achter de komma willen hebben, dan luidt het codewoord:

B7 L2 F1 J1 P3 XT. (Dit is te verkiezen boven S F7 L3 J1 P3 XT als nl. het duizendvoud door afronding = 10.000 kan worden. Als regel effectueren we de extra spatie bij breuken zo mogelijk als facultatief getypte nul). In dit voorbeeld wordt het teken onderdrukt en na afloop Tab gegeven.

Het inbrengen van het codewoord

De lezer zal gemerkt hebben, dat voor de opstelling van het codewoord slechts gebruikt is gemaakt van symbolen die op het toetsenbord voorkomen. Men pons het codewoord van links naar rechts, alle aantallen met 1 pentade ponsend. Aan het begin van elk codewoord worden een willekeurig aantal (mag = 0 zijn) pentades X geskipt. Het invoerprogramma assembleert pentades tot codewoorden, als de soortspecificerende controle-combinatie RT ("Typcode") vooraf is gegaan. Inmiddels kent het invoerprogramma dus vier soorten moleculen. De controle-combinatie RT laat (evenals RD en RG) de wisselstand en de plaats van wegbergen onbeïnvloed; het invoerprogramma blijft "gevoelig" voor alle controlecombinaties.

Opm.: De som van alle aantallen na F,S,J,P en L is in elk codewoord gelijk aan het totale aantal typbare cijfers (na B of G). Deze gelijkheid wordt door het invoerprogramma gecontroleerd; ingeval van onjuistheid stopt de machine. Dit is slechts een van drie controles, waaraan de band bij het lezen van typcodes wordt onderworpen.

Pagina-layout

Regelindeling

Aan het begin van elk getal moet een tabulatorstop staan, ook aan het begin van het eerste; er komt dus een tabulatorstop een paar plaatsen rechts van de kantlijn te staan. Links van de tabulatorstop van het meest rechtse getal op de regel mogen niet meer stoppen staan, dan de bovengenoemde.

Normaal wordt een vast aantal (zeg i) getallen per regel getypt. In dit geval wordt de zg. regelparameter, waar adres 4077 in het geheugen voor gereserveerd is, tydens de invoer = i gemaakt. Dan geeft de centrale typroutine na elk i^{de} getal het signaal TWNR en na een vertraging - om de wagen tijd te geven, terug te lopen - een Tab-signaal, zodat dan de wagen klaar staat voor het typen van het eerste getal op de volgende regel. Als we in een van de kolommen een plaats open willen laten, doen we dit met de aanroep

22 6 X25 =) "acht getal geteld getypt".

Door een Tab-signaal loopt de wagen door tot aan de tabulatorstop voor de volgende kolom; (de telling van) het aantal nog te typen getallen op diezelfde regel wordt met 1 verlaagd. Dit Tab-signaal komt in dit geval altijd, ook als men het laatste getal op de regel wil overslaan. Men moet daarom rechts op de wagen nog minstens een extra tabulatorstop zetten.

Blokindeling

Vaak is het gewenst, dat de regels in zg. blokken van een vast aantal (zeg j) onder elkaar getypt worden, terwijl de afzonderlijke blokken door een extra regel blank gescheiden worden. Indien dit gewenst is, maakt men de zg. blokparameter, in adres 4078, gelijk aan j, d.w.z. het aantal regels per blok. Als men de blok-indeling wil onderdrukken, zet men de blokparameter j = 0. Blokindeling onderdrukken wil zeggen, dat de regels niet geteld worden. In dit geval is er geen verschil tussen de beide volgende aanroepen:

23 1 X26 =) "acht regel geteld voltooid"

en

22 3 X26 =) "acht regel ongeteld voltooid"

In beide gevallen wordt TWNR-signaal en Tab-signaal, gescheiden door een vertraging, gegeven en wordt de telling van de getallen ingesteld op het begin van de regel. Als $j \neq 0$, dan telt de op

deze manier afgemaakte regel in het eerste geval wel, in het tweede geval niet mee in de blokindeling.

Er zijn dus twee methoden, om de blokindeling te onderdrukken: of men zet de blokparameter $j = 0$, of men stelt de regelparameter in op een groter aantal getallen per regel, dan ooit voorkomt, en geeft vanuit het hoofdprogramma het einde van elke regel aan met de aanroep 22 3 X26.

Paginaindeling

Indien blokken bestaan uit een vast aantal regels, kan het gewenst zijn, een vast aantal (zeg n) blokken op een pagina te typen. Voor het maken van een tafel wil men b.v. de pagina's scheiden, door een aantal extra regels blank en aan de bovenkant van de nieuwe pagina wil men misschien opschriften boven de kolommen typen etc. Dit regelt men met de zg. paginaparameter k , waarvoor in het geheugen adres 4079 gereserveerd is. Er zijn voor deze parameter 3 gevallen:

- $k = 0$: de paginaindeling treedt niet in werking, blokken worden dus niet geteld.
- $k = 2n-1$: na n blokken wordt nog een extra regel blank ingelast (dus totaal twee, nl. ook al een, omdat het einde van een blok bereikt was; dit impliceert drie TWNR-signalen!)
- $k = 2n$: na n blokken wordt nog geen extra (tweede) regel blank ingelast, want de lay-outroutine handelt, alsof er nog een nieuw $n+1^{\text{ste}}$ blok onderaan de pagina komt, bestaande uit slechts één regel. Dit arrangement stelt ons in staat, om, wat bij sommige tabellen gewenst is, de eerste regel van elke pagina ook onderaan de vorige bladzijde te typen.

Alleen aan het einde van de pagina komt de besturing met de conditie positief terug in het hoofdprogramma. Hierdoor is de programmeur in staat, aan het begin van de nieuwe pagina zijn maatregelen te treffen, als daar zijn het geven van extra regels blank, het typen van pagina-nummer, hoofden boven de kolommen en, als $k = 2n$, het herhalen van de onderaan de vorige pagina reeds getypte regel.

Naar analogie met twee eerder gegeven aanroepen, kent het invoerprogramma:

22 4 X26 =) "acht blok geteld voltooid"

en

22 5 X26 =) "acht blok ongeteld voltooid"

Beide sturen naar de schrijfmachine twee TWNR-signalen, en dan een Tab-sigitaal, alle gescheiden door vertragingen. In het eerste geval telt het zo beëindigde blok wel, in het tweede geval niet mee in de paginaindeling. Om de paginaindeling te onderdrukken, staan ons weer twee wegen open; de paginaparameter $k = 0$ te stellen is zeker het eenvoudigst.

Een verplichte aanroep aan het begin van elk programma is
22 0 X26 =) "start nieuwe pagina"

Hier worden twee TWNR-signalen, gevolgd door een Tab-sigitaal naar de schrijfmachine gezonden en alle tellingen worden gezet.

Telling van het aantal TWNR-signalen

Als pagina's op het papier niet een vast aantal regels beslaan (zie ook later bij de beschrijving der controle) en toch - als b.v. de schrijfmachine met "pinfeed" is uitgerust - de eerste regels van pagina's aequidistant moeten liggen, maken wij gebruik van het feit, dat het totale aantal TWNR-signalen geteld wordt. Als een nieuwe pagina aan bod komt, wordt deze telling uitgelezen en $= 0$ gezet voor de telling van het aantal op de nieuwe pagina. Het oude aantal staat bij de terugkomst in het hoofdprogramma in S.

Opm.1: Bij paginascheiding geeft de subroutine drie signalen TWNR; eerst één, omdat een pagina klaar is (die valt nog onder de oude telling), dan één, omdat op de nieuwe pagina een nieuw blok aan de beurt is, dan nog één, omdat dit tevens een nieuwe regel impliceert. De laatste twee vallen onder de nieuwe telling

Opm.2: Ook als de pagina-indeling met behulp van $k = 0$ onderdrukt wordt, wordt aan het einde van een blok de totale regeltelling uitgelezen en $= 0$ gezet; is hier de paginaindeling met $k = -0$ onderdrukt, dan is aan het einde van een blok ook de conditie positief bij terugkomst in het hoofdprogramma.

Opm.3: Ook bij de aanroep 22 0 X26 ("start nieuwe pagina") wordt het oude aantal TWNR in S geplaatst, en wordt de nieuwe telling $= 0$ gezet, vóórdat de twee TWNR-signalen van deze aanroep gegeven worden. Als de besturing in het hoofdprogramma terugkomt, is dus ook hier de nieuwe telling alweer tot 2 gestegen.

De aanroepen

22 0 X24 =) "tel en geef TWNR, waarna vertraging"
en

22 3 X24 =) "geef Tab, waarna vertraging"
staan de programmeur tevens ter beschikking.

TWNR is aanmerkelijk groter dan na Tab.

De controle

Het is gezegd, dat de 10-opdracht, die het te typen getal in S plaatst, onmiddellijk aan de subroutineaanroepsprong vooraf moet gaan. De reden hiervoor is, dat na het typen de besturing niet in het hoofdprogramma op de volgende, maar op de vorige opdracht terugkomt! Hier wordt weer het getal in S geplaatst; in de typroutine wordt nu de besturing naar een controleprogramma gestuurd, waar het ten tweede male aangehaalde getal vergeleken wordt met een uit de getypte symbolen opgebouwd controleresultaat. (Bij deze controle wordt ook het codewoord opnieuw opgezocht en ontrafeld) Als de controle klopt, komt de besturing via het layoutprogramma bij de volgende opdracht in het hoofdprogramma terug. Faalt evenwel de controle, dan komt er TWNR, en zoveel Tab-signalen, dat de wagen weer in de positie staat, waarin hij stond, toen aan het mislukte getal begonnen werd, en de besturing komt terug in het hoofdprogramma bij de vorige opdracht, (dus de 10-opdracht) en probeert opnieuw het getal te typen. De extra TWNR telt niet mee in de blokindeling. Typen en controleren wisselen elkaar dus af (ook als een fout is gedetecteerd). Daarom is de aanroep 22 0 X26 ("start nieuwe pagina") verplicht: hier wordt gezorgd, dat de eerstvolgende keer getypt (en niet gecontroleerd) wordt. De achtergrond van het voorschrift over de plaatsing der tabulatorstoppen is nu ook begrijpelijk.

De automatische plaatsing der tabulatorstoppen

Als a,b,c, ... n de afstanden zijn van elke tabulatorstop tot de linkerkantlijn, resp. de voorafgaande tabulatorstop, ponst men het zg. Tab-bandje

RJ 6 22 X25

+ a

+ b

+ c

⋮

+ n

RE roffel blank

Men verwijdt voor het inlezen de tabulatorstoppen en geeft TWNR (dus zet de wagen tegen de linkerkantlijn). Als het boven aangegeven bandje gelezen wordt, geeft de machine a spaties

en stopt; men kan dan een tabulatorstop inzetten; na doorstarten (onafhankelijk van trommel - buffer schakelaar) geeft het programma spaties en stopt; etc. Na doorstarten na de stop na n spaties wordt de nu volgende controlecombinatie gewoon gelezen. Voor elk getal worden eventuele pentades X normaal geskript. Het verdient aanbeveling het Tab-bandje op dezelfde band te laten volgen door het bandje met de typconstantes (d.w.z. layout-parameters en typcodes).

Het Tab-bandje laat de stand van de controlewissel onbeïnvloed.

Résumé der verwijzingen naar het geheugen

Voor de invoer van gegevens:

RJ 6 22 X25: aankondiging van Tab-bandje
 4077X0: adres voor regelparameter
 4078X0: " " blokparameter
 4079X0: " " paginaparameter
 4080X0: " " typecode 0
 ...
 4089X0: " " typecode 9

Voor subroutineaanroepen in het programma:

22 0 X28=): typ volgens typecode 0
 ...
 22 9 X28=): typ volgens typecode 9
 22 0 X24=): tel en geef TWNR, waarna vertraging
 22 3 X24=): geef Tab, waarna vertraging
 22 0 X26=): start nieuwe pagina
 22 6 X26=): acht getal geteld getypt
 23 1 X26=): acht regel geteld voltooid
 22 3 X26=): acht regel ongeteld voltooid
 22 4 X26=): acht blok geteld voltooid
 22 5 X26=): acht blok ongeteld voltooid
 22 26 X19=): pons biband (parameter in S meegeven: in de a-helft het begin, in de b-helft de lengte, geponst wordt RA-combinatie, RB, binaire moleculen en dan 1 pentade blank)
 22 18 X25=): controleer het gehele trommelgeheugen op de paritycheck (gebruikt kanaal 0).

Résumé der autostarts

- "0": start invoerprogramma schrijvend
- "1": start invoerprogramma vergelijkend
- "2": typ A als geheel getal (volgens T G10 S9 J1 XS)
- "3": typ S als geheel getal (volgens T G10 S9 J1 XS)
- "4": typ A als breuk (volgens T B10 F1 P9 XS)
- "5": Typ S als breuk (volgens T B10 F1 P9 XS)
- "6": test trommelgeheugen
- ("7": nog ongebruikt)
- "8": herstel vrij kanaal (= kanaal 127: de voorponsing wordt
bedorven)
- "9": pons biband (begin in a-helft, lengte in b-helft van de
getalschakelaar).