



NR. **3**
VOLUM 2

ND NYTT



A/S NORSK DATA-ELEKTRONIKK
Erich Mogensens vei 38, Oslo 5. Tlf.: 21 73 71.

AV INNHOLDET

Nord-Display systemet

Siv.ing. Kåre Netland IFA, Halden

EDB undervisningen ved Sørlandet Tekniske Skole

Intervju med lektor Knut Brautaset

NOCUS — Nord Computer Users Society

Mini computeren inn i landmålingsberegningene

Beregner Bjørn Halseth

Et møte med Åsmunn Laukholm

Ny markedsføringsjef ved A/S Norsk Data-Elektronikk

Om paging

Torolf Paulsen, senior programmerer ND

ND NYTT

DESEMBER 1971 - Nr. 3, VOL. 2

Redaksjon:

Erich Mogensøns vei 38, Oslo 5. Tlf. 21 73 71

Redaktør:

Jan Aske Børresen

Redaksjonssekretær:

Inger Meidell-Haugland

Tidsskriftet distribueres gratis til alle interesserte. Vennligst send et brevkort til redaksjonen og kom med på distribusjonslisten.

Foto på omslagssidene Inger Sørli, IFA, Halden

Nord-Display systemet

SIV.ING. KÅRE NETLAND, IFA, HALDEN

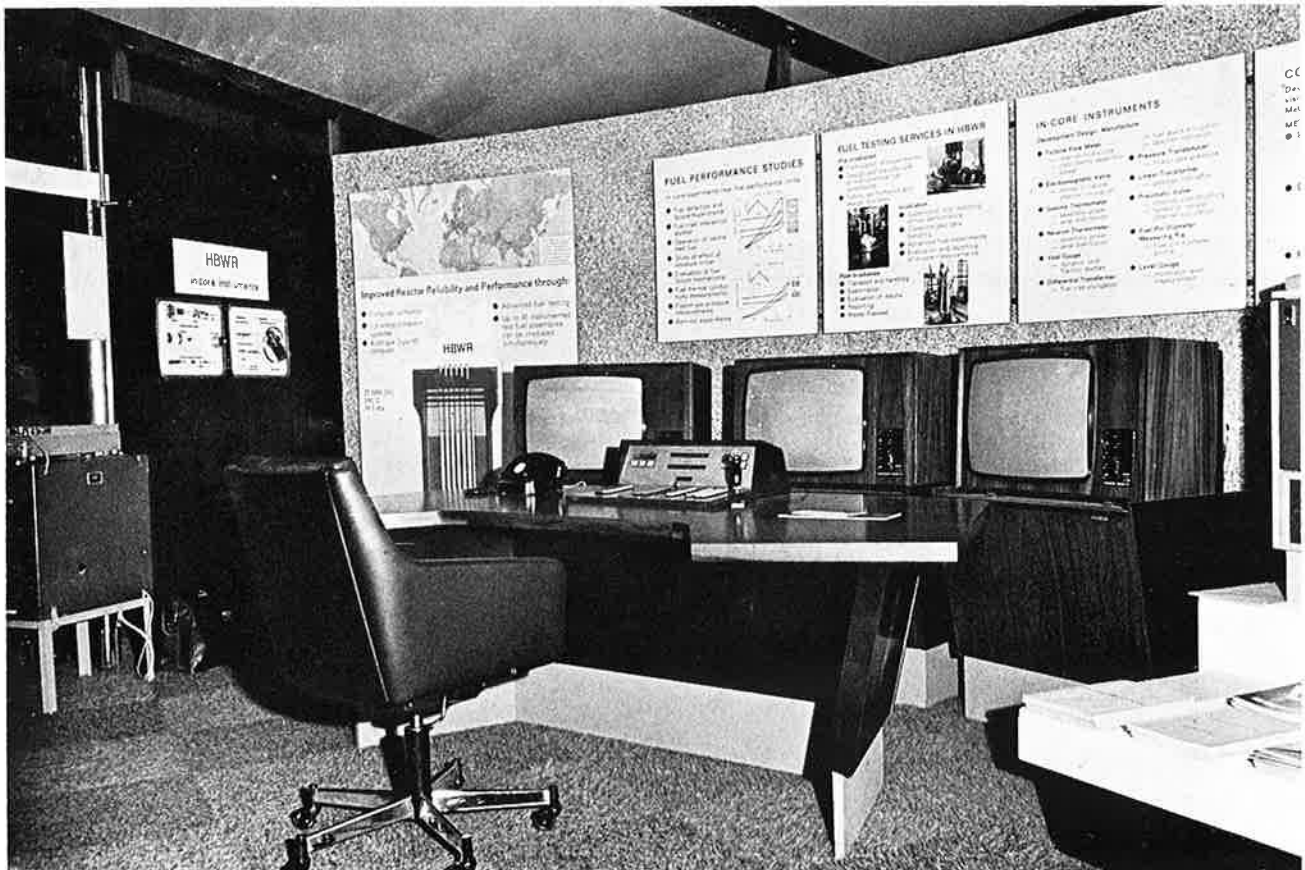
For ca. 1 år siden ble det inngått en samarbeidsavtale mellom Institutt for Atomenergi, Halden og A/S Norsk Data-Elektronikk om utvikling og utprøving av et operatør-kommunikasjonssystem (OPKOM). Et av kravene som ble satt opp var at systemet skulle kunne tilpasses de fleste prosessanvendelser. Andre hovedspesifikasjoner var at både alfanumeriske og grafiske bilder skulle kunne vises på fargeskjermer. Det ble innledningsvis foretatt en del markedsundersøkelser som viste at det var et udekket behov for et tilfredsstillende grafisk system for fargeskjermer. Det ble derfor besluttet å designe og bygge et prototyp-

system som skulle ha følgende hovedspesifikasjoner:

- vise alfanumeriske- og grafiske bilder
- benytte standard farge-TV
- baseres på bruk av NORD-20 for bildegenerering.

Et foreløpig system ble i høst presentert på en vitenskapelig utstilling i Geneve i forbindelse med en konferanse: «Peaceful Use of Atomic Energy», arrangert av FN. Utstyret som var en del av et integrert NORD-prosess-system vakte stor oppmerksomhet blant tilskuerne. Systemet besto av en NORD-1 med prosess-utstyr, NORD-2B, kontrollpult og 3 TV-skjermer, se

Fra utstillingen «Peaceful Use of Atomic Energy» i Geneve.



bildet. I systemet simulerte NORD-1 deler av Haldenreaktoren og fra operatørpulten kunne man få frem på skjermene prosessdata som f.eks. bilde av reaktorkretsene vist på første omslagsside. Her viser rødt tungtvannskretsene, grønt sekundærkretsene med varmevekslere og blått reaktorens utgangskretser.

BRUKERFUNKSJONER

Det endelige utstyret vil få en realistisk utprøving ved å inngå i et større system for overvåking og styring av Haldenreaktoren. Kommunikasjonssystemet vil bli delt mellom reaktoringeniøren og reaktoroperatøren. Ingeniøren vil ha en fargeskjerm og et vanlig tastatur til disposisjon, mens operatøren vil benytte tre skjermer og et konsoll.

All informasjon om prosessen som overvåkes ligger lagret i datamaskinen og operatøren kan kalle frem kretsdiagrammer som angir styrelogikken. Kretsdiagrammene finnes i forskjellige detaljnivåer og operatøren bruker en markør for å finne frem nye detaljer. Kommentarene i kretsdiagrammet er gitt i klar-tekst. Operatøren har også mulighet til å inspisere øyeblikksverdier av variable og sammenligne disse med grenseverdier. Han benytter i denne forbindelse konsollet som har et panel med knapper definert for denne spesielle prosess.

Det vil være et samspill mellom operatør og ingeniør, ved at førstnevnte tar de øyeblikkelige aksjoner, mens ingeniøren overvåker prosessens utvikling og endrer styreparametre, grenseverdier, etc. Forbindelsen mellom displaykontrollen og skjermen kan velges og en operatør som vil ha hjelp til tolking av f.eks. en trendkurve kan overføre dette bildet til ingeniørens skjerm.

BESKRIVELSE

NORD-20 med 4K hukommelse, vil være administratoren i systemet og kommunisere både med hoved-

maskinen (NORD-1), skjermene, tastatur og operatørkonsoll, se figuren. Eksterne videosignaler som TV-kamera, tape recorder, etc. kan programkontrolleres. Koordinatsettene overføres fra hovedmaskinen og programsystemet i NORD-20 vil generere et billedmønster som legges ut på en ekstern hukommelse som er tilknyttet den ønskede skjerm. (Den eksterne hukommelsen utvikles av Chr. Michelsens Inst., Bergen.)

Alfanumeriske data overføres som karakterstrenger til NORD-20 hukommelsen. Normalt genereres karakterene ved hjelp av «hardware», men også en «software» generator er tilgjengelig, dette for å kunne operere med valgbare karakterstørrelser.

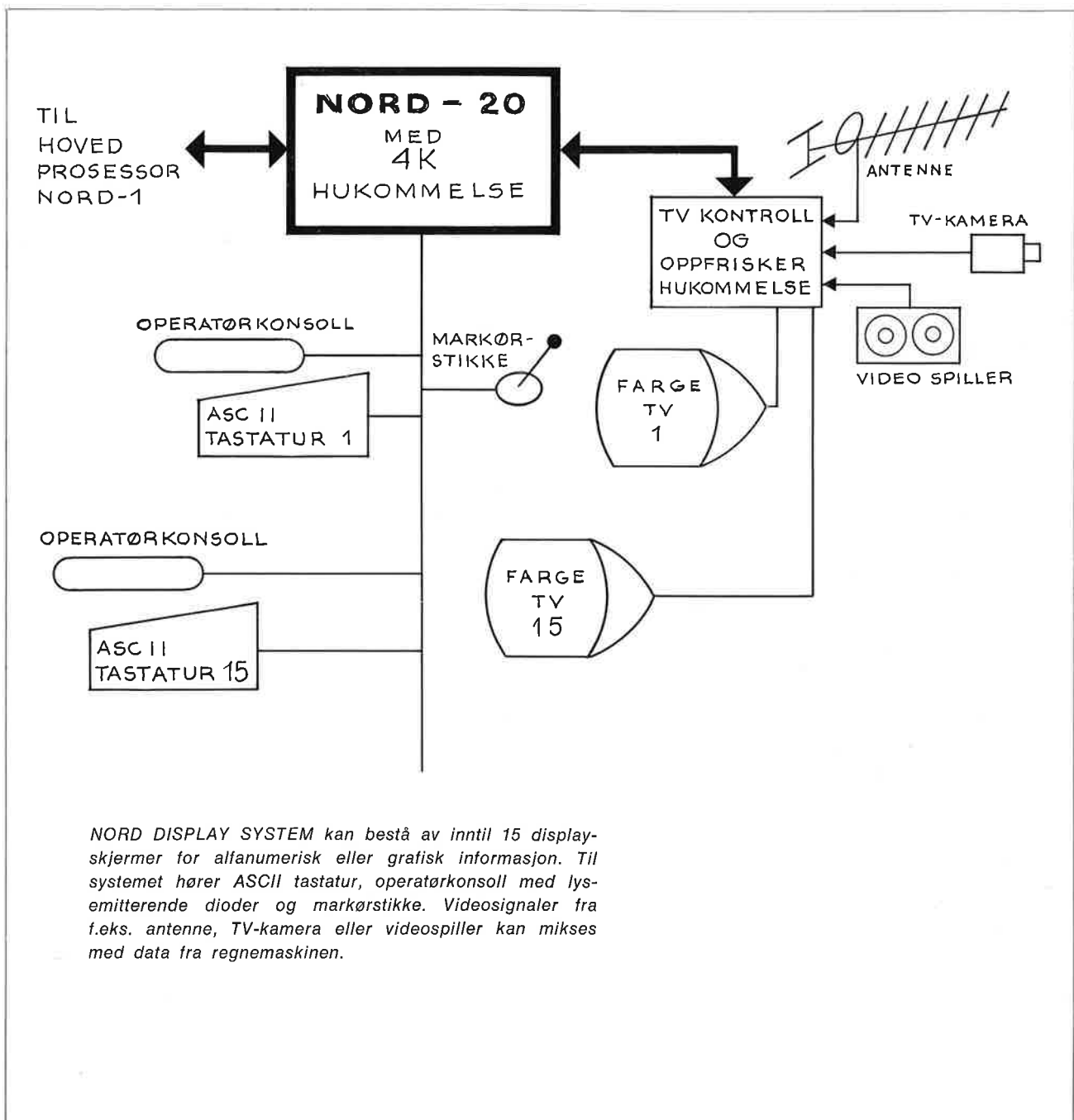
Oppfrisking av bildet foregår mellom den eksterne hukommelsen og skjermen og styres av de samme signaler som synkroniserer TV-apparatene. Skjermene som inngår i systemet er standard farge-TV-skjermer fra Tandberg Radiofabrikk som har bidratt med sine kunnskaper på video- og TV-teknikk området.

ANVENDELSER

NORD-DISPLAY SYSTEM tenkes benyttet i en lang rekke regnemaskinanvendelser og av områder kan nevnes:

- i forbindelse med prosesskontroll
- som terminal
- som interaktivt display
- i undervisning

I et senere nummer av ND-NYTT vil andre anvendelser bli behandlet.



NORD DISPLAY SYSTEM kan bestå av inntil 15 display-skjermer for alfanumerisk eller grafisk informasjon. Til systemet hører ASCII tastatur, operatørkonsoll med lys-emitterende dioder og markørstikke. Videosignaler fra f.eks. antenne, TV-kamera eller videospiller kan mikses med data fra regnemaskinen.

EDB-undervisningen ved Sørlandet Tekniske skole

INTERVJU MED LEKTOR KNUT BRAUTASET



På konferansen om norsk dataindustri som nylig ble holdt i Oslo var utdannelsen innen EDB i søkelyset igjen. Omtrent samtlige foredragsholdere fremhevet behovet for bedre og mer undervisning innen EDB. Det er derfor hyggelig å komme til en skole hvor EDB-undervisningen er blitt mere enn bare planer. Sørlandet Tekniske Skole i Grimstad fikk installert et NORD-1 anlegg i august i år og ND-Nytt har i den anledning intervjuet lektor Knut Brautaset som er ansvarlig for EDB-installasjonen ved skolen.

ND-Nytt: Knut Brautaset, ved siden av å være ansvarlig for EDB-anlegget på SØTS er du også sekretær i et utvalg oppnevnt av departementet for å se på deler av EDB-undervisningen ved de tekniske skoler. Kunne du kort fortelle litt om hva som er gjort frem til idag?

Brautaset: Når det gjelder EDB i de tekniske skoler, har inntil nå hver skole lagt opp undervisningen som de selv ønsket. For et års tid siden var det ferdig en innstilling fra et utvalg oppnevnt av departementet med Harloff som formann. Dette utvalget hadde som oppgave å planlegge opplæring i EDB ved de tekniske skoler. 2-årige EDB-linjer er nå i gang ved Bergen og Trondheim Tekniske Skoler. Innstillingen tok også for seg muligheten av å innføre EDB som et obligatorisk grunnfag. Det ble senere nedsatt et arbeidsutvalg som skulle bistå departementet med innføringen av et slikt obligatorisk fag. Det er enda ikke gitt direkte beskjed om at et slikt fag skal innføres, men det er stor sannsynlighet for at det blir i nær fremtid. Den arbeidsgruppe jeg er sekretær for arbeider med dette obligatoriske faget som jeg tror vil komme til å hete «Grunnleggende databehandling».

ND-Nytt: Kan du gi oss et lite inntrykk av hva dette grunnleggende EDB-faget vil inneholde?

Brautaset: Det er tenkt som en innføring i EDB og vil behandle hvordan en datamaskin er bygget opp, periferenheter, administrativ EDB, teknisk EDB, pro-

sesskontroll og et innføringskurs i et høynivåspråk. ND-Nytt: Det kunne nå være interessant å høre om anlegget på Sørlandet Tekniske Skole.

Brautaset: I august 1971 fikk vi en NORD-1 med 16K hurtighukommelse og fullt utbygget sentralenhet med unntakelse av power fail detect. Til maskinen er knyttet 5 teletype terminaler, kortleser, papirbåndleser og punch, samt en interface som binder sammen sentralenheten med en verktøymaskin. Det er en fres som har stepmotor styrt i to akser og har mulighet for diskrete step i vertikalaksen.

ND-Nytt: Hvilken programutrustning benytter dere?

Brautaset: Det vi benytter nå er et BASIC Time-sharing system. Dvs. at våre 5 terminaler står i time-sharing med sentralenheten og benytter høynivåspråket BASIC. Utenom dette har vi FORTRAN oversetter og selvfølgelig da Assembly oversetter som MAC og en del andre småprogrammer.

ND-Nytt: Hvordan er undervisningen organisert og hvilke erfaringer har dere gjort?

Brautaset: Vi har i dette semesteret (høstsemesteret 1971) satt i gang et obligatorisk kurs 3 timer pr. uke + kveldskjøring på terminalen. Kurset starter med en gjennomgåelse av en audio-visuell studiepakke som gir en kort innføring i virkemåten og oppbyggingen av datamaskinen. Så starter vi med et programmeringskurs i BASIC som omfatter forelesninger og øvinger. Øvingene organiseres slik at elevene sitter to og to ved hver av våre teletyper. De kan da veksle om å betjene teletypen samtidig som de følger med i hverandres programmer, og på den måten utnyttes anlegget på best mulig måte. For å få tilstrekkelig øvningstid på terminalen kan elevene kjøre frivillig om kvelden. De elevene som har kommet lengst, vesentlig 3.-klasser, har et lite team som har vakt 4–5 timer hver kveld. Elevene må på forhånd tegne seg på en liste over hvilke terminaler de ønsker å bruke, og hvor lenge. Det er faktisk så stor interesse blant elevene at det er vanskelig å få slått anlegget av.

*Sørlandet Tekniske Skole, Grimstad.
To av elevene opptatt ved en av de 5 teletypeterminaler.*



ND-Nytt: Når det snakkes om å innføre EDB i de tekniske skoler, er det vel ingen hemmelighet at det har hevet seg kritiske røster, særlig fra lektorer i de tekniske fag. Man er redd at EDB skal stjele så mange timer at det skal gå utover undervisningen i de andre fag. Hva har du å si til dette?

Brautaset: Det viser seg at elevene ved innføring av et fag som EDB går gjennom en «lekefase» som kan ta veldig mye tid. Det er viktig derfor å kjøre med en viss disiplin. Datamaskinen er et meget effektivt regneverktøy som gjør at man kan benytte modeller som tidligere måtte forenkles vesentlig for at regnearbeidet ikke skulle ta overhånd.

ND-Nytt: Vil datamaskinen få innflytelse på undervisningen i fag som fysikk og matematikk?

Brautaset: I fysikkundervisningen vil datamaskinen både kunne benyttes som regneverktøy og som demonstrasjonsanlegg. Simuleringsprogram kombinert med display har vi stor tro på. Dette vil gi muligheten for mange realistiske demonstrasjoner. Når det gjelder faget matematikk, tror jeg datamaskinen vil gjøre det nødvendig å endre noe av pensumet. Numerisk analyse må komme sterkere inn.

ND-Nytt: Retter vi blikket fremover, har du noen formening om hvordan EDB's innflytelse vil bli innenfor de tekniske fag?

Brautaset: Utnyttelse av EDB i de tekniske fag er sterkt avhengig av den enkelte faglærer. Man kan tenke seg at en del laboratorieoppgaver med mye regnearbeid kan bli utført med datamaskinen. Dette gjelder samtlige linjer. På bygningslinjen er landmåling et meget aktuelt område. For elektrolinjen vil datamaskin-assistert konstruksjon få sterk aktualitet. Innen reguleringsteknikk vil maskinen også spille en viktig rolle. Når det gjelder maskinlinjen kan EDB benyttes som støtte til alt som har med varmekraft-

arbeid og konstruksjonsarbeider å gjøre. Dessuten krever måleteknikk en god del statistisk beregning som maskinen passer godt til. For å realisere disse mulighetene må vi ha kurser for personalet slik at flest mulig får tak på dette med programmering og samtidig viten om hvordan de kan utnytte regnemaskinen innen sitt eget fagområde.

ND-Nytt: Vi vet jo at skolens rektor, Einar Grude, er meget interessert i verktøymaskinindustrien. NORD-1 anlegget er tilknyttet en fresemaskin, hvilke planer har dere videre i den retningen?

Brautaset: Det rektor Grude er opptatt av er automatisering av verktøymaskinene for mindre bedrifter, bedrifter som er så små at de ikke har anledning til å anskaffe det spesialutstyr som etter hvert er tilgjengelig. Man kunne kanskje tenke seg en datamaskin som kunne styre flere maskiner, og at maskinen også kunne ta seg av andre oppgaver for bedriften. Vi vil først satse på å dyktiggjøre oss selv. Etter hvert får vi kanskje erfaringer som er verd å meddele andre i form av kurser og seminarer.

ND-Nytt: I hvilken retning tenker dere å bygge ut anlegget?

Brautaset: For å utnytte anlegget optimalt ønsker vi en eller annen form for masselager med kort aksess-tid. Videre vil vi gjerne ha hullkortutstyr, og får vi en linjeskriver, vil anlegget også kunne kjøres i batch mode. Dette da særlig med tanke på FORTRAN kurs.

ND-Nytt: Dette at dere har et regneanlegg, hvilken betydning har det for skolen?

Brautaset: Skolens lærerstab er meget åpen for de muligheter som datamaskinen gir. I den stadige oppdatering som må skje innen de tekniske fag vil adgangen til et slagkraftig regneverktøy være en ekstra stimulans.

SIV.ING. KÅRE NETLAND, FORMANN I NOCUS



BRUKERMØTE

Høstens brukermøte ble avholdt på Hurdalsjøens Hotell og Kurscenter i tidsrommet 31. oktober til 2. november 1971. Hele 78 utsendinger og interesserte var møtt frem søndag kveld da Norsk Data-Elektronikk tradisjonelt inviterte til middag. Kvelden ble meget vellykket med god mat og hyggelig sammenkomst utover i de små timer. Særlig populært var forbindelsen til ND's datasenter og månelandingsprogrammet som kunne betjenes via teletype.

Møtevirksomheten ble åpnet mandag av formannen som ønsket alle velkommen og orienterte om styrets arbeid det siste året. I perioden er det avholdt 9 styremøter, og det har vært kontinuerlig kontakt med ND angående feilrapporteringssystemet, kvalitetskontrollen, dokumentasjonssystemet, tidsfristene for levering av software og manualer, etc.

PRESENTASJONER

Et vidt spekter av emner ble presentert og de kan sammenfattes i følgende hovedgrupper:

- Programmering i høynivåspråk.
- Kommunikasjonsproblematikken i forbindelse med prosesskontroll.
- Evaluering av datamaskiner.
- NORD-maskiner i arbeid.

Det var tydelig at foredragsholderne hadde arbeidet med sine innlegg. De fleste hadde også levert en skriftlig oppsummering av sitt foredrag. Det er ønskelig at denne kvalitetsforbedring av innleggene skal fortsette.

RESOLUSJONER

NOCUS-medlemmene vedtok enstemmig 3 resolusjoner og 1 henstilling til ND. Kort summert var innholdet:

- Bedre og regelmessig ND-BULLETIN.
- Klargjøring av ND's garantitilbud.
- Bedre hardware-testprogrammer.
- Henstilling om videreutbygging av QA-avdelingen.

Korrekt tekst vil komme i møtereferatet.

VALG

Valgkomiteen la frem forslag til formann og 4 styremedlemmer, mens det fra salen kom forslag om 2 andre styremedlemmer. Etter skriftlig avstemming ble følgende styre valgt for en ny 1-års periode:

Formann:

Kåre Netland, Institutt for Atomenergi, Halden.

Styremedlemmer:

Dag Haveråen, Computas A.S., Oslo

Øystein Helle, Universitetet i Bergen

Audun Haugs, Chr. Michelsens Institutt, Bergen

Sverre Morterud, Meteorologisk Institutt, Oslo.

De 3 førstnevnte var med i det foregående styret. Etter vedtektene skal hver av styremedlemmene peke ut sin varamann, mens styret skal velge varaformann, sekretær/kasserer. Dette vil bli gjort på første styremøte.

REFERAT

Referat fra brukermøtet vil bli sendt deltakerne så snart som mulig etter at det har vært behandlet på styremøtet 22. november.

NOCUS adresse:

NOCUS

Kåre Netland

Institutt for Atomenergi

Postboks 173

1751 Halden.

Mini computeren inn i landmålingsberegningene

BEREGNER BJØRN HALSETH,
OSLO OPPMÅLINGSVESEN



For lenge, lenge siden satt en huleboer foran hulen sin. Han var i ferd med å merke av plassen der det nye ildstedet skulle plasseres. Med pinnen han holdt i hånden laget han et kryss i sanden og sa: «På dette punktet skal ildstedet ligge».

Uten å vite det, brukte denne mannen en av de viktigste definisjoner innen geometrien, «et punkt defineres som det sted der to linjer skjærer hverandre». Geometrien, eller nærmere bestemt plangeometrien, er grunnstenen i alle beregninger innen landmålingen, men frem til i dag er disse beregninger stort sett blitt utført for hånd.

OSLO OPPMÅLINGSVESEN

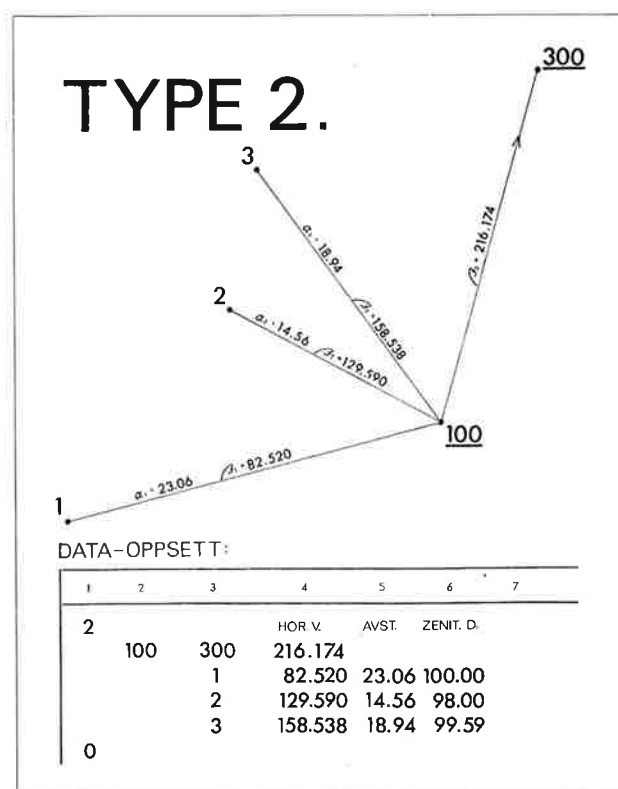
Oslo Oppmålingsvesen har tilsammen ca. 150 ansatte, av disse er ca. 30 ingeniører og 20 oppmålingsfullmektiger som er fordelt på 4 avdelinger, stikningsavdelingen, målebrevsavdelingen, ekspropriasjonsavdelingen og kartverksavdelingen. I tillegg kommer så kontoravdelingen. Arbeidet går i korthet ut på å holde orden på Oslo's titusener av eiendommer og omtrent like mange veier.

Målingene utføres med utgangspunkt i kjente polygonpunkter, og oppgavene består i fastleggelse av detaljpunkter i terrenget. Det er små avstander det dreier seg om og kravet til nøyaktighet er derfor stort både i oppmåling og ved beregning.

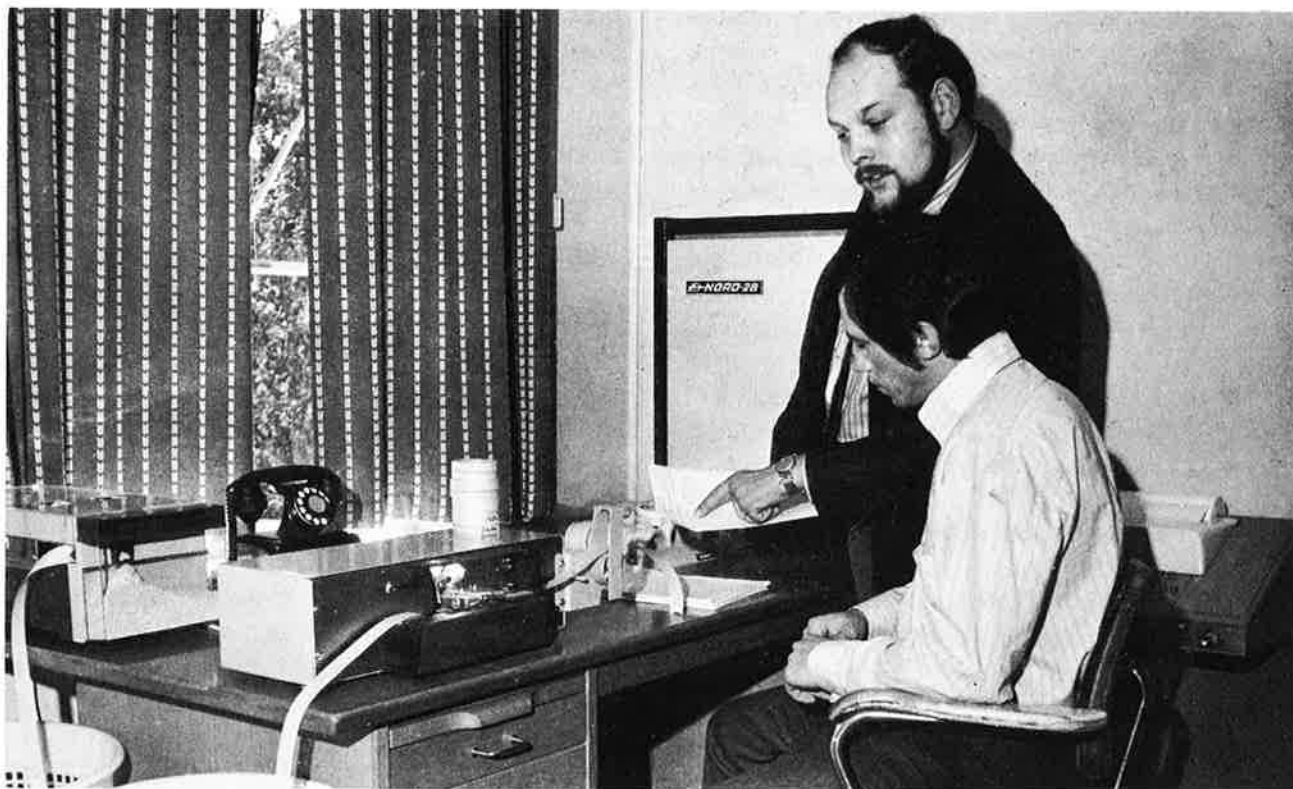
Oslo Oppmålingsvesen kom i kontakt med EDB så tidlig som først på 60-tallet, og allerede i 1964 hadde man laget egne programmer som ble kjørt på regnemaskinen ved Norges Geografiske Oppmåling. I 1969 begynte man så å tenke på et eget anlegg, og våren 1970 ble en NORD-2B bestilt. Maskinen har 8K med standard papirbåndleser, punch og teletype. Et lite datasenter er bygget opp, som betjener hele Oppmålingsvesenet. Til å lede datasenteret har man en beregnersjef, dessuten arbeider 2 damer med punching av data og en operatør betjener maskinen. På bildet er vist NORD-2B med tilhørende puncheutstyr.

Koordinatberegningsprogrammet M3 Programmet som i dag brukes er et koordinat beregningsprogram som dessuten beregner arealer og utstikningsdata. Man har lagt opp programmet slik at maskinen tester på relevansen av de data som leses inn og gir så feilutskriftter på teletypen. Maskinoperatøren har derved anledning til å rette eventuelle feil i dataene. Dataoppsettet er «String»-orientert, dvs. at man har ingen tabulering av de enkelte inngangsdata. Dette forenkler vesentlig arbeidet med punching av dataoppsettene.

En beregningsoppgave starter med innlesning av kjente punkter. Koordinatene til de kjente punktene ligger nå lagret i maskinen, og så følger beregningen av nye punkter på grunnlag av disse. Koordi-



Beregner Bjørn Halseth og operatør Erik Gulliksen diskuterer input til NORD-2 B.



natberegningdelen av programmet er oppdelt i 8 forskjellige beregningstyper, hvor inngangsparametrene for disse er kodet fra 2 til og med 9. Det kan eksempelvis nevnes at type 4 og 5 beregner skjæringspunktet mellom to rette linjer, type 6 beregner det ene av skjæringspunktene mellom en rett linje og en sirkel, type 7 beregner det ene av skjæringspunktene mellom to sirkler, og type 2, 3 og 8 beregner koordinatverdiene på grunnlag av målte størrelser, avstander og vinkler. Type 9 er en sammensatt beregning som ialt beregner 5 detaljpunkter. Denne typen har to rette linjer og en sirkel som inngår i oppsettet. På figuren er vist en polær innmåling (type 2). I inngangsdataene til denne typen oppgis først typenummeret, så oppstillingspunkt, tilsiktnings-

punkt og disse punktenes horisontalvinkel. Deretter følger nytt punktnummer med horisontalvinkel, avstand og zenitdistanse. Avstanden er gitt i meter og målt med målebånd. Punkt 100 og punkt 300 er kjente punkter, hvorav 100 er oppstillingspunktet. Programmet foretar på grunnlag av disse data koordinatberegninger av punktene 1, 2 og 3.

Av andre programmer som er under utarbeidelse kan nevnes et tegneprogram for automatisk optegning av kart, og et utjevningsprogram for et nett av polygonpunkter.

Datasenteret er i dag i full virksomhet og utfører ca. 30 beregningsoppgaver om dagen. Oslo Oppmålingsvesen stiller gjerne sin erfaring på dette område til disposisjon for andre.

Et møte med Åsmunn Laukholm

NY MARKEDSFØRINGSJEF VED
A/S NORSK DATA-ELEKTRONIKK



På NOCUS brukermøte i Hurdal 1. og 2. november ble ND's nye markedsføringsjef, Åsmunn Laukholm introdusert av direktør Lars Monrad-Krohn.

Etter realfagstudiet på Blindern arbeidet Laukholm 3 år hos L. M. Ericsson. Fra høsten 1970 har han vært ansatt i A/S Norsk Data-Elektronikk.

ND-Nytt benytter anledningen til å stille den nye markedsføringsjefen noen spørsmål om ND's produkter og fremtidsplaner.

ND-Nytt: Hvilket av ND's produkter selges mest for tiden?

Å. Laukholm: Det er NORD-2B som nå er omdøpt til NORD-20. Grunnen til dette er at NORD-2B er en prototypkopi hvor CPU består av 10 kort mot 6 kort i produksjonsmodellen. Funksjonsmessig er de to maskinene identiske.

ND-Nytt: Kan du nevne noen av hovedtrekkene for NORD-20?

Å. Laukholm: NORD-20 er en minimaskin med en ordlengde på 16 bit og 2 sett registre. Registerinnholdet bevares derfor automatisk ved interrupt. Interrupt-tiden blir således aldri lenger enn den langsomste instruksjonen. Instruksjonsrepertoaret er delt i to grupper. De fleste instruksjonene utføres i hardware, mens de øvrige fører til interrupt. Dette medfører blant annet at instruksjoner i NORD-1 som ikke er implementert i NORD-20 enkelt kan simuleres i software. Det finnes videre to operasjonsmodi i CPU. I den ene modus vil et ikke autorisert brukerprogram ikke kunne utføre bestemte instruksjoner, f.eks. input/output instruksjoner. Dette er aktuelt i multiprogramerte systemer hvor man kan teste programmer samtidig som maskinen styrer en prosess. NORD-20 egner seg, software- såvel som hardwaremessig, til bruk i et flermaskinsystem.

ND-Nytt: Hvilken periferutrustning leveres det til NORD-20?

Å. Laukholm: Full periferutrustning inklusive kassettbåndspiller og en liten disk.

I/O-bus er lik fra posisjon til posisjon. Det er I/O kortets plassering som definerer periferenhetens nummer.

ND-Nytt: Vil periferutrustningen som utvikles og leveres fra NORD-20 også passe for NORD-2B som allerede er levert?

Å. Laukholm: NORD-2B er levert med et av to I/O systemer. Det ene er identisk med det i NORD-1, mens det andre er identisk med det i NORD-20. De brukere som har den første varianten vil kunne benytte alt periferutstyret da både kassettbåndspiller og liten disk interfaces for NORD-1.

ND-Nytt: Er det andre ting ved NORD-20 du vil fremheve?

Å. Laukholm: Det er en billig maskin. Med 4K hukommelse, teletype og hullbåndleser koster den i overkant av 100 000 kroner. Det finnes minimaskiner hvor samme konfigurasjon koster mindre, men man bør da sammenligne instruksjonssettet og design på maskinene, dessuten spørre hvor meget det vil koste å utvide anlegget.

ND-Nytt: Hva med fremtidige produkter?

Å. Laukholm: Vi stiller store forventninger til vår nye datamaskin NORD-10 som er en videreutvikling av NORD-1. Disse er programkompatible slik at den software som er og vil bli utviklet for NORD-1 direkte skal kunne anvendes på NORD-10.

Ved konstruksjonen av NORD-10 blir det lagt vekt på driftssikkerhet og «service-vennlighet», og bare velprøvet teknologi benyttes.

ND-Nytt: Hva er det som skiller den nye maskinen fra NORD-1?

Å. Laukholm: NORD-10 vil bli utstyrt med en såkalt mikroprosessor som normalt vil ta seg av operatorkommunikasjon. Ved feil på anlegget vil mikroprosessoren kunne brukes til feilsøking. Feil i sentralregneenhet, hukommelse, datakanel etc. vil da lettere kunne finnes enn ved bruk av konvensjonelle metoder. Dette er en teknikk som har vist seg meget effektiv under utviklingen av NORD-5.

Hvert interruptnivå vil ha sin egen registerblokk slik at bytte av nivå vil skje innen en memory-cycle.

NORD-10 vil videre bli utstyrt med paging som gir et differensiert memory protect system og mulighet for dynamisk relokering og virtuell hukommelse.

Den nederste del av hukommelsen (1K) vil være meget hurtig (cycle-tid 100 ns.). Her vil registerblokkene og pageindekstabellene ligge. Små programmer hvor eksekveringstiden er kritisk kan legges her, dog ikke i den del som opptas av registerblokker og pageindekstabeller.

NORD-10 vil for de fleste instruksjoners vedkommende effektivt kunne utnytte en hukommelseshastighet på ca. 400 ns. Floating point multiplikasjon og divisjon til ta ca. 10 μ s.

ND-Nytt: Når vil den første leveranse finne sted?

Å. Laukholm: Den vil finne sted i begynnelsen av 1973.

ND-Nytt: Du nevnte NORD-5. Dette er vel den mest interessante maskin ND har konstruert til nå?

Å. Laukholm: Ja NORD-5 er en meget hurtig regnemodul konstruert spesielt til å operere sammen med minst en NORD-1 for vitenskapelige og matematiske arbeidsoppgaver.

Sentralenheten (CPU) i NORD-5 har for eksempel en hukommelsescycle på 0.5 mikrosekund, 64 hardware registre, og kan utføre en 64-bit floating point multiplikasjon på ett mikrosekund.

Den første av disse NORD-5 datamaskinene vil bli levert sammen med 3 NORD-1 i et system til Meteorologisk Institutt hvor den vil løse oppgaver innen numerisk værvarsling.

ND-Nytt: Vil ND lansere andre produkter i nær fremtid?

Å. Laukholm: Foruten NORD DISPLAY SYSTEM kommer vi med en kasettbåndspiller og en ny disk.

Kasettene er av samme type som vanlige musikk-kassetter, men har bedre spesifikasjoner. Den kan lese og skrive begge veier med en hastighet på ca. 300 kar./sek. Data er organisert på samme måte som på vanlige magnetbånd, men med sync-pulser i stedet for record gap. Man kan således søke gjennom båndet med høy hastighet, stoppe og komme i fase ved hjelp av sync-pulsene. En datablokk kan skrives om uten at innholdet i naboblokkene ødelegges. Kasettbåndspilleren fungerer derfor som et masselager.

Den nye disken har en fast og en utbyttbar skive. Lagringskapasiteten er 2,5 mill. karakterer pr. skive, og kontrolleheten kan ta inntil 4 systemer.

ND-Nytt: Hvilke oppgaver mener du firmaet bør legge vekt på?

Å. Laukholm: Vi vil fortsatt satse på å dyktiggjøre oss innen de felter vi tradisjonelt står sterkt: forskning, undervisning og prosesskontroll. Som en følge av firmaets ekspansjon vil vi i fremtiden dessuten måtte satse på eksport. Den første leveranse til Sverige vil finne sted i februar 1972. Det gjelder en NORD-1 med tromme som skal tilkoples antennen på kommunikasjonsstasjonen for satellitter i Tanum og styre denne, samt logge funksjoner i systemet.

Vi ønsker den nye markedsføringssjefen lykke til i sin nye stilling.

Om Paging

TOROLF PAULSEN, SENIOR PROGRAMMERER ND



Hurtighukommelse er kostbar, og de fleste regnemaskiner er derfor utstyrt med hukommelse i flere nivåer. Som hurtighukommelse brukes ferittkjerne (core), og masselager kan være tromme, disk, magnetbånd, dataceller, etc.

Et programs instruksjoner og data kan bare refereres i hurtighukommelsen. Det betyr at når et program skal utføres, må det hentes inn fra masselageret til hurtighukommelsen. Hvis programmet er for stort for hurtighukommelsen, får man problemet: Hvilke deler av programmet skal til enhver tid ligge i hurtighukommelsen, og hvilke deler skal være på masselageret? En måte å løse dette problemet på er å bruke paging.

I det følgende vil vi beskrive noen fordeler og ulemper ved paging, deretter se på hvordan paging på NORD-1 virker og til slutt beskrive noen begreper i paging.

Fordelene kan kort listes slik:

1. DYNAMISK RELOKERING

Et program kan flyttes rundt i hurtighukommelsen. Det behøver ikke stå i faste plasser mens det utføres. Det kan deles opp i blokker som kan stå huller til bulter. Det kan utføres mens det bare er delvis loadet, og den tilgjengelige del av hurtighukommelsen kan variere i størrelse etter som det passer. Alt dette skjer automatisk, uten at den som har skrevet programmet behøver å bekymre seg om det.

2. MEMORY PROTECT

Memory protect er ikke lenger knyttet til faste områder av hurtighukommelsen, men følger i stedet programmets deler. Selv om programmet flyttes rundt, følger memory protect automatisk med. Man kan også beskytte mindre områder enn ved vanlig memory protect, og man kan ha flere forskjellige slags memory protect funksjoner.

3. VIRTUELL HUKOMMELSE

Dette var den opprinnelige ideen da paging oppsto. Virtuell hukommelse brukes til å gi programmereren

illusjonen av at regnemaskinen har større hukommelse enn den egentlig har. For å få til dette, må programmereren få bruke et sett adresser som er forskjellig fra det adressesett som hukommelsen bruker. Det må også lages en mekanisme som oversetter programgenererte adresser til de korrekte hukommelsesadresser. En adresse brukt av programmereren kalles en virtuell adresse, og settet av alle virtuelle adresser kalles virtuell hukommelse. En adresse brukt av hukommelsen kalles en fysisk adresse, og settet av alle fysiske adresser er den fysiske hukommelsen. Et program skrives alltid for den virtuelle hukommelsen. Programmereren behøver ikke bry seg med hvordan programmet skal deles opp når det utføres. Det ordnes automatisk av regnemaskinen.

Dersom flere programmer kjøres samtidig, kan de isoleres helt fra hverandre, uavhengig av memory protect. Et program kan således aldri ødelegge et annet.

ULEMPENE VED PAGING ER FØLGENDE:

1. Utførelsestiden for et program kan gå opp til nesten det dobbelte. Dette er uakseptabelt for mange typer programmer, f.eks. sanntids programmer som må utføres raskt, eller beregningsprogrammer med mye regning og lite input/output.
2. Programmere som bruker virtuell hukommelse, lager ofte unødvendig store programmer. Dessuten blir programmene ofte skjodesløst organisert. Disse tingene kan begge være årsak til unødvendig mye overhead i regnemaskinens paging-mekanismer, slik at programmene blir lite effektive.
3. I en regnemaskin med pagingsystem kan hurtighukommelsen ofte bli utnyttet dårlig p.g.a. at den deles i blokker (pager). En del av den siste blokken vil da alltid stå ubrukt.
4. Når programmer utføres, sendes blokker av dem stadig mellom hurtighukommelse og masselager. Prinsippene som brukes for å velge ut hvilke blok-

ker som skal transporteres, og når de skal transporteres, er meget viktige. De er ofte avgjørende for om man skal få et godt eller et dårlig paging system.

PAGING PÅ NORD-1

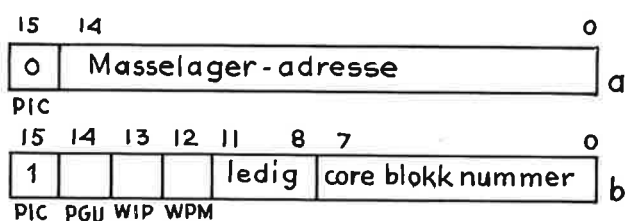
De fleste maskinleverandører har paging på noen av sine maskiner.

I det følgende vil paging for NORD-1 bli beskrevet, slik som det finnes på de maskiner som allerede har paging.

Paging er en oppdeling av den virtuelle hukommelsen i blokker av fast størrelse. En slik blokk kalles for en page. En page er på 256 ord. Den fysiske hukommelsen (core) deles i blokker av samme størrelse. Programmets instruksjoner og data, som vanligvis befinner seg på masselageret, hentes under utførelsen inn til den fysiske hukommelsen. En page fra masselageret havner i en ledig blokk i core. En oversetter (hardware) oversetter de virtuelle adressene til fysiske adresser. Dersom en referert page ikke ligger i en blokk i core, får vi ingen oversettelse. I stedet får vi et interrupt, missing page interrupt, og pagen må hentes før programmet kan fortsette. Vi kan også få interrupt dersom en ulovlig instruksjon (IOF, TRR, MCL etc.)* forsøkes utført, eller om et program forsøker å skrive i en beskyttet page.

Oversettelsen foregår ved hjelp av en pagetabell. Den er på 256 ord og ligger i en blokk i den fysiske hukommelsen. Hver page i den virtuelle hukommelsen har et ord i pagetabellen. Den virtuelle hukommelsen består da av 256 pager à 256 ord, dvs. 65 536 ord. Dette tilsvarer en 16 bits adresse. Hvert ord i pagetabellen inneholder en bit som sier om pagen er i core eller ikke. Dersom pagen bare er på masse-

lageret, inneholder resten av ordet i pagetabellen pagens masselageradresse. Dersom pagen er i core, inneholder ordet i pagetabellen et blokknummer samt page-used bit, protect-bit og modified-bit.



FIGUR 1 Ordet i pagetabellen a) dersom pagen er på masselageret b) dersom den er i core

Bit 15 PIC (Page In Core) er 1 når pagen er i core, 0 ellers.

Bit 14 PGU (PaGe Used) er 0 så lenge pagen ikke er referert, 1 ellers.

Bit 13 WIP (Written In Page) er 0 så lenge det ikke er skrevet noe i pagen, 1 ellers.

Bit 12 WPM (Written PerMitted) er 1 dersom det er lov å skrive i pagen, 0 ellers.

Bit 11, 10, 9 og 8 er ledige. Her kunne man tenke seg andre protect-bits, f.eks. om det var lovlig å lese data fra pagen eller ikke, og om det var lovlig å hente instruksjoner eller ikke.

Bit 7-0 er et core blokk nummer (bn).

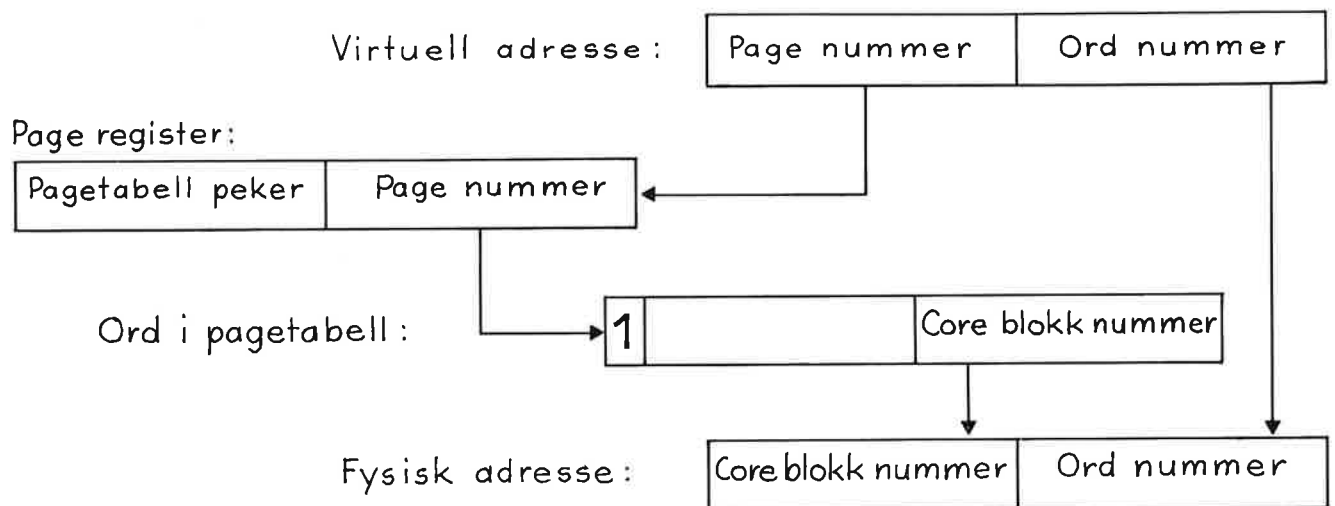
Adresse-oversettelsen foregår slik:

En virtuell adresse tenkes delt i et pagenummer og et ordnummer innen pagen (pn, on). Et pageregister peker på begynnelsen av pagetabellen. Vi føyer pn til pageregisteret og får en adresse som peker inn i pagetabellen. Derfra henter vi det blokknummeret (bn) som står der. Den fysiske adressen blir da (bn, on). Se figur 2.

De fire bitene i pagetabellen brukes slik:

PIC forteller om pagen er i core eller ikke. Den settes av software og brukes av hardwareoversetteren.

* IOF = Interrupt off
TRR = Transfer to register
MCL = Masked Clear



Figur 2

PGU settes av software når et program startes første gang. Hver gang en page refereres, settes PGU til 1 av hardware. Software kan sette PGU til 0 med jevne mellomrom og inspisere hvilke PGU'er som blir satt til 1, dvs. hvilke pager som blir brukt av programmet. På denne måten kan software danne seg et bilde av hvilke pager som blir brukt, og hvor ofte. Dette kan påvirke håndteringen av pagene. Mye brukte pager kan for eksempel holdes permanent i core. Dette vil spare transport av pager til og fra masselageret.

WIP settes ved starten til 0 av software. Når det skrives i en page, settes WIP til 1. Dette kan brukes når pager skal fjernes fra core. Dersom WIP er 0, er pagen uendret, og den behøver ikke transporteres tilbake til masselageret.

WPM er 0 dersom det ikke er lov å skrive i pagen. Dette betyr at man har et effektivt memory protect system. En hvilken som helst kombinasjon av pager kan beskyttes. Denne beskyttelsen henger sammen med pagen og ikke med core-blokken, så den er uavhengig av hvor i core pagen ligger.

Et program som utføres under paging, kan ikke utføre privilegerte instruksjoner. Da blir det avbrutt. Privilegerte instruksjoner er IOT, TRR, MCL, MST, ION, IOF, INTEN og INTDS.*

Et program som utføres under paging, kan heller ikke komme utenfor det område i core som programmet er tildelt. Adresseoversetteren bruker de blokknumrene som står i pagetabellen, og lager bare lovlig adresser. Forutsetningen er naturligvis at programmet ikke har adgang til pagetabellen.

KAN ULEMPENE VED PAGING ELIMINERES? HJELPEHUKOMMELSER

Utførelsestiden går opp til nesten det dobbelte dersom pagetabellen ligger i core. Dette kan man unngå ved å bruke hjelpehukommelser. Men la oss først se

* IOT = Input/Output
MST = Masked Set
ION = Interrupt On
INTEN = Interrupt enable
INTDS = Interrupt disable

på hvorfor tiden øker. Pagetabellen står i core, så alle hukommelsesreferanser må gjøres to ganger. Først gjør adresseoversetteren et oppslag i pagetabellen, og så brukes den genererte fysiske adresse til å hente fra eller skrive til core. Den opplagte løsningen er å legge pagetabellen inn i en hurtig registerblokk. Da går utførelsestiden ned igjen. Men dette kan bli dyrt, spesielt hvis pagetabellene blir store. Dessuten kan det være mange pagetabeller, f.eks. i et multiprogrammeringssystem, hvor mange kjører samtidig. Da måtte man skifte pagetabeller ofte, og det ville også ta tid. Det fins en annen måte å få utførelsestiden ned på, nemlig ved å bruke en liten assosiativ hukommelse i tillegg til den vanlige. I denne lille hukommelsen settes de sist refererte ordene i pagetabellen. Når man vet at adresseoversetteren refererer samme ordet i pagetabellen så lenge programmet bruker samme siden, så er det rimelig å tenke seg at det kan spares tid ved at adresseoversetteren først ser etter i den assosiative hukommelsen. Eksperimenter har vist at utførelsestiden går nesten ned til det normale selv med en ganske liten assosiativ hukommelse (4 ord kan være nok).

LOKALITET

Når man skriver programmer som skal utføres under paging, må man være oppmerksom på begrepet «lokalitet». For å illustrere dette begrepet, vil vi vise et eksempel i to utgaver:

adresse/instruksjon	adresse/instruksjon
0300/LDF I ADA	0300/LDF I ADA
0400/ADA,FTALL	0340/ADA,FTALL
01376/FTALL,040001; 100000	0360/FTALL,040001; 100000; 0
01400/0	

Ser vi på den første utgaven, merker vi oss at for å få utført en eneste instruksjon, må vi ha 4 sider i core, fordi hukommelsesreferansene er så spredd. Adressene som brukes, er 0300, 0400, 01376, 01377,

og 01400. Siderne er på 256 ord hver, altså 4 sider for denne ene instruksjonen. Lokaliteten er dårlig! Den andre utgaven derimot bruker bare en side. Alle hukommelsesreferanser er samlet innenfor denne, og lokaliteten er god. Det er innlysende at når flere programmer konkurrerer om plassen i core vil et program med god lokalitet utføres raskere enn et med dårlig. Eksperimenter har også vist dette. Man har loadet store programmer med mange subrutiner, hvor subrutinene er loadet inn i alfabetisk rekkefølge etter navnet. Etterpå ble de loadet i tilfeldig rekkefølge, og antallet missing-page-interrupter gikk ned. Enda lenger ned kom man ved å gruppere sammen subrutiner som kalte hverandre, slik at programmet fikk enda bedre lokalitet. Selv om man har en hel virtuell hukommelse til rådighet, bør man ikke spre seg for mye, men lage konsentrerte programmer med godt organiserte data.

FRAGMENTERING

I pagede systemer kan hukommelsen ofte bli utnyttet dårlig pga. fragmentering, dvs. at core alltid må deles i et antall hele blokker. Dersom ikke programmet fyller et helt antall blokker, vil det alltid gå tapt en del av den siste blokken som programmet står i. Dette kalles intern fragmentering. Pagetabeller tar også plass. Dette kalles tabellfragmentering. Ved å minke pagestørrelsen, minker intern fragmentering, mens tabellfragmenteringen øker. Økes pagestørrelsen, øker intern fragmentering, mens tabellfragmenteringen minker. Dersom man bare ser på utnyttelsen av core, er det bra med små sider. Men dersom man tenker på pagetransporteffektivitet, er det klart at man bør ha store sider, siden disker og trommer er langsommere. Mange paging-systemer har vært lite vellykket pga. disse to motstridende kravene.

Når et program utføres, vil visse deler av det ikke bli referert pga. utfallet av tester som er i programmet. En del av programmet som hentes fra masse-

lageret er overflødig, dvs. core utnyttes dårlig. Vi kan definere kompresjonsfaktor som forholdet mellom antallet refererte pager under en utførelse og det totale antall pager i programmet. Vi ser at dersom pagestørrelsen er lik størrelsen av programmet, er kompresjonsfaktoren lik 1. Dersom pagestørrelsen var ett ord, ville kompresjonsfaktoren nå en minimumsverdi. Dette vil si at små pager utnytter core bedre enn store. Eksperimenter har vist at halvering av pagestørrelsen har ført til en reduksjon av kompresjonsfaktoren på 10–15 %. (Denning 1970.)

PAGINGTROMMER

For å kunne bruke små pager, må pagetransporttiden ned. Dette vil si at det er bedre å bruke tromme enn disk, og det er enda bedre hvis man kan bruke f.eks. en ekstra core store (hvor man ikke kan eksekvere instruksjoner, men hvor overføringstidene er små). I alle fall er det bedre å bruke tromme (på NORD-1), fordi trommen er dobbelt så rask som disken, og også fordi trommen er en pagingtromme. Det vil si at man fra program til enhver tid kan spørre trommen om hvor den er i rotasjonen, og så velge ut trommetransporter etter dette. Dersom man kjører mange brukere samtidig, vil det lett samle seg opp en del ventende trommetransporter. Disse kunne sorteres i forskjellige køer etter sektoradresse, slik at man var sikret minimale ventetider. Spesielt når man kjører mange programmer samtidig er det stor forskjell på effektiviteten ettersom man har en paging-tromme eller ikke.

Pagingsystemer er vanligvis demand paging, dvs. at pager hentes fra masselager først når de er referert. Mens pagen hentes fra masselageret, må et annet program kjøres, for ellers vil effektiviteten gå ned.

PAGING ALGORITMER

Når en page hentes, plasseres den i en ledig blokk i core. For å få ledige blokker, må en eller flere pager fjernes. Her skal vi beskrive tre forskjellige paging

algoritmer, dvs. tre forskjellige prinsipper for å fjerne pager:

FIFO First In, First Out. Dette betyr at når en page skal fjernes, tar man den som har stått lengst i core. Den er enkel å programmere og bruker liten tid. Den virker best på programmer som gjennomløpes sekvensielt, uten for mye løkker eller gjentakelser.

LRU Least Recently Used. Den fjerner pager som har stått lengst ubrukt i core. Pager som blir ofte referert, har størst sjanse til å bli i core.

WS Working Set. Dette er det sett av pager, ved tidspunktet t , som er referert av programmet i tidsintervaller fra $t - \Delta t$ til t .

En page som ikke er referert i dette intervallet, hører ikke lenger til working set og kan fjernes.

Valget av paging-algoritme er viktig. Det påvirker et pagingsystem i vesentlig grad.

GLOBAL ELLER LOKAL DELING

Multiprogrammeringssystemer som bruker paging, deler core mellom brukerne etter to forskjellige prinsipper, global og lokal deling. Lokal deling betyr at hver enkelt bruker har sine egne blokker i core som bare hans program får bruke. Når en ny page må hentes, må en av programmets egne pager fjernes. Dette vil si at utførelsen av dette ene programmet ikke påvirker de andre programmene. Global deling betyr at når et program mangler en page, kan en hvilken som helst page fjernes fra core, uten hensyn til hvilket program den tilhørte. I verste fall kan dette starte en kjedereaksjon av manglende pager, fordi den pagen som ble fjernet, var akkurat den som det neste programmet skulle bruke.

Pagede systemer som bruker global deling, har en tendens til å bryte sammen når core blir for sterkt utnyttet. Da ødelegger de forskjellige programmene for mye for hverandre, og resultatet blir en katastrofal økning av antall pager som transporteres til/fra masselager. Derfor er det bedre å bruke lokal deling.

I denne artikkelen er det prøvd å beskrive fordeler og ulemper ved paging. Fordelene er jo åpenbart store, og med forsiktighet er det mulig å unngå de verste ulempene. Det ser ut som paging er blitt ganske populært, når man ser på det store antall maskiner som har det. Når det kommer raskere masselagre, vil en av de vesentligste hindringene for paging bli borte.

Det er skrevet masse om paging.

Peter J. Denning: Virtual Memory.

Computing Surveys 2, 3, Sept. 1970, pp. 153–189 kan anbefales.

Her fins også referanser til over 80 andre artikler. De som er interessert i paging på NORD-1, kan lese Virtual Memory System for NORD-1. Febr. 71, som kan fåes hos ND.

Distribusjon av ND-Nytt

Det er dessverre ikke mulig for oss med de første nummer av bladet å nå alle som er interessert. ND-Nytt blir sendt uoppfordret til alle firmaer som er medlem av NOCUS, alle kunder og alle som tidligere abonnerte på det gamle ND-Nytt. Tidsskriftet distribueres gratis, og vi ber om at alle som ønsker et eget eksemplar og som ikke allerede får tilsendt bladet om å sende et postkort til redaksjonen.

Bidragstere til ND-Nytt

ND-Nytt's ønskemål er å bli et forum for idéutveksling om datamaskinanvendelser. Vi vet at det finnes et uttall av interessante prosjekter som fortjener å bli bedre kjent. Vi oppfordrer derfor de mange brukere av NORD regnemaskiner til å bli bidragstere i bladet.



Julemotivet er fotografert
fra TV-skjermen til NORD
DISPLAY SYSTEM. Programmert
av Stig Nyberg.