

IBM: Er wordt ons aangeboden een IBM 7040 of een IBM 7044. Het meest opvallende van deze offerte is dat hij niet zo heel veel met onze specificaties te maken heeft waar het communicatie-apparatuur en backing store betreft. De offerte wordt dan ook begeleid met een indrukwekkende brief van 14 kantjes, waarin eigenlijk uitgelegd wordt, dat wij beter gebaat zijn bij wat de IBM kan aanbieden dan bij wat we gevraagd hebben.

Op vier punten wijkt de offerte wezenlijk af van wat we gevraagd hebben, te weten multiprogramming, load-and-go-software, de afwezigheid van magnetische banden, en ponsband inplaats van ponskaarten.

Naar aanleiding van het interrupt system hebben we geschreven "We feel inclined to stress the point that multiprogramming techniques should be more than a theoretical possibility, because.....". en dan de motiveringen. Hier tegenover komt het antwoord:

"The use of multiprogramming to run simultaneously several independent programs is often advised, in order to improve the efficiency of the utilisation of the equipment. In this relation it is worth~~XXX~~while to point out two other concepts, that are much less discussed in literature but serve the same purpose and are used to a great extent in many installations with our customers:

1) the use of off-line equipment for conversion of data from paper tape or punched cards to magnetic tape and the conversion from results obtained with the main computer on magnetic tape to printed reports, paper tape or punched cards. The main installation is then a so-called tape-oriented system. The very high speeds of magnetic tape for input and output facilitate an optimal use of the main computer. Dismounting and remounting of tape reels can be reduced if tape switching equipment is used.

2) the use of a monitor system by which a series of jobs can be executed in sequence without intervention of a machine operator between the jobs. In this way the set up times between jobs are eliminated. In the environment of a University installation there ~~XXX~~ always will be numerous different programs under development. These programs require during their development many short periods of computer time for testing. All these small jobs can be stacked in advance and run then in sequence under the control of the monitor. You will find more about this concept in the section on software."

Oit is nu net, wat we niet wilden. Essentieel in deze organisatie is off-line apparatuur van en naar magnetische band en dat zijn hele machines. Een bijkomstig nadeel is, dat deze conversie meestal gepaard gaat met een of andere co-deconversie, die je ~~XXX~~ vooral bij de invoer -transport naar magnetische band extra beperkingen pleegt op te leggen, zoals bv. een vast "terminal character" op het primaire invoermedium. Zulk soort dingen zijn prohibitief hinderlijk.

Erger nog dan dat de organisatie als geheel te log wordt: wat je in zo'n geval nl. doet is eerst programma's op een tape opnamelen, dan met die tape naar de machine gaan, de monitor verzorgt de "scheduling" en dan gaat de machine aan het werk, het ene programma na het andere. Deze organisatie vermindert de "instantane beschikbaarheid" van de machine aanzienlijk: elk programma blijft in principe liggen totdat het in de volgende "batch" opgenomen wordt. Ik ben nog steeds vurig van plan het ideaal "klaar terwijl U wacht" aanzienlijk dichterbij te benaderen.

Over "load and go" hebben wij ook verschillende ideeën. Ik wil de programma's formuleren in een passende taal en die gaan zo de machine in: of de machine die tekst in twee fasen verwerkt, moet de machine weten, maar dat gaat de gebruiker in geen geval aan: de volgende keer, dat hij zijn programma wil uitvoeren, komt hij weer met zijn programmatext in de oorspronkelijke "source language". Dit is in lijn-

rechte tegenstelling ~~XXX~~ tot de FORTRAN-philosophie en tegen die achtergrond moeten we de aanbeveling voor ponskaarten dan ook zien, die o.a. luidt: "To file compiled programs on punched cards is not unattractive. In one card 20 instructions in binary form can be stored." en evenzo "If a program has to be assembled from a new part and subroutines that are already available in punched card form in object language, one deck of punched cards can be composed to feed to the computer. If different pieces of paper tape are fed to the main computer there will be a loss of computing time, unless another program is in storage to take over during set-up-times of paper tape." Ik versta onder "load-and-go", dat je aan vertaalde programma's, gescheiden van hun uitvoering, geen bestaansrecht op permanente media toekent. De hemel bespare ons voor de administratieve rompslomp, die je krijgt, wanneer je van elk programma onvertaalde, vertaalde en half-vertaalde versies hebt rondslingeren. Een en ander impliceert, dat ik ~~XXXX~~ -helemaal afgezien van onze ervaringen op dit ~~XXXXXX~~ gebied met de 1620- niet erg onder de indruk ben van de aanbeveling van hun software.

Onze voorkeur naar ponsband boven ponskaarten werd in de "Letter" uiteindelijk gemotiveerd door "The main objection is that their inclusion tends to introduce unhappy conventions that go back to the time when punched cards were processed by less flexible machinery, but which cannot be justified any longer (e.g. the superstition that only a single alphabet should be allowed)." Hiertegenover antwoordt de IBM "As long as no extended alphabet is generally accepted we use a 64-character set. This is in complete agreement with the 6-bit byte structure used throughout the equipment (sic!)". Deze opmerking is ronduit schunnig, wanneer je er bij weet, dat de afwezigheid van "general acceptance" te wijten is aan.... obstructie van de IBM in de speciale commissie van de Computer Manufacturers Association! De opmerking dat het 64-character alphabet "in complete agreement" met hun apparatuur is, moet dan ook waarschijnlijk uitgelegd worden, dat een characterset van meer characters in "disagreement" met hun apparatuur is.

In dit verband wil ik citeren de brief, die Dr.H.Haller van de "Kommission Fuer Rechenanlagen der Deutschen Forschungsgemeinschaft" als reactie op de "Letter" geschreven heeft:

"Sehr geehrter Herr Professor Dijkstra!

Fure die Uebersendung Ihres Letter to the Manufacturers danke ich Ihnen verbindlichst. Ich habe Ihre Spezifikationen mit Vergnuegen gelesen und stimme in allem voll mit Ihnen ueberein. Eine kleine Ausnahme bildet Ihre Ansicht ueber Magnetbaender, von denen ich glaube, dass man sie wegen der grossen Speicherkapazitaet in einer geringen Anzahl von Exemplaren (1-2) acht dann versehen sollte, wenn man ueber einen grosseren backstore verfuagt, der in der Speicherhierarchie doch einen anderen Platz (Kapazitaet, Geschwindigkeit) einnimmt.

Was Ihre Ansicht ueber Lochkarten anbelangt, so kann ich Sie nur voll unterstuetzen. Ich bin neugierig, wie die befragten Hersteller zu Ihren Spezifikationen Stellung nehmen werden und ich hoffe, im Laufe des Fruehjahres Gelegenheit zu haben, mich mit Ihnen ueber diese Frage persoendlich unterhalten zu koennen.

Mit den besten Gruessen, Ihr sehr ergebener
Dr.H.Haller "

Hoewel deze man met de brief, op een punt na, instemt, neemt hij nog apart de moeite om zijn instemming met onze afwijzing van ponskaarten expliciet uit te spreken. Ik citeer deze brief niet, om mij achter de autoriteit van Haller te verschuilen, wat evenwel niet wegneemt, dat ik het prettig vond om deze brief te krijgen. (Met zijn opmerking, dat je voor een enkele magneetband nog wel emplooi hebt, ben ik het bij nader inzien en door discussie met een vriendje bij English Electric wel eens; maar dat is dan nog heel iets anders als "a tape oriented

system").

Backing store wordt geleverd in de vorm van schijven; de capaciteit hiervan is indrukwekkend, nl. 160 miljoen bits. De wachttijd, die optreedt bij verplaatsing van de armen (hoewel indrukwekkend, als je bedenkt, dat hier een batterij lees-schrijfkopjes mechanisch verplaatst wordt en binnen nauwe toleranties in zijn uiteindelijke bestemming gepositioneerd wordt) is 50 tot 180 ms en dat is me wat aan de hoge kant. Ik ben bang, dat deze wachttijden niet te verwaarlozen zijn, en dat efficiencyoverwegingen ons dwingen, hier wat optimaliserend te werk te gaan, een optimalisatie, waarvan ik niet weet, hoe ik het doen moet.

Zij bieden aan in eerste instantie een 7040 (de 7044 is drie keer zo ~~XXX~~ snel; hiervoor geven zij geen uitgewerkte offerte). Fixed point optelling duurt 16, resp. 5 nms, fixed point vermenigvuldiging 33-46, resp 22-35 nms. (hier is de factor veel kleiner, omdat de vermenigvuldiging niet memory limited is.) Floating point optelling 25 resp. 14 nms. Dit zijn inderdaad de snelheden, die ik me had voorgesteld (mijn voorstelling keek niet op een factor twee). Met de 7040 hoeven we niet bang te zijn, iets veel te langzaams te hebben, met de 7044 hoeven we niet bang te zijn, iets veel te snels in huis gehaald te hebben.

Nu de machine zelf, de CPU (Central Processing Unit); deze bestaat in drie versies (afgezien van de twee snelheden) "Basic instruction set", daar kan je verkrijgen "Extended instruction set" en daar kan je weer verkrijgen "Floating point operations". Alleen de volledige machine zou in aanmerking kunnen komen.

Zolang we niet kijken naar opdrachten, die met communicatie verband houden, maakt de opdrachtencode een redelijke indruk, met uitzondering van de manier, waarop de indexregisters gehanteerd worden. Ter verhoging van de feestvreugde wordt bij B-modificatie de inhoud van het indexregister niet bij het basisadres opgeteld, maar om volslagen duistere reden er van afgetrokken. Dit zou je kunnen "herstellen" door de inhoud van de B-registers (indexregisters) invers te interpreteren, ware het niet, dat je dan gedwongen wordt tot een voorstelling van negatieve getallen in 2-complement vorm. Verder representeert de machine zijn getallen met teken en modulus, dus de rotzooi is compleet. Er zijn drie indexregisters, en in de opdracht zitten drie bits, waarmee je indicering kunt aangeven. In het geval van multipale indicering wordt van de aangewezen bitrijen de "inclusive or" gevormd (een 1 waar ~~XXXXXXXXXXXXXXXXXXXXXXXXXXXX~~ minstens 1 van de registers een 1 heeft) en de aldus gevormde bitrij wordt van het adres afgetrokken. Waar je het voor gebruiken wilt, is mij een raadsel. De opdrachten, die de indexregisters vullen, gebruiken dezelfde bits om aan te geven, welk indexregister gevuld moet worden: zij kunnen dus zelf niet geïndiceerd worden, wat heel hinderlijk is. (En onvergeeflijk, als je bedenkt, dat de opdrachtlength 36 bits is). De floating point opdrachten zijn een mooi set, vooral omdat er opdrachten voor floating dubbele lengte bij ~~XX~~ zitten; helaas zijn er voor de binaire exponent inclusief teken slechts 8 bits ter beschikking, het bereik ligt dus tussen 10 tot de macht plus en min 40, en dat is te weinig, om er geen last van te hebben.

Om de interrupt te beschrijven, zijn 7 volle bladzijden van het reference manual nodig; ik heb hier een etmaal aan besteed en ik heb mij niet kunnen overtuigen, dat dit een gezond stel op elkaar afgestemde faciliteiten is. (Uit het manual blijkt, dat het nog ingewikkelder is, dan in de 7090; het gerucht, dat het interruptiesysteem van de 7090 zo gecompliceerd is, dat gebruikers-bezitters niet van de waterdichtheid ervan overtuigd zijn, heb ik in Muenchen kunnen verifiëren. Als iemand het me uit kan leggen, houd ik me aanbevolen.)

De sectie over de "output typewriter", pg 59, kan ik een ieder aanbevelen.

Electrologica. Aangeboden wordt een X8, dwz. een machine, die intern lijkt op de X1, maar 8 keer zo snel. (Inmiddels is mij ter ore gekomen, dat besloten is een sneller kerngeheugen te bestellen, waardoor de benaming "X13" wat meer in overeenstemming met de werkelijke snelheidsverhoudingen zou zijn.)

Ik weet niet, wat de invloed hiervan op de prijs is, ik houd mij daarom aan de machine, zoals hij ons schriftelijk is aangeboden. De prijs van de aangeboden configuratie is praktisch dezelfde als die van de aangeboden IBMconfiguratie (2 miljoen), nl. een 7040. De X8, zoals aangeboden, is in zijn non-multiplicatieve operaties bijna tweemaal zo snel als de 7040, in de multiplicatieve ongeveer twee maal zo langzaam. Ook hier geldt dus, dat de snelheid ligt binnen de marges, die we ons hadden voorgesteld. (De relatieve snelheden van multiplicatieve en niet multiplicatieve operaties ligt met het oog op moderner machinegebruik wat gunstiger, waar een groter percentage opdrachten ten bate van interne organisatie -non-multiplicatief als regel- uitgevoerd moesten worden.)

Persoonlijk betreur ik, dat bij het ontwerp zo'n groot gewicht is gaecht aan compatibiliteit met de X1; dit niettegenstaande het feit, dat de offerte ons dit als een voordeel doet voorkomen. ("Door deze overeenkomst zullen alle bestaende programma's van de X1 onmiddellijk beschikbaar zijn voor de X8" en "Zoals reeds genoemd is de gehele subroutine-bibliotheek van ~~XXXXX~~ het X1-systeem voor deze installatie beschikbaar, wellicht na enkele wijzigingen, die mede een gevolg kunnen zijn van de grotere faciliteiten waarover de X8 beschikt. O.a. is beschikbaar een volledig ALGOL 60 vertaalprogramma, dat zal worden aangepast aan de specifieke eisen, die door de X8 worden gesteld.")

De compatibiliteit betreft die met een X1, uitgerust met een interne drijvende arithmetiek. Oorspronkelijk was de X1 een drieregistermachine, de registers A en S, waarin tevens multiplicatieve en logische operaties konden plaatsvinden, en het B-register, dat speciaal fungeerde als adresmodificatieregister. Alle drie de registers zijn volledige accumulatoren. Ten bate van de arithmetiek ~~XXXXXXXXXX~~ met drijvende komma is er een apart register geschapen, dat een drijvend getal kan herbergen, de drijvende accumulator.

In twee opzichten is compatibiliteit met de X1 verlaten, wat ik beide als aanzienlijke verbeteringen beschouw. De synchronisatie met externe apparatuur is opnieuw opgezet. (In de X1 was dit langzamerhand een soort van rommeltje geworden; omdat het hier een isoleerbare hoek van de programma's betreft, waren wijzigingen hier nog te overzien.) Het effect van deze schone lei is

- 1) in de eerste plaats, dat geen ingreep de machine meer voor een "essentiele haastsituatie" zal plaatsen
- 2) de ingrepen niet meer geperst hoeven te worden in het keurslijf van 7 klassen.

De tweede verbetering betreft een nieuwe betekenis aan wat vroeger de "C-correctie" betekende. (Ook deze wijziging heeft geen ernstige repercussies, omdat de praktijk geleerd heeft, dat de C-correctie in de X1 nooit gebruikt wordt.) Hiermee is een dubbel doel bereikt: de aanmerking, dat de code een "onbruikbare" faciliteit bevat, geldt niet meer, ten tweede is hiermee ruimte in de code geschapen om opdrachten ook te kunnen corrigeren met het S-register. De praktijk van de X1-programmering heeft nl. geleerd, dat het hinderlijk is, dat in het functioneel gebonden B-register geen multiplicatieve en logische operaties mogelijk zijn; doordat de drijvende accumulator nu een gedeelte van de rol van het "arithmetisch tussenregister" gaat overnemen, komen de oorspronkelijke registers A en S meer beschikbaar voor "adresoperaties". Van de faciliteit van S-correctie stel ik me veel voor.

De zwakke compatibiliteitseis, dat alle opdrachten, die in de X1 ~~ook~~ mogelijk zijn ook in de X8 mogelijk zouden moeten zijn, werd onvoldoende geacht: hierbij werd

werd tevens de eis gesteld, dat de opdrachten, die in beide machines voorkomen bit voor bit dezelfde representatie zouden krijgen. Dit heeft tot gevolg gehad, dat de uitbreidingen van de opdrachtcode in de overgebleven gaten gepast moet worden. Dit heeft aanleiding gegeven tot enige beperkingen in de varianten, die bij de nieuwe opdrachten kunnen voorkomen plus een wat chaotische afbeelding vande nieuwe opdrachten op de functiebits. (Van dit laatste heeft de programmeur natuurlijk niet de minste last, maar het is niet zo leuk, als je het weet.) Het is daardoor een machine, waarin de historische groei, zij het op secundaire punten, duidelijk zijn sporen heeft achtergelaten. Het pleit, om in de opdrachtrepresentatie schoon schip te maken -wat niet los te denken is van het voorstel om de ~~woordlengte~~ woordlengte te vergroten- heb ik verloren. Ik mok daar niet meer over, want van de redelijkheid van de argumenten ben ik inmiddels wel overtuigd. (In dit verband is een van de opmerkingen, dat, nu twee woorden voor de representatie van een drijvend getal ter beschikking gesteld worden, we hier eigenlijk een heel mooi waardebereik hebben; dit is te verdedigen. Als je de woordlengte langer maakt, wordt het gaandeweg verleidelijk, om een drijvend getal in een enkel woord te representeren, maar voordat je daar aan toe bent, zonder dat het naar wordt moet je al tegen de 40 bits zitten, bij voorkeur nog wat verder.) Resultaat van het een en ander is, dat de X8 met 32768 woorden tot een random access geheugen van 884736 bits komt, 10 procent onder ~~de~~ de door ons genoemde ondergrens ("1 a 2 miljoen bits"). Als pleister(tje) op deze wond mogen we bedenken, dat programma's in machinecode -en ik denk nu speciaal aan de vaste programma's, wier aanwezigheid in het kerngeheugen door het "systeem" verondersteld wordt, zoals "masters" of "supervisors"- vergeleken bij andere machines vrij compact gerepresenteerd worden.

Voor de X8 pleit: (naast het reeds genoemde)

het gering aantal registers, dat een speciale rol speelt. (Hierop struikelen machines als de Univac 1107, om een sterk voorbeeld te noemen);
 een trommel (merk philips) met 512 sporen van elk 1024 woorden, die als continu geheugen geadresseerd mag worden (midden in de transferopdracht mag een spoorwisseling voorkomen.)
 de tot op zeker hoogte reële mogelijkheid om op het ontwerp invloed uit te oefenen.

Nevenvoordelen zijn, dat de rijksuniversiteit van Utrecht er een besteld heeft en dat het mathematisch centrum erg graag het prototype zou willen hebben. Een beperking van dit laatste voordeel is het feit, dat het mathematisch centrum vooreerst mikt op een installatie zonder trommel, en dat maakt een heel andere machine.

Tegen de X8 pleit: (naast het reeds genoemde)

dat de trommel geleverd moet worden door Philips, die met trommels voor de X1 nu al een jaar over de leveringstermijn heen is. (Of betekent het feit, dat de trommel desondanks in het ontwerp van de X8 is opgenomen, dat Electrologica goede moed heeft, dat de Philips-trommel in 1965 wel voor elkaar zal zijn?)

Dat Electrologica door zijn beperkte omvang een beperkt programma moet hebben, wat onder andere tot uiting zal komen, dat ze niet vreselijk toeschietelijk zullen zijn om fancy-communicatieapparatuur aan te sluiten: de kans, dat ze de ontwikkelingskosten daarvoor over een voldoende groot aantal klanten kunnen uitmeren is te klein.

Tenslotte: de machine is niet bij uitsluiting ontworpen met het oog op parallele programmering. (Het is misschien niet helemaal fair: dit gebrek deelt de X8 met bijna alle machines.)