

SISU **rapport**

nr 17

DVI – Digital Video Interactive
– en teknikutvärdering

Thomas Bill

SISU

Svenska Institutet för Systemutveckling
Box 1250, 164 28 KISTA

DVI – Digital Video Interactive

– en teknikutvärdering

ISSN: 0282-9924

Copyright
SISU – Svenska Institutet för Systemutveckling
December 1991

Innehåll

1. Inledning	1
2. Analog video	3
2.1 Scanning	3
2.2 Upplösning	4
2.3 Rörlig video	4
2.4 Färgvideo	4
2.5 Standarder för analog video	5
3. Digital video	7
4. Presentation av DVI-tekniken	9
4.1 Ett DVI-system	9
4.2 Maskinvara	10
4.3 Komprimering	12
5. DVI-programmering	15
5.1 Programvaror	15
a) Bibliotek	15
b) AVVS Routines	15
c) Graphics Routines	15
d) RTX	15
5.2 Att programmera DVI	16
5.3 Utbildning	17
5.4 Verktyg för DVI-utveckling	17
5.5 DVI-utveckling på SISU	17
5.6 Ny version av DVI	18
6. Alternativ till DVI	19
7. Slutsatser	21
8. Referenser	23

1. Inledning

En viktig del av SISUs verksamhet är att ta in och testa ny teknik som kan förväntas ha stor betydelse för såväl SISUs medlemsföretag som det övriga samhället. En teknik som för närvarande är mycket omtalad och omskriven är multimedia. Med multimedia brukar menas att man blandar flera olika informationstyper som t ex siffror, text, bild, ljud och video i ett och samma datorsystem. SISUs MDI-grupp (Människa Dator Interaktion) har under två år intensivt bevakat utvecklingen inom området, bland annat genom att tillsammans med medlemsföretagen bygga upp ett kompetensnät kring multimedia. En del av denna bevakning har varit att närmare utvärdera en teknik som kallas DVI, Digital Video Interactive, som nu avrapporteras till medlemsföretagen.

Den här rapporten behandlar inte det generella begreppet multimedia utan fokuserar på hantering av video i en persondator, vilket kan förväntas bli en viktig del i många multimediatillämpningar. Att hantera video i en persondator har varit möjligt sedan flera år tillbaka. Då har det dock handlat om användning av så kallade *laserskivor*. Dessa bygger på äldre, analog teknik. Analog videoteknik och digital datateknik är som olja och vatten, de kan existera tillsammans men blir aldrig riktigt blandade. Möjligheterna att med datorns hjälp manipulera analog video är mycket begränsade, för att inte säga obefintliga. Utveckling har därför lett till att flera företag börjat experimentera med digital video. Fördelarna med digital video är många - t ex går det att använda datorn för att redigera videofilmer, olika typer av manipulationer som t ex spegling och rotering av video i realtid går att utföra, och framför allt blir det möjligt att verkligen integrera video med andra datatyper, t ex text, bild, grafik och ljud.

Det stora problemet med digital video är att det är mycket resurskrävande - 1 sekund digital video motsvarar ungefär 22 Mbyte data. Det innebär två saker - för varje sekund video man vill lagra krävs 22 Mbyte lagringsutrymme och för att visa video krävs en dataöverföringshastighet på 22 Mbyte/s. Lösningen på problemen är att komprimera video för att reducera informationsmängden. Ett flertal tekniker håller på att utvecklas. Den teknik som för närvarande håller högst kvalitet är DVI från Intel.

Den här rapporten har skrivits i samband med att SISUs MDI-grupp tagit fram ett DVI-program som gör det möjligt att navigera runt i ett antal videosekvenser. En av anledningarna till valet av tillämpning är att vi velat pröva på en tillämpning som verkligen pressar DVI-tekniken till det yttersta för att få en riktig bild av vad tekniken klarar av. Som rapporten kommer att påpeka så är avsaknaden av moderna verktyg för att bygga användargränssnitt en av de största bristerna med DVI-tekniken.

Rapporten riktar sig *inte* till personer som vill veta vad multimedia är utan riktar sig i första hand till de som kommit över nyfikenhetsstadiet och vill gå på djupet inom en del av multimedia-området.

I kapitel 2 och 3 beskrivs de övergripande principerna för analog respektive digital video. I kapitel 4 presenteras DVI-tekniken och i kapitel 5 går vi in på hur man utvecklar DVI-program och de erfarenheter vi gjort. I kapitel 6 och 7 presenteras alternativ till DVI och de slutsatser vi dragit.

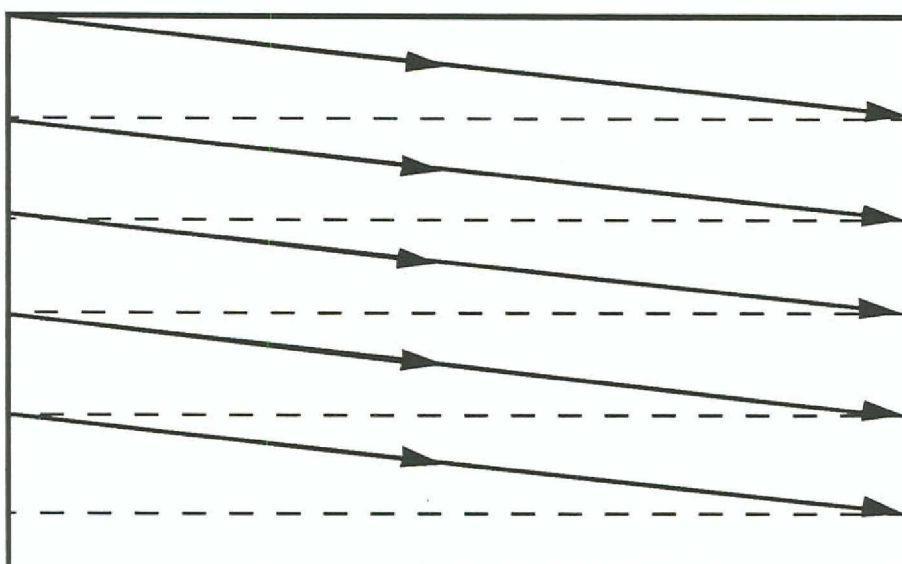
2. Analog video

Innan vi börjar diskutera digital video är det lämpligt att snabbt gå igenom de grundläggande principerna för analog video. Observera att följande genomgång grovt förenklar tekniken och läsaren hänvisas till litteraturen i referenslistan för en mer grundläggande genomgång.

2.1 Scanning

En videokamera omvandlar en tvådimensionell bild till en eller flera elektriska signaler genom att *scanna* bilden enligt ett bestämt mönster. Den ljuskänsliga sensorn i kameran rör sig över bilden tillräckligt snabbt för att inte bilden skall hinna ändra sig för mycket. Den elektriska signalen ändrar sig i förhållande till färg eller ljusstyrka i varje punkt och bildar därigenom videosignalen.

Scanningen går från det översta vänstra hörnet av bilden horisontalt mot högerkanten och bildar en sk *scanning line*, se figur 1.



Figur 1. Scanning, den ljuskänsliga sensorn rör sig från vänster till höger, samtidigt som den rör sig nedåt.

Samtidigt som scanningen rör sig horisontalt mot högerkanten rör sig scannpunkten sakta vertikalt nedåt. När högerkanten nås hoppar scannpunkten över till vänsterkanten och det hela fortsätter tills dess att botten av bilden nås. När botten av bilden har nåtts förflyttar sig scannpunkten tillbaka till startposition. Under den tid det tar att flytta scannpunkten, *the horizontal blanking interval*, stängs signalen av helt och hållet. Den signal som finns mellan två sådana noll-intervall representerar då en bildruta, eng *frame*.

Om videosignalen används för att modulera elektronstrålen i en tv-monitor enligt samma scanmönster som originalet, så kommer den ursprungliga bilden att uppstå. För att elektronstrålen skall följa samma scanmönster som kameran finns även synkroniseringsinformation med i signalen. Synkroniseringssignalen kan sändas i en separat kabel, eller bakas in i videosignalen.

2.2 Upplösning

Ett mått på bildens upplösning är antalet scanlinjer i varje bild. Den vertikala upplösningen är därför antalet horisontella scanlinjer. NTSC (den amerikanska tevestandarden) använder 525 linjer och PAL (den europeiska tevestandarden) använder 625 linjer.

Den horisontella upplösningen definieras av antalet vertikala linjer som kan återges, vilket beror på storleken hos scanpunkten. Ju mindre scanpunkten är desto tätare kan linjerna vara placerade och ändå registreras av scanpunkten.

2.3 Rörlig video

För att videon skall uppfattas som rörlig måste flera bilder presenteras per sekund. De vanliga TV-standarderna presenterar 25 eller 30 bilder per sekund. Men det är inte tillräckligt för att flimmer skall kunna undvikas. Det mänskliga ögat är nämligen mycket ömtåligt för flimmer. För att undvika flimmer visar man istället varje bild två gånger. Varje bildruta består av två fält. Det ena fältet består av bilden som uppstår av de udda scanlinjerna, och det andra fältet består av alla jämna scanlinjer. För att undvika flimmer visas först det ena fältet och sedan det andra. Den här tekniken kallas *interlacing*. För PAL med 25 bilder per sekund betyder detta att 50 fält per sekund visas. För NTSC med 30 bilder per sekund blir det följaktligen 60 fält per sekund.

2.4 Färgvideo

Svartvita bilder består endast av en komponent, luminansen (ljusstyrkan), vilket gör att det räcker med en signal för att koda svartvit video. Färgvideo å andra sidan kan kodas på flera olika sätt.

För att visa färgbilder på en skärm moduleras tre strålar istället för en. En signal för rött, en för grönt och en för blått. När dessa strålar träffar fosforpunkterna på skärmen bildas de önskvärda färgerna i bilden.

Det finns två sätt att lagra en färgvideosignal. Det första sättet är att lagra och distribuera de tre färgkanalerna separerade från varandra. Detta sätt att koda bilden kallas helt logiskt därför RGB (Red, Green, Blue). Ett problem med RGB är att det krävs fyra separata kablar, en för varje färg och en synkroniseringsignal, vilket gör distributionen komplicerad. Det här problemet, tillsammans med några andra, gör att RGB endast används i professionella utrustningar. Istället används vanligen en kodningsteknik som kallas *komposit*.

En kompositsignal kan distribueras via en enkel kabel. De olika TV-standarderna NTSC, PAL etc är alla exempel på kompositformat. Att noggrannare beskriva hur kompositkodning fungerar ligger dock utanför den här rapporten. Ett system med kompositkodning kan sända färgvideo på en signal med samma bandbredd som var och en av de fyra RGB-kanalerna har. Självklart blir inte kvalitén den samma, men det är nästan inte märkbart för tittaren. Vid kompositkodning sänds färginformationen med mycket hög frekvens ovanpå signalen som bär information om luminansen. Om frekvenserna väljs på ett korrekt sätt, sker mycket lite interferens mellan de två signalerna.

2.5 Standarder för analog video

Det finns tre system för TV och video som dominerar i världen; NTSC, PAL och SECAM. NTSC används i Nord Amerika och Japan, SECAM i Frankrike och vissa östeuropeiska länder, medan PAL används i Europa. Dessutom finns lokala varianter på de olika standardformaten. Det som framför allt skiljer dem åt är antal linjer per bild och antalet bilder per sekund.

	NTSC	PAL	SECAM
Linjer/bild	525	625	625
Bilder/sekund	30	25	25

Andra skillnader ligger i hur själva signalen ser ut. De olika systemen kodar t ex färg på olika sätt.

3. Digital video

En vanlig TV-signal är analog och innehåller därför enormt mycket information. För att den analoga signalen skall kunna bearbetas och lagras i datorn måste den digitaliseras, dvs förvandlas till nollor och ettor. När en analog signal konverteras till en digital, så görs det genom två processer, *sampling* och *kvantisering*. Med *sampling* menas att tiden delas upp i diskreta steg och med *kvantisering* att amplituden på signalen görs om till binära heltal.

De två processerna tillsammans brukar kallas för A/D-omvandling. Resultatet blir en sekvens binära tal, där varje tal representerar amplituden under den tid som definieras av samplingsfrekvensen. Ett uppenbart problem är att den analoga signalen består av ett oändligt antal diskreta steg och att vi därför behöver ett oändligt antal bitar för att exakt representera den analoga signalen. När det gäller video begränsas dock informationsmängden av skärmens upplösning. En NTSC-signal består av 512 x 480 rasterpunkter. Trots detta kommer vi snabbt upp i mycket stora informationsmängder. Mängden information kan illustreras med följande räkneexempel:

Anta att vi vill digitalisera en NTSC-signal för video med färg med en hastighet av 30 bilder per sekund. En färgbild kräver 512x480 bitar gånger 24 bitar för färg, det blir 737 kbyte per bild. Vi behöver leverera 30 bilder per sekund, vilket ger 22.1 Mbyte per sekund!

Den digitaliserade signalen måste sedan lagras på lämpligt sätt. Eftersom det är frågan om mycket stora datamängder behövs det mycket stora lagringsvolymmer. Vid produktion av digital video är det nödvändigt att ha ett mycket stort skivminne, minst 1 Gbyte är ett krav vid lite större tillämpningar. För distribution av den slutliga produkten till slutanvändarna är det mer eller mindre nödvändigt att använda sig av CD-ROM-skivor. Priset för att framställa CD-ROM har på senare år sjunkit, vilket gör det möjligt att ta fram CD-ROM till rimliga priser även i mindre kvantiteter. Dessutom finns idag både billiga och bra CD-ROM-läsare på marknaden och en inte alltför vågad gissning är att CD-ROM-läsare ingår som standard i en persondator om ett par år.

Eftersom ett CD-ROM kan lagra 650 Mbyte, får vi plats med ca 30 sekunder video på ett CD-ROM, vilket inte är särskilt användbart. Ett annat problem är att en CD-ROM läsare endast kan leverera 150 kbyte per sekund. Ska vi kunna läsa video från ett CD-ROM måste vi med

andra ord ner till 150/30 kbyte per bild, dvs 5 kbyte per bild. Vi måste alltså minska storleken på en bild från 737 kbyte till 5 kbyte. Med 5 kbyte per bild får vi plats med 72 minuter video på ett CD-ROM, vilket är acceptabelt. Företagen bakom DVI-tekniken har lyckats med konsstycket att komma ner till 5 kbyte per sekund genom att använda en rad olika komprimeringstekniker.

Möjligheten att kunna spela upp fullskärms färgvideo från ett CD-ROM kan sägas vara den absolut största poängen med DVI.

Ett annat sätt att minska mängden information är att minska antalet bilder per sekund och att minska bildytan. Det är en teknik som Apple har använt när de tagit fram sitt system för digital video, QuickTime. Självklart innebär detta betydligt sämre kvalitet.

4. Presentation av DVI-tekniken

Det här kapitlet kommer översiktligt att beskriva DVI-tekniken. Läsare med krav på större detaljrikedom hänvisas till referenslistan sist i rapporten.

DVI-tekniken utvecklades ursprungligen av RCA, som ligger bakom mycket av dagens TV-teknik, men ägs och utvecklas numera av Intel i samarbete med IBM.

Sammanfattningsvis är DVI en maskin- och programvara för att med en dator presentera

- Stillbilder med hög upplösning
- Video med TV-kvalitet
- Stereoljud
- Grafik

Med DVI lagras all information, även video, digitalt på CD-ROM eller skivminne. Idag finns tekniken för DOS och PS/2-maskiner. Macintosh- och Unix-versioner av systemet är också under utveckling. Produktnamnet under vilket DVI säljs är ActionMedia. Det system SISU använt och utvärderat avser Intels version av ActionMedia.

4.1 Ett DVI-system

Det här avsnittet beskriver den DVI-utrustning som har testats på SISU. Stommen i systemet är en 25 MHz 80386 Intel PC/AT med ActionMedia-kort för inspelning och uppspelning av digital video. Maskinen är dessutom utrustad med CD-ROM enhet, bandenhet, 613 Mbyte hårddisk, 16 Mbyte internminne, VGA-kort, SONY Trinitron-skärm för uppspelning av video och en NEC Multisync-skärm som används som systemmonitor vid programmering. Utöver detta tillkommer högtalare som kan skaffas i en vanlig stereobutik, videokamera och om man så önskar en CD-spelare för inspelning av CD-ljud.

De extra tillbehören medför i sin tur att en rad andra tillbehör måste skaffas. Programvaran är Intels systemprogramvara för DVI, produktionsverktyg för bland annat inspelning av video och stereoljud samt C-bibliotek för DVI-programmering. Dessutom behövs Microsoft C. För att konvertera kompositssignalen till RGB har en konverterare av märket VIDI/O använts. Den senare har importerats från Thorn i England.

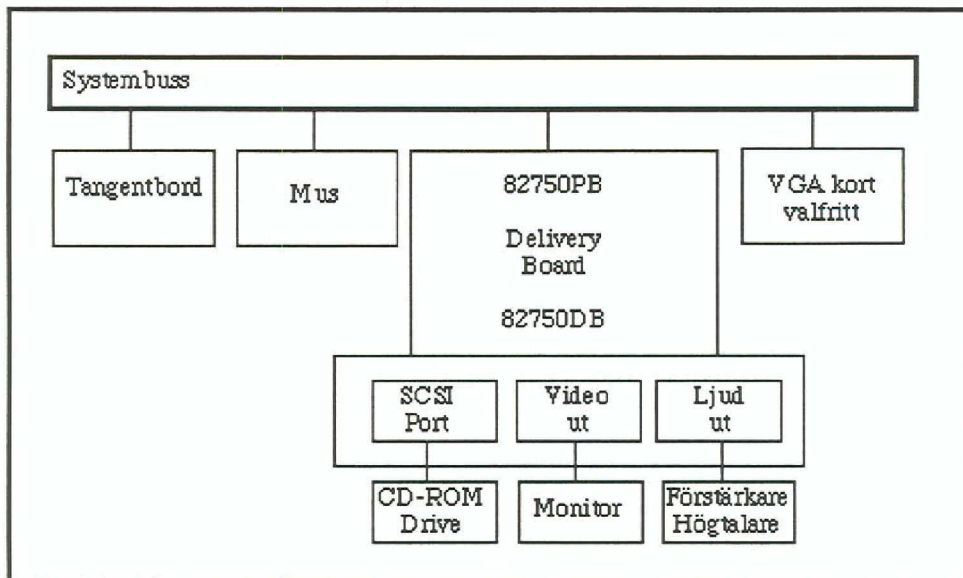
En nackdel med tekniken är det stora kravet på extra utrustning. Eftersom kringutrustningen tillverkas av olika företag är risken för problem vid montering och konfiguration uppenbar. Ett problem i samband med vårt arbete har varit att få fram korrekta konfigurationsfiler för att få de olika delarna att fungera tillsammans. Tillbehören har alla olika krav på t ex minnestilldelning och tyvärr krockar dessa ofta, vilket gjorde konfigureringen av systemet minst sagt tidsödande. Nya problem dyker upp om andra program skall köras med DVI-konfigurationen. Det enda sättet att lösa dessa problem har varit att ha en rad olika konfigurationsfiler som väljs beroende på vilka tillämpningar som skall köras. En klumpig lösning kan tyckas, men tyvärr inte allt för ovanlig i PC-världen.

Den ungefärliga kostnaden för att köpa ett DVI-system som är realistiskt att använda som produktionsmaskin ligger runt cirka 150 000 kronor, vilket förutom ovan beskrivna dator även inkluderar videokamera, konverterare, sladdar, nödvändig extra programvara typ författarverktyg och C-kompilatorer med mera. Därtill kommer kostnader för utbildning i England - cirka 8 000 kronor per person och kurs. Den totala kostnaden för att köpa nödvändig DVI-utrustning och skaffa sig baskunskaper för att komma igång med DVI-produktion ligger runt 200 000 kronor. Detta är dock en låg kostnad i förhållande till arbetskraftskostnaden för att producera en riktig DVI-tillämpning.

En utrustning som enbart ska användas för uppspelning av DVI-program blir avsevärt billigare än en produktionsmaskin. Kostnaden för en sådan blir runt 30 000-40 000 kronor beroende på hur bra persondator man köper.

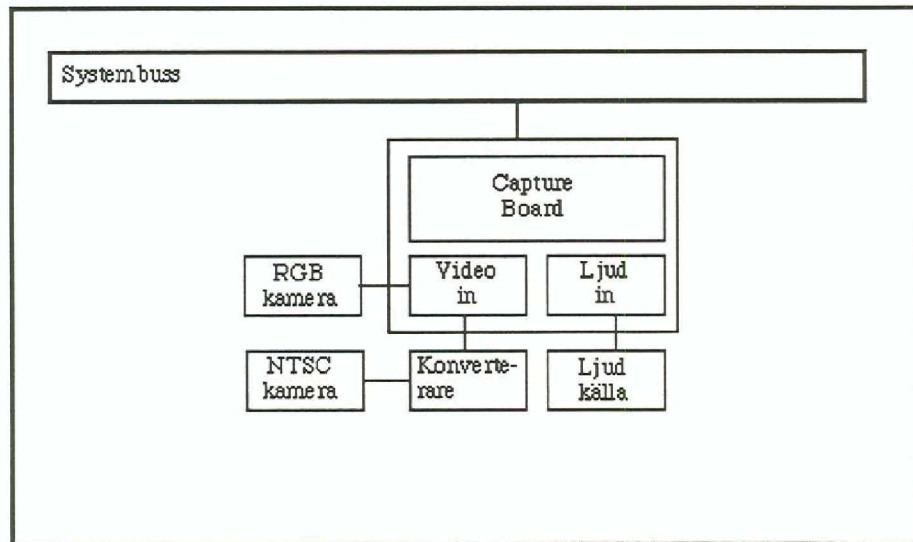
4.2 Maskinvara

Kärnan i DVI-tekniken är Intels video-processor VDP (Video Display Processor), som består av två processorer, 82750DA (DA) och 82750PA (PA). Dessa två processorer arbetar helt separat i förhållande till värddprocessorn. Pixel-processorn PA ansvarar för komprimering och dekomprimering av data, generering av grafik och olika effekter. Alla komprimeringsalgoritmer skrivs direkt i mikrokod för pixelprocessorn. Självklart finns färdiga sådana tillgängliga. Att skriva egna komprimeringsalgoritmer i mikrokod är något för de verkliga proffsen. Mikrokoden är de program som beskriver hur de enskilda instruktioner i processorn skall utföras. Skärmprocessorn DA ansvarar i sin tur för att leverera grafik och video på skärmen, att beräkna värdet för varje pixel och att sköta synkronisering mm.



Figur 2. En systemkonfiguration för uppspelning av digital video.

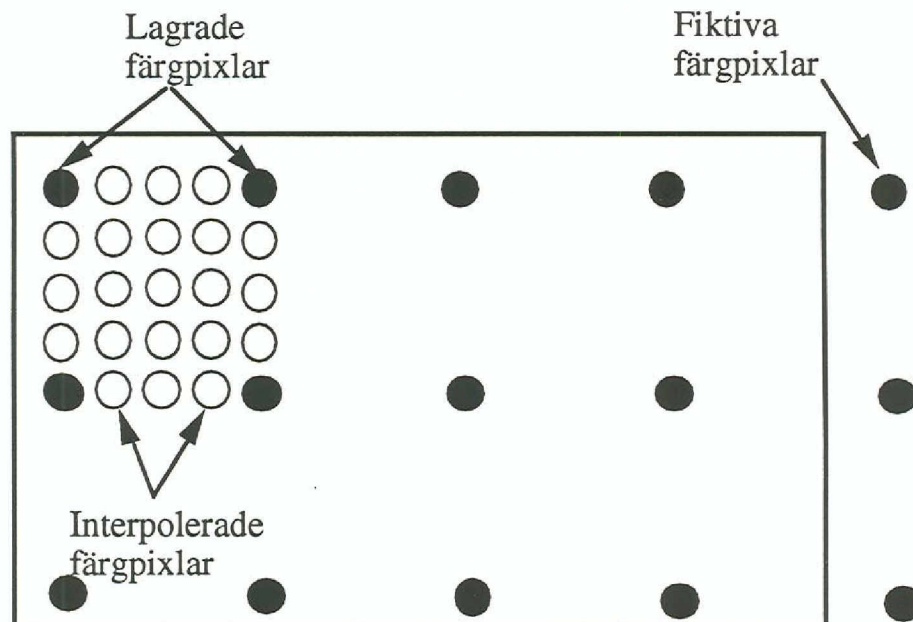
För att använda DVI behövs ett antal extra kort och då först och främst de kort som hanterar video. De två korten som hanterar video går under benämningen ActionMedia Board Set. Korten är ActionMedia 750 Capture Board, för digitalisering av video och bild direkt från kamera. Det andra kortet är ActionMedia 750 Delivery Board för uppspelning av video, stillbilder och grafik på datorn. För att enbart spela upp färdiga DVI-tillämpningar behövs endast Delivery Board. En systemkonfiguration för endast uppspelning kan ses i figur 2. För att kunna spela in video på datorn behövs dessutom Capture Board och en del tillägg som syns i figur 3. Det behövs inga särskilda kommentarer till konfigurationen i figur 2, men väl till den i figur 3. Konverteraren i figur 3 gör om en vanlig komposit NTSC-signal till RGB-signal, vilket är det enda videofORMAT som Capture Board förstår. Detta är något märkligt, eftersom RGB-kameror är mycket ovanliga och nästan alla DVI-användare måste skaffa sig en komposit/RGB omvandlare, som förutom att den är dyr, är svår att få tag på.



Figur 3. Tillägg för att kunna spela in video.

4.3 Komprimering

Något som utmärker DVI i jämförelse med andra tekniker för digital video är komprimeringsteknikerna. Komprimeringsprocessen innehåller flera steg. Speciellt för DVI-tekniken är den sk deltakomprimeringen som mycket förenklat går ut på att endast skillnaden mellan två bilder lagras. En annan metod för komprimering som används tillsammans med deltakomprimering kallas *colorsubsampling*, se figur 4.



Figur 4. Bilden visar principen för hur *colorsubsampling* fungerar. Endast var fjärde färgpixel är lagrad.

Colorsubsampling går ut på att det är möjligt att spara mycket mindre information om krominansen (färgen) i en bild än om luminansen (ljusstyrkan). Forskning har nämligen visat att människan är mindre känslig för störningar i krominansen än i luminansen. DVI utnyttjar detta genom att mindre information om krominansen lagras än om luminansen. Tack vare det går det att reducera informationen om krominansen med en faktor fyra, utan att en tittare upplever någon större försämring av video-kvalitén. Rent praktiskt görs detta genom att endast spara information om krominansen hos var fjärde pixel och sedan genom att beräkna medelvärden ta fram de mellanliggande pixlarna.

Tidskomplexiteten hos olika komprimeringsalgoritmer kan vara mycket olika. Generellt kan sägas att ju mer komplex algoritm desto högre kvalitet efter dekomprimering. Av den anledningen finns det två olika kvalitetsnivåer att välja på vid DVI-produktion. Den nivå som oftast används kallas RTV, Real-Time Video och tillåter att video komprimeras och dekomprimeras direkt på datorn, vilket är en stor fördel i jämförelse med t ex laserskivor som måste pressas innan de kan användas.

RTV innebär en kvalitet som kan vara för dålig för den slutgiltiga produkten, men den duger under utveckling. Kompressionsfaktorn med RTV är ungefär 10. När utvecklingen av tillämpningen är klar går det att skicka videosekvenser till en komprimeringsanläggning i USA, där videon komprimeras på en superdator. Efter den sk superkomprimeringen får man tillbaka ett band eller en CD-skiva med de komprimerade videosekvenserna. Denna kvalitetsnivå kallas PLV, Production Level Video, och kompressionsfaktorn är ungefär 120. Det är denna kompressionsfaktor som krävs för att kunna spela video från CD-skivan.

Den typen av komprimering som PLV utgör brukar kallas asymmetrisk, eftersom den inte går att genomföra på samma dator som sedan skall dekomprimera videon. Tyvärr är det mycket dyrt att använda sig av superkomprimering, ca 1 500 kronor per minut video.

5. DVI-programmering

Det här kapitlet kommer att behandla de bibliotek som finns tillgängliga för att programmera DVI, samt SISUs egna erfarenheter av DVI-utveckling.

5.1 Programvaror

a) Bibliotek

Den programvara som behövs för att köra DVI finns i ActionMedia 750 Software Libraries, som består av fem underbibliotek. För att använda biblioteken skall koden skrivas i C och kompileras med Microsoft C. C-bibliotekets rutiner är uppdelade i olika klasser där varje klass motsvarar ett undersystem i DVI. De olika klasserna är AVSS, RTX, Graphics, Standard Routines och System Routines. Rutinerna i biblioteket är namngivna så att det direkt framgår till vilken klass de hör. Varje klass med rutiner är i sin tur uppdelad i underklasser, vilket medför att det är lätt att hitta de rutiner man vill använda ur biblioteket.

b) AVSS Routines

AVSS (Audio/Video Support System) innehåller rutiner för att presentera komprimerade video- och ljudfiler från CD-ROM eller hårddisk i realtid. AVSS-filer kan startas och stoppas. De går att göra paus på för att sedan fortsätta, att flytta omkring på skärmen och de går att blanda tillsammans med text och grafik. Det går att styra ljudstyrkan och mycket mer.

c) Graphics Routines

I detta bibliotek finns programvaran för att utnyttja coprocessorerna Intel 82750PA och 82750DA. Många rutiner är av traditionellt slag, som att rita cirklar, polygoner och linjer etc. Dessutom finns rutiner för att hantera bitkartor, videoeffekter, komprimering och dekomprimering av video. Det finns även möjlighet att hantera VGA-grafik om ett sådant kort finns installerat på maskinen.

d) RTX

RTX (Real-Time Tasking Executive) är ett realtidsoperativsystem ovanpå MS-DOS. RTX sköter om att spela upp video och ljud simultant, men det går även att utnyttja RTX för mer tillämpningsspecifika uppgifter.

Dessutom finns två bibliotek till, Standard Routines och System Routines, men de är av mindre intresse.

5.2 Att programmera DVI

Själva kärnan i hela DVI-systemet är AVSS. AVSS-rutiner kontrollerar samtliga funktioner som har att göra med ljud och video. Video och ljud lagras på AVSS-filer, som är uppbyggda av sk strömmar. En ström motsvarar en komponent av video-informationen eller en ljudkanal. Det går även att lägga till andra strömmar, tex för leverera data till tillämpningen. Det kan vara för grafik tillägg till bilder eller dylikt.

Förutom vanliga funktioner för att spela, stoppa och stega videofiler så innehåller AVSS funktioner för att kontrollera de flesta parametrar som används vid behandlingen av videofiler, tex var på skärmen videon skall spelas upp, hur många bilder per sekund som skall visas, ljudvolym etc.

Funktioner för att från programmet manipulera AVSS-filer saknas, det finns t ex ingen funktion för att direkt läsa informationen i AVSS-filens huvud där information om filens innehåll finns. För att manipulera AVSS-filer finns istället ett verktyg som ingår i programpaketet, men dessa går inte att styra från ett annat program. Tyvärr är AVSS-filernas uppbyggnad, som mycket annat i systemet, väldigt sparsamt dokumenterat.

En fördel med DVI-systemet är att det till viss del möjliggör *multitasking*, vilket understöds av RTX. Detta system innehåller en samling rutiner som via händelser (eng. events) styr vilka processer som skall köras. RTX fungerar relativt bra, men kan ibland utföra vissa konstigheter. Rutinerna är inte särskilt svåra att använda och är dokumenterade på ett tillfredsställande sätt. En stor miss är att det inte finns några färdiga rutiner för att enkelt använda musstyrning i program, istället får programmeraren implementera dessa själv vilket inte är helt lätt med tanke på alla olika grafiklägen som finns.

I grafiksystemet ingår de flesta rutiner som behövs för att skapa vanlig datorgrafik samt manipulera videobilder. Något som saknas och som borde läggas till är en rutin för att skapa cirkelbågar. Tyvärr är texthanteringen väldigt sparsamt implementerad. Texthanteringen består endast av två rutiner, en för att placera ut en textstäng på valfria koordinater i en bitkarta (eng. bitmap) och en för att beräkna bredden i punkter för en textsträng i den aktuella bitkartan. En egenhet med strukturen hos bitkartor är att de går att placera i vanligt eller utökat minne, dvs i minne som ej är bildminne. Detta verkar praktiskt eftersom videominnet inte är särskilt stort. Det konstiga är att de bitstrukturer som läggs utanför bildminnet är helt oåtkomliga för grafikfunktionerna som inte kan läsa annat än bildminnet. Den här egenheten lär nog försvinna i senare versioner av biblioteken.

Eftersom DVI-systemet är så pass komplext borde det finnas många och bra exempel på hur det skall hanteras. Tyvärr gör det inte det. Istället får man pröva sig fram, vilket kan bli mycket tidskrävande. En "DVI Programmers Guide" borde finnas för att beskriva mer i detalj hur olika saker skall implementeras. Visserligen finns programexempel till varje

funktion i biblioteket men dessa beskriver oftast bara hur den enskilda funktionen skall användas, men inte hur olika funktioner skall kombineras. Det märks klart och tydligt att DVI-tekniken fortfarande befinner sig i utveckling, särskilt på programvaran.

5.3 Utbildning

Komplexiteten i DVI-tekniken gör att det är lämpligt för blivande utvecklare att delta i någon av de kurser som Intel arrangerar. Kurserna håller en mycket hög kunskapsnivå och kräver goda förkunskaper hos deltagarna. Det finns kurser som går mycket djupt in på detaljerna i DVI-tekniken, men även kurser som behandlar de verktyg som finns. Kurserna ges för närvarande i England och USA. Den kurs vi gick i England får fullt godkänt som en effektiv introduktion till DVI.

5.4 Verktyg för DVI-utveckling

Ett av de största problemen med DVI är avsaknaden av moderna verktyg för att bygga DVI-tillämpningar. Det verktyg som vi testat, Authology, ger så dåligt stöd att ren C-programmering är att föredra.

Authology är ett verktyg för att bygga tillämpningar som använder sig av DVI. Det finns stöd för att hantera video, ljud grafik etc. Användargränssnittet byggs upp med direktmanipulation, dvs olika objekt som har med användargränssnittet att göra placeras ut på skärmen med hjälp av musen ifrån en palett av geometriska objekt. Tyvärr är verktygets eget användargränssnitt minst sagt bedrövligt. Det arbetsätt som påtvingas utvecklaren är mycket omständligt. De som är vana att arbeta med moderna grafiska användargränssnitt ledsnar mycket snart på Authology. Ett annat problem är att det är instabilt, programmet kraschar regelbundet, särskilt när det skall hanterat video. Förutom att verktyget i sig är mycket skralt, blir de tillämpningar som skapas med verktyget inte särskilt bra. Allt detta gör att man snart lämnar Authology och istället programmerar direkt i C.

Ett verktyg som rekommenderas av Intel i Sverige är Lumena för redigeringen av stillbilder. Detta har vi dock inte utvärderat.

Ytterligare två verktyg har lanserats på marknaden under den tid vi bedrivit vårt DVI-projekt - MediaScript och D/Vision. Inget av dessa har vi haft möjlighet att testa.

5.5 DVI-utveckling på SISU

I syfte att verkligen testa DVI tekniken har SISU utvecklat en större tillämpning. Tillämpningen gör det möjligt att sätta ihop filmsegment från en lokal eller stad och sedan navigera omkring i den aktuella miljön. När användaren rör sig i korridorer eller på gator finns det hela tiden stöd att tillgå i form av kartor. När det är möjligt att svänga visas detta tydligt för användaren. På vissa platser finns särskild information att tillgå, detta meddelas användaren som själv genom att peka och klicka kan välja att tillgodogöra sig denna. Det kan vara frågan om en

guide som beskriver ett föremål på videon eller en person i ett rum som presenterar sig själv och sin verksamhet. I den tillämpning som tas fram på SISU blir det möjligt att gå omkring på SISU och gå in i olika rum där anställda berättar vad de arbetar med.

5.6 Ny version av DVI

Nyligen lanserades en ny generation av DVI-korten. Skillnaden mot de gamla korten är att de nya korten utrustats med snabbare processorer. En version av DVI-biblioteken för MS Windows kommer enligt Intel att lanseras under våren 1992. Förhoppningsvis innebär det att även bättre utvecklingsverktyg kan komma att utvecklas.

6. Alternativ till DVI

Vilka alternativ finns till DVI-tekniken? En produkt som verkar mycket lovande är QuickTime från Apple för Macintosh. QuickTime är systemprogramvara för att köra digital video på Macintosh, utan extra hårdvara. Vissa grundkrav finns dock på den Macintosh som skall användas. Priset för att kunna köra på vanliga Macintosh-datorer är en betydligt sämre kvalitet än DVI. Det går endast att ha små fönster och antalet bilder per sekund är drygt hälften, dvs drygt 15 bilder per sekund. Idag finns QuickTime bara tillgänglig i en betaversion för utveckling. När produkten släpps finns dock anledning att tro att kvalitén kommer att ha förbättrats. Med dagens maskinvara kommer QuickTime dock inte i närheten av DVI vad beträffar kvalitén. Styrkan hos QuickTime ligger i dess lättanvändhet. Redan nu finns editorer som gör det möjligt att mycket enkelt och smidigt redigera multimedia-dokument. Värt att notera är att QuickTime kommer att vara gratis.

Även andra produkter är att vänta på marknaden. En mycket intressant sådan är Digital Film från SuperMac. Digital Film liknar DVI, men använder sig inte av deltakomprimering, vilket gör att en minut video behöver ca 20 Mbyte. Digital Film kommer endast att finnas för Macintosh.

CD-I är en utveckling av CD-tekniken som Sony och Philips står bakom. Tanken är att skapa en standard för multimediateknik inom konsumentvärlden. På en CD-skiva av ungefär samma typ som idag går det att lagra både ljud och bild blandat. Ljudet spelas upp med den ordinarie stereoanläggningen och bilder visas på TV-skärmen. Än så länge saknas möjligheten att lagra och spela upp video. En nackdel med den här tekniken är att den fordrar mycket dyra produktionssystem. En fördel är självklart den potentiella marknaden för produkter som utvecklas för CD-I. En konkurrent till CD-I på konsumentmarknaden är Commodores CDTV som också bygger på CD-teknik.

Gemensamt för alla alternativ till DVI är att de är steget efter när det gäller komprimeringstekniken.

7. Slutsatser

Helt klart är att de tekniker som skapat DVI har gjort ett fantastiskt jobb. Kvalitén på den video som levereras är mycket god, under förutsättning att PLV-komprimering används. Problemet med DVI-tekniken är inte kvalitén på video, ljud och grafik utan komplexiteten. Att tekniken blir komplex ligger i sakens natur, men så stor del som möjligt av denna komplexitet måste gömmas om tekniken ska bli användbar och få någon större spridning. Bra verktyg kan dölja en del av komplexiteten för en stor del av utvecklarna, tyvärr saknas sådana verktyg för DVI. Att programmera DVI direkt i C kräver mycket djupgående kunskaper om både DVI-tekniken och plattformens operativsystem. Att dokumentationen inte är den bästa gör inte det hela lättare. Ska man använda DVI är det nog ett grundkrav att man går någon av Intels DVI-kurser, som tyvärr inte ges i Sverige för närvarande.

Vi anser inte att DVI-tekniken är mogen för några kommersiella satsningar, men den kommer snart att vara det. Möjligheten att använda DVI i MS Windows kommer säkert att betyda mycket för spridningen av tekniken. Det är därför viktigt att de som vill hänga med i utvecklingen börjar testa tekniken och framför allt börjar fundera på hur video ska kunna användas i olika tillämpningar. Det sista är nämligen en av de svåraste frågorna. Kanske kommer video att kunna användas som hjälpfunktion i olika programvaror? Vilka möjligheter öppnas när det blir möjligt att skicka digital video över telenätet?

För att experimentera med digital video anser vi att det räcker med QuickTime, ställs hårdare krav på kvalitet och prestanda är dock DVI att föredra.

En slutsats vi dragit av vårt DVI-projekt är att multimediaområdet lider en mycket stor brist på standardiserade gränssnitt mellan olika komponenter. I stort sett kan man säga att vi aldrig råkade på två olika komponenter som direkt kunde kopplas ihop med varandra. Problemens svårighetsgrad varierar givetvis beroende på vilken datormiljö man väljer.

Videovärlden i sig är dessutom en mycket stor härva av olika standarder som tar mycket lång tid för en person från databranschen att tränga in i. När videovärlden sedan ska kopplas ihop med datavärlden som har sin egen härva standarder kan problemen bli mycket stora i praktiken. Detta talar alla multimediaförespråkare mycket tyst om idag.

8. Referenser

Luther, Arch C., *"Digital Video in the PC Environment"*, McGraw-Hill, 1989

Harney, Kevin mfl., *"The i750 Video Processor: A Total Multimedia Solution"*, Communications of the ACM, April 1990

Frenkel, Karen A., *"The Next Generation of Interactive Technologies"*, Communications of the ACM, July 1989

Rosengren, Peter, *"Tekniken som gjorde comeback"*, Industriell Datateknik, nr 4 1989

Fox, Edward A., *"Advances in Interactive Digital Multimedia Systems"*, IEEE Computer, October 1991.

Tidskriften "TECHniques for DVI application developers" kan beställas från:

Intel Corp.
313 Enterprise Drive
Plainsboro, NJ 08543