KAN

SPC/3

Rapport over forprojekt

Fortrolig

(Henrik Bøje)
Per Eldon
Thomas Olesen
Bo Schmidt
Claus Tøndering

14. august 1991

Indhold

1. Baggrund

I foråret 1991 blev der nedsat en arbejdsgruppe med den opgave at finde frem til et eller flere forslag til hvordan DDE's næste maskinserie bør konstrueres. Denne nye maskine har fået det foreløbige navn »SPC/3«. Arbejdsgruppens konklusioner og forslag til maskinkonstruktion præsenteres i dette skrift.

Hvorfor skal DDE overhovedet lave en ny maskinserie? Baggrunden er den enkle at Supermax' ydelse ikke længere er tilfredsstillende. Når Supermax sammenlignes med konkurrenternes maskiner, sker det hyppigere og hyppigere at Supermax slet ikke kan følge med rent ydelsesmæssigt.

Det største problem er CPU'ernes adgang til fælleslager. Båndbredden på fællesbussen er for lille, og denne del af Supermax-designet har det ikke været mulig at forbedre i samme grad som de enkelte modulers ydelse i de 9 år Supermax har eksisteret. Yderligere forbedring af CPU'ers og ydre enheders ydelse vil kun i begrænset omfang forbedre Supermax' ydelse, idet disse forbedringer ikke løser det basale flaskehalsproblem i fællesbussen.

Fremkomsten af ny teknologi giver os også en række muligheder som med fordel kan udnyttes: Moderne processorer er godt på vej til at blive 64-bits processorer i stedet for 32-bits, og der er kommet cache coherency mekanismer. Disse teknologier kan ikke implementeres i Supermax på en tilfredstillende måde.

Kravene til I/O er stigende: Ydelsen af de enkelte moduler, det samlede antal moduler og den tilbudte funktionalitet. Det er krav vi har svært ved at honorere i Supermax.

Grundlaget for projektet og udfaldsrummet for maskinarkitekturen har været givet af følgende:

1. Generelle krav som primært er erfaringer fra Supermax.
2. Ydelseskrav som er forventninger til fremtiden.
3. Resourcer og tilgængelig teknologi for DDE.

Punkt 3 har dikteret en række overordnede beslutninger. F.eks. anses det ikke muligt for DDE at udvikle et »Tandem«-lignende fejltolerant system.

2. Generelle krav

SPC/3 skal være en maskinserie, ikke blot en enkelt maskine. Den skal være modulær, udbygbar og skalerbar, så det bliver let at udvide en lille maskine til en større. Vi har fra Supermax gode erfaringer med at anvende de samme basale kort i hele maskinserien; det ønsker vi også at kunne gøre i SPC/3.

SPC/3 skal kunne tilføjes ny teknologi over tid. Det må forventes at større klumper skal udskiftes for at addere ny teknologi. Systemarkitekturen deler maskinen op i et CPU/lager-bussystem og et I/O system. Det forventes at I/O systemet vil være det mest stabile og CPU/lager-systemet vil ændre sig mere radikalt over tid.

Det turde være hævet over enhver diskussion at SPC/3 skal være en datamaskine der er baseret på et UNIX-lignende styresystem, som understøtter standarder som POSIX og XPG3.

Vi kan endvidere se at adskillige af verdens andre maskinleverandører beskæftiger sig med meget store maskiner med ydelse på flere tusinde MIPS, hvorfor vi mener at der er et reelt behov for den slags maskiner. Vi har derfor besluttet at satse på meget store og hurtige maskiner.

Vi forventer endvidere at fremtidens brugere vil køre på X-terminaler.

3. Ydelseskrav

Arbejdsgruppen har forsøgt at opstille en række ydelseskrav til SPC/3. En maksimalt udbygget SPC/3 skal kunne yde som vist i følgende tabel. Det ses at der skelnes mellem hvad systemet skal kunne yde og hvad en maskine skal kunne yde. Dette er fastlagt ud fra at de mange brugere godt kan sidde på flere maskiner i et netværk, mens fx databaseoperationer skal foretages på én maskine.

Krav	Kort begrundelse
1000 X-terminaler på systemet	Vi kunne i dag sælge 500 bruger-systemer. Fremtidens terminaler er X-terminaler
3000 MIPS på systemet	1 X-teminal = 1 MIPS idag = 3 MIPS i 1995 → 1000 X-terminaler = 3000 MIPS
500 MIPS på en maskine	1 Bruger = 1 TPS, 1 TPS = ½ MIPS → 1000 Brugere = 500 MIPS
10Gbyte RAM på systemet	1 X-terminal = 10Mbyte → 1000 X-terminaler = 10Gbyte
50000 processer på systemet	50 processer pr bruger
Busbåndbredde 25 Mbyte/s 375 Mbyte/s i burst på en maskine	Bussen idag mættes ved 40 TPS, 1 Mbyte/s og 15 Mbyte/s i burst → 40 TPS, 25 Mbyte/s og 375 Mbyte/s i burst → 1000 TPS
3000 disk-accesser/s, dvs et diskinterface med min. 50 parallelle diske på en maskine	1 TPS = 3 disk-accesser/s → 1000 TPS = 3000 disk-accesser/s, 1 disk = 60 accesser/s → 50 diske = 3000 disk-accesser/s

Krav	Kort begrundelse
Diskbåndbredde 10 Mbyte/s på en maskine	1 bruger = 10Kbyte/s, 1000 brugere = 10Mbyte/s
Net I/O = 10 Mbyte/s	Disk server: disk I/O = 10 Mbyte/s → net I/O = 10 Mbyte/s, Terminal I/O: et ether-net (1Mbyte/s) = 100 X-teminaler → 1000 X-terminaler = 10 Mbyte/s
Net I/O kræver 125 MIPS	400 Kbyte/s = 1 mioc = 5 MIPS → 10 Mbyte/s = 125 MIPS

Det er imidlertid vigtigt at understrege at hvis det er muligt at opnå at én maskines ydelse opfylder det ønskede krav til systemets ydelse – så systemet altså kan bestå af blot én maskine – vil man få en række andre fordele, som fx fælles proceshierarki og bedre fordeling af processorkraften. Vi har derfor valgt at sætte lighedstegn mellem en maskine og systemet.

4. Styresystemkrav

Det er ønskeligt at styresystemet er et standardsystem. Det vil i denne sammenhæng sige at systemet kan købes fra en styresystemleverandør og med forholdsvis ringe indsats kan lægges ind på maskinen. Det skal dog understreges at denne indlæggelse aldrig bliver banal, fordi der altid vil være tilpasninger at lave i I/O drivere, tilpasninger som følge af den valgte processor, tilpasninger som følge af den valgt bus og cache coherency mekanisme, m.m. Endvidere vil selve den proces at sætte sig ind i en fremmed kerne kræve en del tid.

Mulige kandidater her er:

- SVR4/MP-ES fra USL
- OSF/1 fra OSF
- ODT fra SCO

Valget af styresystem er ikke så meget dikteret af tekniske overvejlser som af politik: På den ene side har det altid været DDE's politik at følge AT&T's (nu USL's) udviklingslinie. Vi har kendskab til SVR4, og besværet med at indlægge SVR4/MP-ES vil formodenlig være mindre end med de øvrige kandidater. På den anden side føler vi at ACE-initiativet muligvis kan blive en succes. Dette tilsiger os at vi bør satse på ODT, som bliver den UNIX-variant der skal køre på ACE's RISC-maskine (kaldet ARC), som SPC/3 efter alt at dømme vil blive kompatibel med (se afsnittet »ARC« senere i denne rapport). Men på den tredie side er der en stærk klike inden for ACE som formodenlig vil prøve at gøre SVR4 til ACE's UNIX-styresystem, og det taler atter for at vi skal satse på SVR4/MP-ES.

Det er altså i skrivende stund ikke indlysende hvilket styresystem der vil blive det sejrende, og DDE må være parat til at sadle om undervejs. Vi anbefaler at DDE i første omgang sigter mod SVR4/MP-ES, og at vi løbende vurderer ODT som alternativ kandidat. Når ODT har nået et stadium, hvor den kan komme på tale som et værdigt alternativ til SVR4, tages styresystemovervejelserne op til fornyet behandling. Sandsynligheden for at vi skal skifte fra SVR4 til ODT vurderes til at være mindst 50%.

Der er et problem i forbindelse med at sikre binær kompatibilitet med Supermax. Ved at vælge SVR4/MP-ES vil vi næppe få uoverstigelige problemer med at få binær bagudkompatibilitet med Supermax-programmer der kører på MIPS i MI-mode, i hvert fald i den udstrækning disse programmer holder sig til XPG3-systemkald. ODT er imidlertid et little-endian system, og der er ingen garanti for at det vil understøtte big-endian programmer. Vi kan så eventuelt selv tilføje understøttelse for vore gamle binære programmer, men de vil i alle tilfælde være et fremmedelement i ODT's little-endian verden.

5. CPU valg

SPC/3 bliver bygget med MIPS' R4000-chip. Dette er vedtaget, dels fordi vi kender R4000, dels fordi ønsket om binær kompatibilitet med Supermax tilsiger det. Det forventes at R4000 over tid vil komme i en række bedre ydende udgaver. Der er indtil videre skitseret processorer med større cache memory og med højere clockfrekvens. R4000's ydelse forventes at være ca. 50 MIPS initialt med efterfølgende 75 og måske 100 MIPS udgaver.

De egenskaber R4000 indeholder tilbydes også i andre processorfamilier, så fra et hardwareteknisk synspunkt kan SPC/3 ligeså godt benytte en anden processor. Et naturligt alternativt valg kunne være RS/6000 der udvikles af Motorola til IBM og Apples fælles maskinserie. Problemet med RS/6000 er primært at der vil gå for lang tid før denne chip er klar, og at der endnu er for få oplysninger tilgængelige om den. Endvidere vil high-end udgaver af RS/6000 formodentlig ikke blive offentligt tilgængelige.

6. Arkitektur

Ønsket om at bruge et standardstyresystem stiller visse krav til arkitekturen. Der er ikke tvivl om at maskinen skal være en multi-CPU maskine, og alle væsentlige standardstyresystemer til den slags maskiner baserer sig på tæt-koblede CPU'er.

Endnu væsentligere er det imidlertid at den applikationsprogramgrænseflade som stilles til rådighed i fremtidens UNIX-lignende systemer baserer sig på tæt-koblede maskiner. Vi kan simpelthen ikke give tilstrækkelig god understøttelse af fremtidens programmeringsinterfaces hvis vi ikke har en tæt-koblet arkitektur.

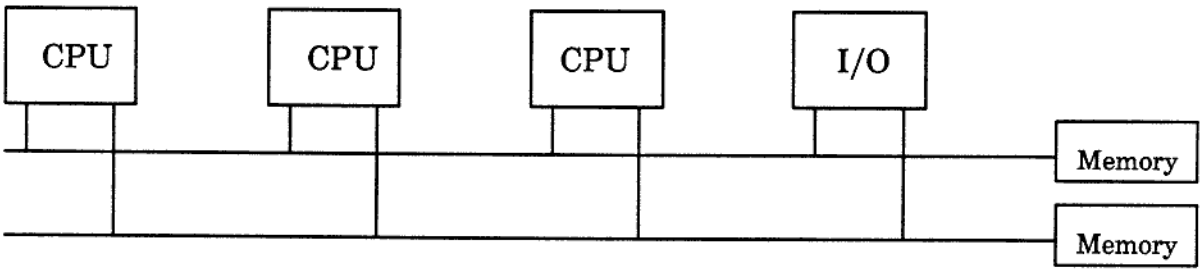
Imidlertid giver tæt-koblede CPU'er en række ulemper. Der er bl.a. ret snævre grænser for hvor mange CPU'er man kan sætte sammen på et fælles lager. En af Supermaxens styrker har været at dens arkitektur er en mellemting mellem tæt og løst koblet.

Det væsentlige problem i at lave en højt ydende multi-CPU maskine er at sikre en høj busydelse. At sætte en hurtig bus på en række tæt-koblede CPU'er er ikke tilstrækkeligt hvis en høj ydelse ønskes. Vi har derfor valgt at benytte mere end én bus, og maskinens topologi er lavet således at busserne bliver så alsidigt benyttet som muligt.

SPC/3 vil blive udviklet efter følgende tre-trins plan:

6.1 Første trin (SPC/3-1)

Første trin er udviklingen af en tæt koblet maskine med dual bus, SPC/3-1. Den ser ud som vist på følgende figur:



Figuren viser et antal CPU'er som deles om noget lager. Der vil kunne benyttes op til 16 CPU'er. Det er en traditionel tæt-koblet maskine, med en enkelt snedighed: Der er to lagerbusser. Ved at lave to lagerbusser i stedet for én, kan maskinens performance øges. Lager fordeles ligeligt på de to busser.

Maskinen vil kunne laves i en billigere enkeltbus-udgave.

Denne maskine vil kunne laves med et standardstyresystem med forholdsvis få ændringer.

Kassen mærket »I/O« dækker over et eller flere kort som enten er »basale I/O kort« eller »I/O bridges«. Det basale I/O kort giver de nødvendige tilslutninger der gør det muligt at lave et kosteffektivt lille system. I/O bridgen skaber en bro til et antal I/O kanaler der kan indeholde et stort antal I/O controllere. I/O kortene vil blive nærmere omtalt nedenfor.

6.1.1 Modeller under SPC/3-1

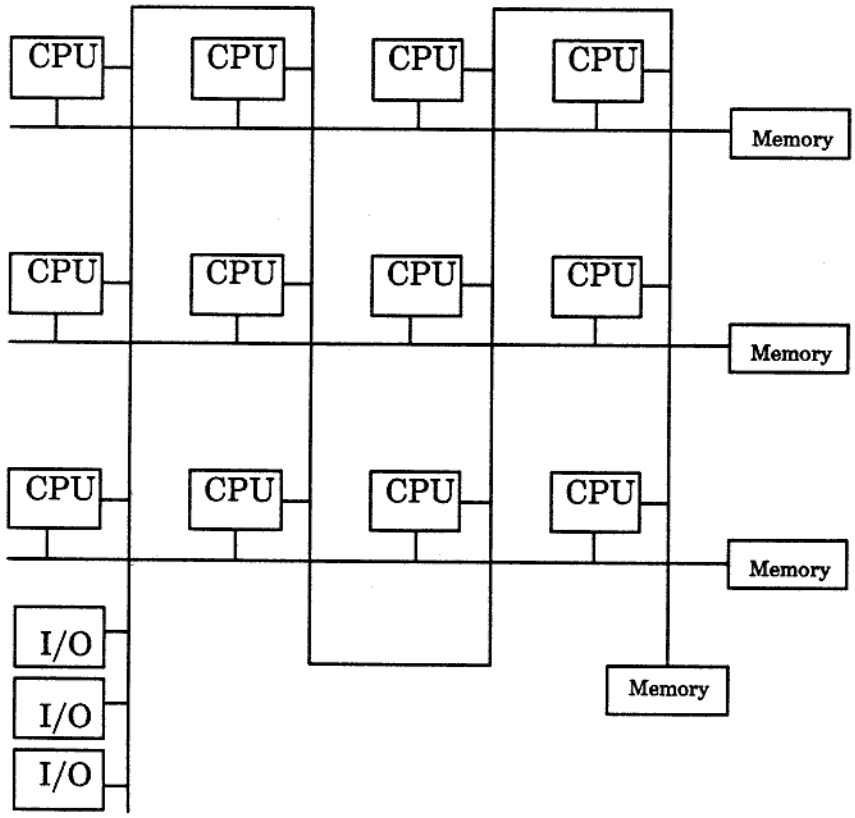
Under trin 1 af SPC/3-projektet kan vi bygge 4 modeller à la Supermax-modellerne med et passende ydelsesoverlap. Som et udgangspunkt laves 4 systemer med henholdsvis 1-2, 1-4, 2-8 og 4-16 processorer à 50 MIPS. Disse modellers sammensætning kan tabuleres således:

	Small	Medium	Large	X-large
Antal processorer	1-2	1-4	2-8	4-16
Antal MB hovedlager	300	600	1200	2400
Antal basale I/O-kort	1	1	1	1
Antal I/O bridges	0	1	2	4
Antal I/O controllers	0	8	16	32
Antal ethernets	1	1	2	3
Bus	Enkelt	Enkelt	Dual	Dual

Som det vil fremgå af beregninger senere i dette skrift vil vi med en ydelse på 50 MIPS pr. processor, få en samlet ydelse på små 600 MIPS for den største model. Den burde med andre ord kunne trække et par hundrede brugere der kører X.

6.2 Andet trin (SPC/3-2)

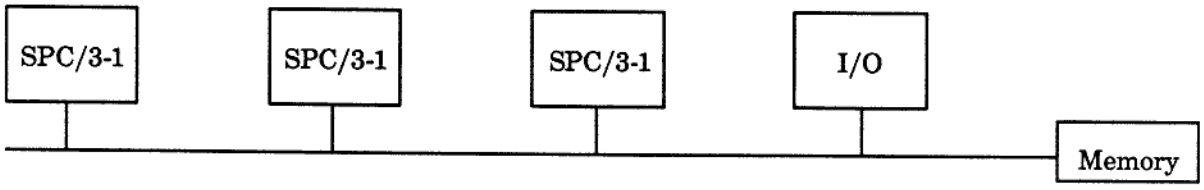
Efter gennemførelsen af første trin – eller måske af lanceringspolitiske grunde samtidig med gennemførelsen af første trin – kan vi gå i gang med at lave andet trin, SPC/3-2. Dette er et meget vanskeligt trin rent hardware-mæssigt, og vil muligvis ikke kunne gennemføres; men forsøget bør gøres. Andet trin består af en sammenkobling af flere første-trins-maskiner hvor den ene bus bruges anderledes:



Her er et antal SPC/3-1'er (med enkeltbus) anbragt i parallel og samtlige CPU'er er forbundet med en anden (fælles-) bus, der er forsynet med noget lager.

I/O-kortene er placeret på fællesbussen.

Funktionen af en SPC/3-2 bliver måske klarere hvis den foregående figur forenkles således:

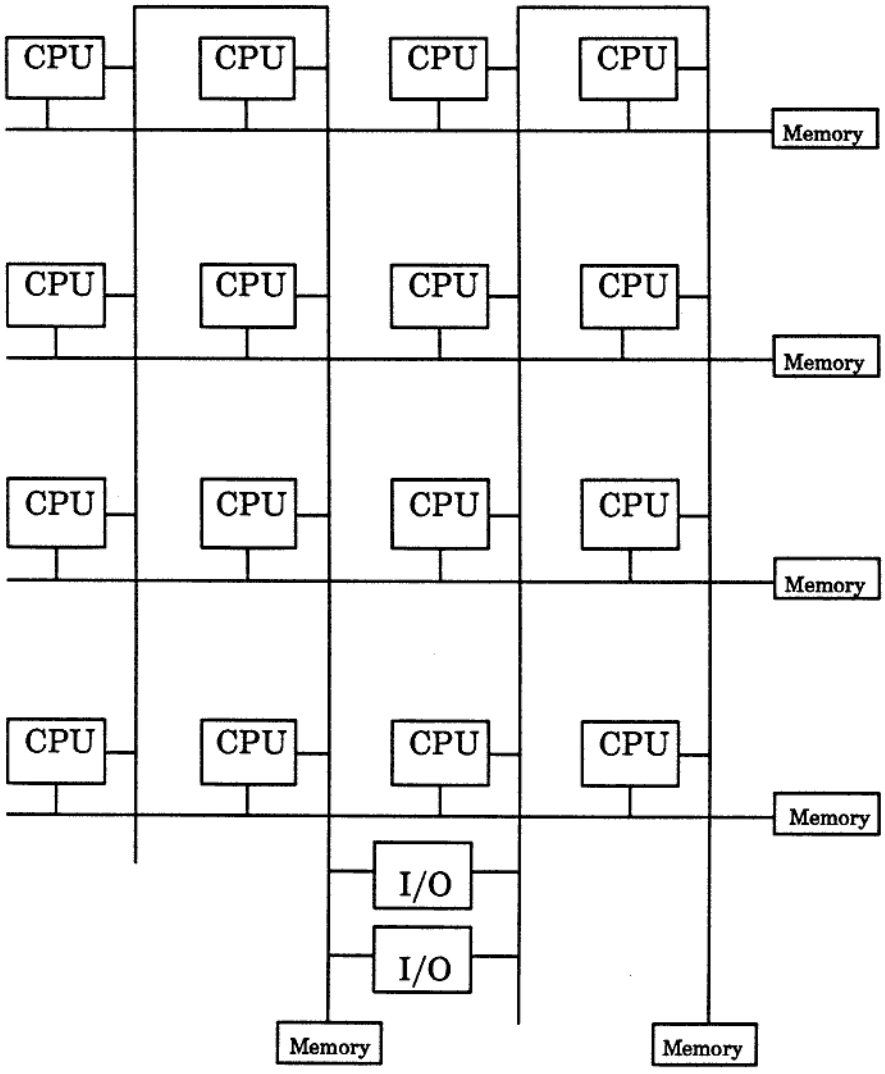


Den intelligente læser vil se ligheden mellem denne tegning og en Supermax. En Supermax er en tæt/løs sammenkobling af et antal CPU'er. En SPC/3-2 er en tæt/løs sammenkoblig af et antal SPC/3-1'er. Inden for hver SPC/3-1 kan man afvikle standard multitrådede applikationer som i alle andre almindelige multi-CPU-maskiner, men desuden har man mulighed for flytning af processer på en måde der minder om Supermaxen og med Supermaxens fordele over den tæt-koblede maskine.

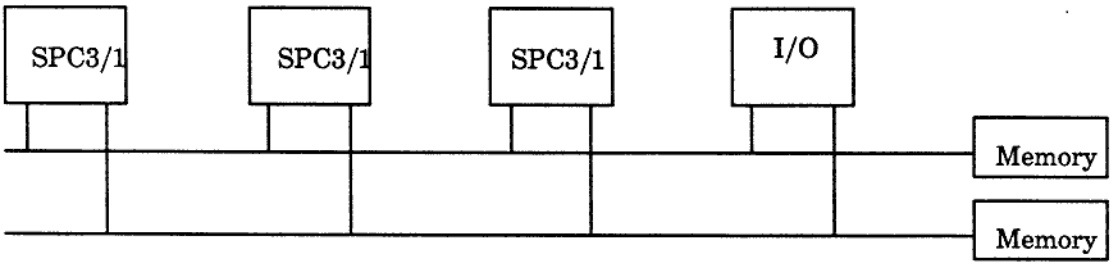
En sådan maskine vil formodentlig kunne indholde 4 rækker à 8 CPU'er.

6.3 Tredie trin (SPC/3-3)

Når andet trin er gennemført og fungerer, kan vi gå i gang med at lave tredie trin, SPC/3-3. En SPC/3-3 minder om en SPC/3-2, hvor fællesbussen er delt i to, hvilket giver mulighed for parallelforbindelse af endnu flere SPC/3-1'er:



Denne tegning vil igen med fordel kunne betragtes således:



Den intelligente læser vil se ligheden mellem denne tegning og en SPC/3-1 med dobbeltbus. En SPC/3-1 er en tæt kobling af CPU'er. En SPC/3-3 er en tæt/løs kobling af SPC/3-1'er.

Vi har men denne topologi det problem, at CPU'erne ikke kan se hele maskinens lager. Vi vælger at løse dette problem ved at begrænse den ene af søjlerne (kaldet »slavesøjlen«) til kun at køre user-mode, hvor vi har kontrol med hvilket lager den ønsker at accesse. Den anden søjle (kaldet »mastersøjlen«) kan køre i både user-mode og kerne-mode.

6.4 Udviklingsmæssige fordele ved tre-trins-metoden

SPC/3-1 er »relativt« ligetil at lave, og det er »relativt« simpelt at lægge et styresystem ind på den. Vi får med andre ord mulighed for at komme hurtigt ud med maskinen. Når så det er sket, vil vi kunne gå videre med de mere avancerede og komplicerede konstruktioner, og selv om vi skulle blive nødt til at opgive disse, vil vi stadig have en god maskine i SPC/3-1.

6.5 I/O-system

I/O systemet består af et basalt I/O kort eller af et antal I/O kanaler med dertil hørende I/O controllere.

Det basale I/O kort tilsluttes direkte i systembussen og indeholder tilslutninger til diske, LAN, konsol, m.v. I/O kortet muliggør et kosteffektivt lille system.

For at kunne skalere systemet passende implementeres det egentlige I/O system som et antal I/O kanaler. En I/O kanal er et standardbussystem der kan indeholde et antal I/O controllere. I/O kanaler forbindes til den egentlige systembus ved hjælp af en I/O bridge. Der er et antal mulige kandidater til I/O kanal bus: Futurebus+, VME, MCA, EISA m.fl.

Der skal udvikles I/O controllere til essentielt I/O. Essentielt I/O er primært I/O hvor ydelsen er vigtig, dvs. disk- og LAN I/O. Det skal vurderes for andre typer I/O om det er ønskeligt/muligt at benytte 3. parts materiel. Bemærk at en væsentlig del af en I/O controller er programmel hvor en del er maskinspecifikt.

Bemærk at brugen af en standardbus som I/O kanal har en række kommercielle komplikationer.

Det basale I/O kort kan muligvis give en brugbar multiprocessorplatform på et tidligt tidspunkt.

6.6 Diske

Som det fremgår af ydelseskravene kræves der ret meget af diskene på maskinen.

Dette kan løses med et array af diske som er koblet tæt sammen på et intelligent kort. Dette kort kan håndtere de enkelte diske, spejling, foldning, sikkerhedskopiering m.m. Udadtil præsenterer dette kort sig som én disk, fx på et eller to SCSI2-interfaces.

Denne konstruktion, som har fået det ambitiøse navn GoogolplexPack*), vil fx kunne indeholde ca. 100 diske, inclusive nogle reservediske der automatisk kan indkobles efter behov.

Et disk-projekt er essentielt for SPC/3. GoogolplexPacken er én mulig løsning. At GoogolplexPacken er en selvstændig enhed giver følgende fordele:

- Den kan udvikles og sælges uafhængigt af SPC/3. Den kan således eventuelt også sættes på en Supermax eller en anden computer.
- Koden bliver simplere og derved hurtigere end på et system hvor de enkelte diske styres separat.
- Spejling, hårde interrupts, svartider m.m. belaster ikke hoved-CPU'erne.
- Den kan simpelt forsynes med egen power-backup, således at data skrevet til GoogolplexPackens cache er at betragte som sikre.

*) En googolplex (dansk: gogolplex) er et spøgefuldt navn på det tal der skrives som et 1-tal efterfulgt af 10^{100} nuller.

7. ARC

ACE (Advanced Computing Environment) er betegnelsen for et initiativ (ikke et konsortium!), hvor adskillige store datamatproducenter (DEC, MIPS, Pyramid, Sony, SGI, Microsoft, SCO, Compaq, Siemens osv. osv. osv.) er gået sammen om at definere PC'ernes afløser. Resultatet er blevet en 4-dobbelt specifikation: 2 hardware-platforme og 2 software-platforme, hvor begge hardware-platforme kan kombineres med begge software-platforme.

De valgte hardware-platforme er dels den nu velkendte Intel x86-baserede PC, dels en ny maskine der har fået navnet ARC (Advanced RISC Computing). ARC bliver en MIPS-baseret multi-CPU maskine.

De valgte software-platforme er dels OS/2, dels en videreudvikling af SCO's udgave af UNIX, der går under navnet Open DeskTop (ODT).

Der foreligger nu en enkelt-CPU specifikation af ARC. En fuld multi-CPU specifikation vil formodentlig først foreligge om ca. to år.

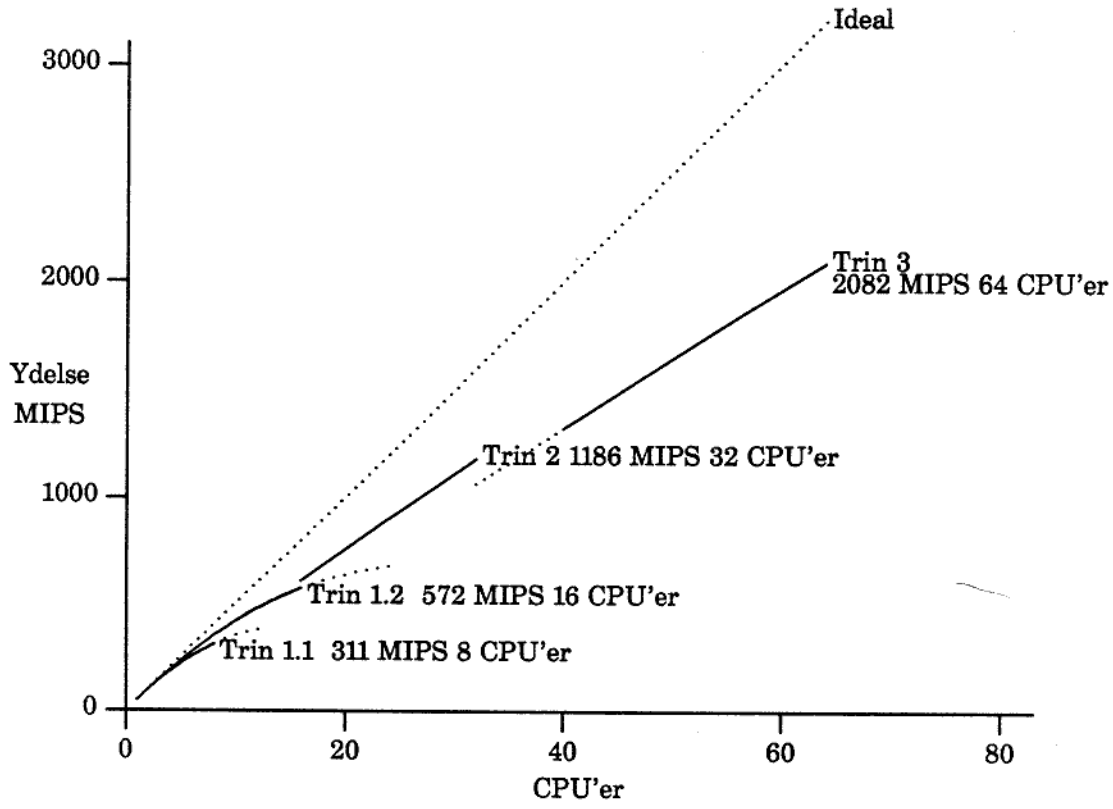
I erkendelse af at ACE kan gå hen og blive en meget væsentlig faktor i fremtidens computerverden har DDE tilsluttet sig dette initiativ.

Det er oplagt at SPC/3 bør baseres på ARC specifikationen. Dette ser foreløbig ikke ud til at ville volde nogen problemer. Den ARC specifikation som vi i øjeblikket har kan vi godt følge. Problemer kan opstå når ACE med tiden fremkommer med en multi-CPU udgave af ARC; på det tidspunkt vil vi nemlig allerede have vores multi-CPU maskine klar. Vi bør derfor løbende følge med i udviklingen i ACE.

ARC specifikationen bør opfattes som en del af designspecifikationen for SPC/3.

8. Vurdering af ydelse

I et forsøg på at vurdere hvorledes MIPS-ydelsen for de forskellige udviklingstrin bliver, er et større simuleringsarbejde blevet gennemført. Den følgende figur opsummerer resultaterne af disse simuleringer:



De fuldt optrukne linier i figuren viser sammenhængen mellem antal CPU'er på en maskine og det antal MIPS man effektivt får ud af den, når der tages hensyn til vort allervæsentligste problem: Busbåndbredde. Ud for hver kurve står den maksimalt opnåede ydelse.

De punkterede forlængelser af de fuldt optrukne linier angiver værdierne for uinteressante konfigurationer.

Den punkterede kurve mærket »Ideal« angiver sammenhængen ved uendelig busbåndbredde, hvor vi får den fulde udnyttelse af de 50 MIPS pr. CPU.

Trin 1.1 er en trin 1 maskine med én bus. Trin 1.2 er en trin 1 maskine med dual bus.

Simuleringer er gennemført ud fra en række forudsætninger:

- Hver CPU kan yde 50 MIPS på den givne bus.
- Hver CPU søger at benytte bussen i en ottendedel af cyklerne.

– Når en CPU venter på en ledig buscyklus venter den inaktiv.

– Accesserne er jævnt fordelt, pånær at 45% af accesserne tilhører byger på 16.

– Det er muligt at koble 16 CPU'er på busserne.

I trin 2 og 3 er desuden gjort følgende antagelser:

– Hver CPU's busforbrug øges med 10% som går til fællesbussen.

– Fællesbussen har den halve ydelse af de lokale busser.

– Det er muligt at koble 32 CPU'er på fællesbussen.

I trin 3 er desuden gjort følgende antagelser:

– CPU'erne i slavesøjlen yder ¾ af normalt, ¼ af tiden går til processkift.

– Antallet af user-mode instruktioner mellem hvert systemkald er det samme som på en Supermax.

– CPU'erne i slavesøjlen belaster busserne som andre CPU'er.

Det ses at man med SPC/3-1 kan komme op på små 600 MIPS, mens man med SPC/3-3 kommer op på godt 2000 MIPS. Disse 2000 MIPS ligger stadig et stykke vej fra vort mål på 3000 MIPS, men må alligevel i første omgang anses for tilfredsstillende. Med en dobbelt så hurtig CPU vil vi kunne komme op på 3000 MIPS.

Det ses endvidere at en SPC/3-3 yder ca. 3½ gang så meget som en SPC/3-1. Simuleringer er også gennemført, hvor det antoges at bussen gik i mætning dobbelt så hurtigt som ovenfor. I dette tilfælde blev SPC/3-3's ydelse næsten 6 gange så stor som SPC/3-1's. I alle tilfælde ses det altså, at der er sund fornuft i at føre SPC/3-serien videre fra trin 1 til trin 2 og 3.

9. Fejltolerance

Fejltolerance har været diskuteret grundigt. Gruppens konklusion er at det ikke er muligt for DDE at udvikle en maskinserie med »Tandem«-egenskaber. SPC/3 vil imidlertid kunne tilbyde »high availability«.

High availability opnås ved hjælp af 4 teknikker:

1. Check (paritet, ECC) af alle adresse- og dataveje. Data beskyttes og checkes i hele systemet så fejl opdages på et så tidligt tidspunkt som muligt og fejlen rettes om muligt.
2. Mulighed for at lave redundante systemer ligesom Supermax »dual hosted disk system«.
3. Anvendelse af spejlede diske (evt. alternativ teknik) kombineret med udskiftning af diske under drift.
4. Power backup på disk og disk cache.

Udskiftning/addering af printkort under drift er ikke mulig, idet ændringerne i en standardkerne vil være for omfattende.

10. Tidsestimater

10.1 Tidsestimater for udvikling af hardware til trin 1

Det følgende tidsestimat er for en trin 1 system »small« maskine. De øvrige modeller fremkommer ved at tilføje tiderne for mekanik, I/O bridge og I/O controlleres.

		Mandemåneder
	1. Produktdefinition	3
	2. Busspecifikation	3
D E	3. Udvikling af businterface	12
	4. Produktion og testteknologi	3
A	5. CPU-modul	5
B	6. Lagermodul	7
C	7. Basalt I/O modul	9
A	8. Backplane	3
	9. Mekanik	6
	10. Diagnostic software	12
	I alt	63

De følgende maskinmodeller indeholder alle I/O kanaler. Der er opgivet estimater for kabinet mekanik, I/O bridge og I/O controllers. Bemærk at der skal udvikles et antal I/O controllers. Bemærk at der også skal udvikles kabinetter til diske.

	Mandemåneder
1. Mekanik	4
2. I/O kanal teknologistudium	2
3. I/O bridge	6
4. I/O controller	9

Eksempelvis er tiden til system »large« med disk, Ethernet LAN, FDDI LAN og WAN

controllers følgende: Tid til »small« + mekanik + I/O kanal + N × I/O controllers = 63 + 4 + 8 + 4 × 9 = 111 mandemåneder.

10.2 Tidsestimater for udvikling af hardware til trin 2

Væsentlige dele af trin 2 er allerede udført under trin 1. De primære problemer skal behandles under trin 1 og trin 2 udgøres fortrinsvis af ny mekanik, backplaneteknologi, test, m.v.

	Mandemåneder
1. Mekanik	4
2. Backplane, test m.v.	9

10.3 Tidsestimater for udvikling af hardware til trin 3

Samme tid som for trin 2.

	Mandemåneder
1. Mekanik	4
2. Backplane, test m.v.	9

10.4 Tidsestimater for udvikling af hardware til GoogolplexPack

Implementeringen af disksystemet er ikke fuldstændigt defineret. Implementeringen kan tænkes at ske som:

1. En integreret del af en SPC/3.
2. Et komplet stand-alone system.

Punkt 1 kræver et processor/lager modul, et disk I/O modul og kabinetter til diske. Kun processor/lagermodulet er udviklet specielt til GoogolplexPack idet disk I/O modul og diskkabinetter alligevel skal udvikles. Punkt 2 kræver foruden udvikling af et særligt kabinet (mekanik) udvikling af et processor/lagermodul. Tidsestimat for de enkelte dele fremgår af nedenstående tabel.

Udviklingstiden for hardware kan muligvis være mindre hvis større dele af de basale SPC/3 hardwaremoduler kan benyttes.

	Mandemåneder
1. Mekanik	4
2. Processor/DMA/memory module	9
3. Disk I/O modul	9

10.5 Tidsestimater for udvikling af styresystem trin 1

Det forudsættes at der benyttes SVR4/MP som p.t. findes i en 386-udgave; DDE må selv porte den til MIPS.

Det skal atter engang understreges at et skift til ODT midt i projektet er en særdeles realistisk mulighed.

		Mandemåneder
1. Programmørerne har sat sig ind i SVR4/MP		2 × antal personer
2. Bootstrapping (PROM) kodet		3
3. HW-afhængige dele af SVR4/MP kodet om til MIPS		
3.1. Exceptionhåndtering kodet	4	
3.2. Interrupthåndtering kodet	2	
3.3. Lagerhåndtering kodet	3	
3.4. Initialisering kodet	2	
3.5. Big/little endian-håndtering kodet	1	
3.6. Håndtering af flydende kreds kodet	2	
I alt		14
4. Drivere til basalt I/O-kort kodet		
3.1. Netdriver kodet	6	
3.2. Diskdriver kodet	5	
3.3. Konsoldriver kodet	2	
I alt		13
5. Integrering af komponeneter foretaget		2
6. Test foretaget		10
I alt		42 + 2 × antal personer

10.6 Tidsestimater for udvikling af styresystem trin 2

Det forudsættes at udviklerne har indgående kendskab til trin 1's styresystem. Da vi i skrivende stund mangler dette kendskab, bør nedenstående tidsestimater tages op til fornyet overvejelse når de første faser af trin 1 er gennemført.

Idémæsig plan for trin 2: Lav ændringer så brugerlager og styresystemkode er lokalt, og shared memory og styresystemdata er globalt. Sæt en markering i prosessers proc-struktur, angivende hvilken række processen tilhører.

Når dette er lavet (første checkpunkt) skulle en lille trin 2 maskine kunne køre. Andre rettelser er at flytte styresystemets stacke til lokalt lager, herved muligøres de store trin 2 maskiner hvilket er meningen med det hele, og vi når andet checkpunkt.

Til sidst kan det være at Oracle eller andre databaseprodukter har brug for rækkelokalt delt lager. Hvis dette skal implementeres uden specielle DDE-features, kan det gøres ved at flytte delt lager alt efter anvendelsen. Dette er sat på som den sidste del af trin 2, men det er ikke en del af det basale projekt.

	Mandemåneder	
1. Simpel programafvikling opnået		
1.1 Rettelser i bootning lavet	1	
1.2 Ny lageralokering skrevet	1	
1.3 Ny procesopstartrutine skrevet	1	
1.4 Proceskøen opdelt, ny schedulering skrevet	1	
1.5 Diskdriver rettet så alt mellemlander i cachen	1	
1.6 /proc-filsystemet rettet	1	
1.7 Swapning rettet	2	
1.8 Test gennemført	4	
I alt for at nå første checkpunkt		12
2. Programafvikling i store konfigurationer opnået		
2.1 Proc- og u-strukturer er opdelt	1	
2.2 Strukturer fordelt lokalt/globalt	1	
2.3 Beskrivelse af løstkoblet miljø tilføjet til strukturer	1	
2.4 Ny icc_call-mekanisme skrevet	2	
2.5 Ny signalhåndtering skrevet	2	
2.6 Steder hvor stacken ikke adresseres lokalt fundet og rettet	2	
2.7 Nødvendige streamsrettelser gennemført	2	
2.8 Test gennemført	6	
I alt for at nå andet checkpunkt		17
Første + andet checkpunkt:		29
Hvis det Oracle eller andre kræver det:		
3. Shared memory flyttet til lokalt lager		
3.1 Lagerallokering rettet	½	
3.2 Pagefault ændret	1	
3.3 Mekanisme der flytter share memory tilbage rettet	1½	
3.4 Test gennemført	2	
I alt		5
I alt		34

10.7 Tidsestimater for udvikling af styresystem trin 3

Det forudsættes at udviklerne har indgående kendskab til trin 2's styresystem. Da vi i skrivende stund mangler dette kendskab, bør nedenstående tidsestimater tages op til fornyet overvejelse når trin 2 er gennemført.

Idéen i trin 3 er at halvdelen af CPU'erne (slaverækken) ikke selv kan udføre systemkald, men lader prosessen vandre over på masterrækken, der svarer til CPU'erne i trin 2.

	Mandemåneder
1. Bootprocedurer tilpasset	1
2. Scheduler ændret	1
3. Lagerallokeringændret	1
4. Pagefault ændret omkring delt lager	1
5. Slaverækkesystemkald skrevet	2
6. Shared memory I/O tilføjet i slaverække	1
7. Test gennemført	3
I alt	10

10.8 Tidsestimater for udvikling af software til GoogolplexPack

Det anbefales at tre softwarefolk arbejder på projektet, hvoraf mindst én har indgående drivereekspertise.

	Mandemåneder
1. Materielfastlæggelse (sammen med HW-gruppen)	2
2. Detailanalyse	3
3. Kerne	2
4. Interface mod SPC/3	3
5. Spejling »on the fly«	1
6. Servicefunktion	2
7. Sikkerhedskopiering	1
8. Kryptografering	2
9. Konfigurering	2
I alt	18

11. Opfordring

Det anbefales at DDE snarest går i gang med at lave SPC/3 efter de retningslinier der er angivet i det foregående.