The Personal Computer as an embedded
digital entertainment unit

Daniel Palmqvist
2003-11-27
Abstract

With the growing popularity of the DVD video format, hardware manufacturers have started to include more and more features, in order to make their products more competitive. A modern unit often has support for presenting music and pictures, as well as different variants of digital video. There are however a large number of formats with improved compression ratio. Some of them are still only available for use on a personal computer.

One of the major advantages of using the PC as a digital entertainment unit is its flexibility. While a PC can be upgraded with new software, using these storage formats on stationary DVD players would require the adding of new hardware support.

The main question asked in this thesis, is whether or not a custom PC, built out of standard components, and equipped with the proper software, could challenge mass produced products, in terms of value for the customer.

The first part of the thesis is dedicated to the construction of a prototype entertainment unit and the discussion of possible techniques and approaches. The prototype is then evaluated to establish if the requirements have been met.

The final conclusion drawn is that such a unit can offer all the possibilities and functions of a hardware based player. In the prototype, additional functionality is demonstrated, as a way of showing the increased possibilities with a software-decoding based approach. As a result, a PC based entertainment unit can offer the computer experienced customer a more complete product.
## Contents

1 INTRODUCTION 2

2 BACKGROUND 2

3 PROBLEM DEFINITION 3
  3.1 Evaluation metrics 3

4 HARDWARE PROTOTYPE 4
  4.1 Basic hardware requirements 4
  4.2 Components 4
    4.2.1 Motherboard and CPU 4
    4.2.2 Graphics 5
    4.2.3 Sound 7
  4.3 Assembly of hardware 8
    4.3.1 Case specification and manufacturing 8
    4.3.2 Cooling 9

5 SOFTWARE PROTOTYPE 11
  5.1 Requirements 11
  5.2 Components 12
    5.2.1 Operating system 12
    5.2.2 Third party software 13
  5.3 Development environment 13
  5.4 Basic program layout 14
    5.4.1 Audio player 15
    5.4.2 CD player 17
    5.4.3 DVD player 17
    5.4.4 Movie player 18

6 PROTOTYPE EVALUATION 21
  6.1 Competitors 21
  6.2 DVD player 22
    6.2.1 Hardware 22
    6.2.2 Software and functionality 23
  6.3 Mp3 player 24
6.3.1 Hardware 24
6.3.2 Software and functionality 25

6.4 Movie player 26

6.5 User interface 27

6.6 General hardware and environmental aspects 28

6.7 Economic aspects 29

7 CONCLUSIONS 30

7.1 Continued work 31

8 REFERENCE 32

APPENDIX 35

User group evaluation 35
# Nomenclature

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Meaning / Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC3*</td>
<td>Audio Compression ñ 3. Audio compression format.</td>
</tr>
<tr>
<td>ATX*</td>
<td>Motherboard form factor standard.</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>DAC</td>
<td>Digital Analogue Converter</td>
</tr>
<tr>
<td>DD*</td>
<td>Dolby Digital. Audio compression format</td>
</tr>
<tr>
<td>DivX*</td>
<td>Video compression format.</td>
</tr>
<tr>
<td>DVB*</td>
<td>Digital Video Broadcast</td>
</tr>
<tr>
<td>DVD*</td>
<td>Digital Versatile Disc</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transformation.</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>IDE*</td>
<td>Intelligent/Integrated Drive Electronics</td>
</tr>
<tr>
<td>MFC*</td>
<td>Microsoft Foundation Classes. Programming API.</td>
</tr>
<tr>
<td>Mp3*</td>
<td>Mpeg layer 3, audio compression format</td>
</tr>
<tr>
<td>MPEG(1,2)*</td>
<td>Motion Pictures Expert Group compression formats</td>
</tr>
<tr>
<td>OSD</td>
<td>On Screen Display</td>
</tr>
<tr>
<td>PAL</td>
<td>Phase Alternate Line. TV broadcast signal standard.</td>
</tr>
<tr>
<td>PCB</td>
<td>Printed Circuit Board</td>
</tr>
<tr>
<td>PSU</td>
<td>Power Supply Unit</td>
</tr>
<tr>
<td>RGB</td>
<td>Red Green Blue</td>
</tr>
<tr>
<td>SPDIF*</td>
<td>Sony/Philips Digital InterFace. Digital audio interface standard.</td>
</tr>
<tr>
<td>SVCD*</td>
<td>Super Video CD</td>
</tr>
<tr>
<td>TOSlink*</td>
<td>Optical-digital audio interface standard.</td>
</tr>
<tr>
<td>VBR</td>
<td>Variable Bit Rate</td>
</tr>
<tr>
<td>VCD*</td>
<td>Video CD</td>
</tr>
</tbody>
</table>

*) Trademark
1 Introduction

The era of digital entertainment began in 1985, when Philips first introduced the Compact Disk. Since then the market has grown to include a number of formats and standards for both audible and visual content. All of these standards have their own specifications and hardware requirements. They are often related to machines, designed and optimised for one specific media type like DVD players and Mp3 players.

In recent years the devices have started to include support for several additional formats. A common combination is a DVD, VCD and Mp3 player. All of these formats are related to each other since they are all based on the Motion Pictures Expert Group (MPEG) standards.

In the area of multimedia for personal computers, there are, however, a vast number of less known formats and media codecs. Many of them still have not found their way into embedded implementations, even though they might be more storage efficient than the established de facto standards.

In order to take advantage of these media formats, the decoding unit has to be configurable and upgradeable with new software. Unfortunately most hardware implementations will not fall under that category.

2 Background

As the DVD format grew more and more popular I found myself stuck trying to choose a DVD player. I would probably have bought a cheap unit from the supermarket, if it was not for my special interest in computer hardware. Instead I became increasingly interested in the further investigation of available, possible choices.

The more I looked into the specifications of each product, the more I realized that there was not any product that fulfilled all of my expectations. That was when the thought first occurred of trying to build my own media device out of PC parts. This was both an attempt at constructing a device, tailor made for my needs, but more importantly, also to see how well it could compete with industrially manufactured units.
3 Problem definition

The goal of this project is to investigate the field of custom made, PC component based, digital entertainment units. A focus is to be set on a comparison of such a prototype unit and its mass produced commercial competitors. Can a PC based product offer the customer a higher value for the money?

The prototype is to be constructed out of commonly available computer parts with a total value not exceeding the project budget of SEK 6000. After the assembly of the hardware, it is to be equipped with a suitable set of software components, trying to maximize the available functionality. The prototype must also fulfil the predefined requirements specified in chapters 4.1 and 5.1.

Upon completion, the prototype is to be evaluated using a predefined set of metrics, chosen to represent different possible consumer requirements.

3.1 Evaluation metrics

Some of the obvious metrics in a comparison of consumer electronics are technically related such as quality of the decoded signal, the amount of features available in each unit and expansion possibilities. Many users find simplicity, economics and usability just as important as the performance. These aspects must therefore also be carefully considered. A list of identified metrics can be found in table 3.1.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Picture quality</td>
<td>Theoretical aspects and user group evaluation</td>
</tr>
<tr>
<td>Sound quality</td>
<td>Theoretical aspects and published test results</td>
</tr>
<tr>
<td>Usability</td>
<td>User group study</td>
</tr>
<tr>
<td>Supported formats</td>
<td>Information from data sheets</td>
</tr>
<tr>
<td>Playback functionality</td>
<td>Weighted functionality benefits. User group comments.</td>
</tr>
<tr>
<td>Possible expansions</td>
<td>Discussion around hardware and software capabilities.</td>
</tr>
<tr>
<td>Price</td>
<td>Calc. of prototype expenses. Manufacturer recommended pricing of competitive products.</td>
</tr>
</tbody>
</table>
4 Hardware prototype

The purpose of the prototype was to create a representative hardware configuration, optimized for digital multimedia presentation. As the possible amount of features in the accompanying software would greatly depend on hardware capabilities, it was crucial that the prototype was equipped with proper components.

4.1 Basic hardware requirements

The unit was to be constructed out of PC compliant components. The chosen hardware configuration had to have enough computational power to decode, and process, video and music stored using the commonly available variants of the MPEG compression standard. At a minimum that includes DVD, Mp3, VCD and DivX. It must also be capable of delivering the video signals to a PAL-compatible display unit, and the audio signal to a surround receiver. Since the entire user operation is to be handled by remote, the unit must also be capable of receiving 38 KHz modulated IR signals.

4.2 Components

A standalone DVD player is a highly integrated device with tailor made mpeg decoding circuits and CPU unit all on the same PCB. In a personal computer however, there are modules for each task, like soundcards, memory modules, and graphic cards. Not all of these are well suited for use in this type of application.

The first task faced was to choose the proper parts. One of the primary goals was to offer high quality signal output, and as many of the connector possibilities of a normal DVD as possible. This would also have to be achieved without exceeding the established budget.

4.2.1 Motherboard and CPU

The decoding of a compressed video stream requires much computational power. Older formats such as MPEG1 are less demanding, but for example high resolution, MPEG4 based, codecs such as Divx5 will require a relatively fast CPU to run smoothly and with maximum post process quality enhancement. The system requirement of the Divx5 codec specifies P2-450Mhz as the minimal configuration [1]. Although such a slow processor might be sufficient for currently available codecs, future software will most definitely have higher requirements. Using a faster model also makes it
possible to perform other CPU intensive simultaneous tasks, as well as additional post-processing picture enhancement.

The only two possible brands of CPU's identified as suitable for this project were the Intel Pentium or the AMD Athlon series. The choice of using another, non x86 compatible, CPU would limit the amount of suitable software, and divert from the expressed requirement of PC-compatibility. The advantage of the Athlon processor was a better price/performance ratio than its Intel competitor. On the other hand, Athlons tends to generate more heat[2] and must therefore be equipped with a larger heatsink.

In normal desktop computers, standard ATX form factor motherboards are the most commonly used ones. With the dimensions of 305x244mm [3] that type of board was unnecessarily large to use in this type of project. There are however other, smaller and less common, motherboard formats. Many of them are manufacturer specific ones, used for special series of desktop computers and servers. They are therefore hard, or even impossible, to obtain as separate components.

The product finally chosen was the A7V133-VM micro-ATX motherboard from ASUS. Micro-ATX is a motherboard standard adopted by several manufactures. By decreasing the number of PCI, and memory slots, the dimensions of a standard ATX board is reduced to 244x244mm [4].

The different brands of micro-ATX boards offer a similar standardized set of features and expansion capabilities [4]. The reason for choosing ASUS was the good quality, indicated by a low percentage of faulty cards returned to the retailer. Apart from quality it was also an economic issue. Since the total price of the unit was very important, shipping costs would make it impossible to order each unit from different stores. The board obtained were therefore not entirely optimal for the purpose. One feature it lacked was onboard Ethernet controller. This was however not that much of a problem since it had enough PCI slots, and a 100Mbit Ethernet card is not particularly expensive.

4.2.2 Graphics
Displaying computer generated graphics onto a TV can be done in several ways. One of them is to attach a VGA to S-video converter to the 15pin VGA connector of the graphic card. The advantage of this method is that its computer independent and can be used on any type of graphic card as long as the resolution and refresh frequency
is within an acceptable range. The downside is that it must be put outside of the box, since it uses the VGA connector, and that it can be relatively expensive.

The most common way to get TV output is to use a video card with a TV-out, S-video or composite connector. This normally produces a duplicate of the monitor picture. Some of the cards however, support a dual screen configuration, where the TV-out and VGA outputs display two different images. This feature could be useful in i.e. a future configuration tool, where the tool would be presented on a separate screen.

The motherboard chosen had already a built in graphic card, but since it had no TV-out, an additional video card for the AGP expansion slot became the most appealing solution. It would be possible to construct a home made, simple, VGA to PAL-RGB converter to make use of the built in graphic card instead, but it was considered out of the scope of this project.

In the choice of suitable add-on graphic card, the most important aspect was the visual quality of the TV-out signal. 2D VGA quality, 3D performance and dual screen configurations were also taken into consideration but with lower priority. 2D picture quality becomes important when used together with a projector and 3D performance is important if the box is to be used for playing computer games.

The price of the graphic cards is often related to the 3D performance. In this type of application, were 3D is of minor importance and gaming is not the primarily field of use, a card with decent performance at 800x600 resolution and 32bit colour depth will do. A TV has a resolution of only 720x576 (PAL) and using a much higher setting on the computer will just be a waste of resources.

The cards that were considered for the project can be seen in Table 4.1. There are a vast number of manufacturers so a first selection was made to get a representative selection of different graphical processing units (GPU’s). Some of them, like the Geforce 3, were excluded because of a much higher pricing.

**Table 4.1** Considered choices of graphic cards

<table>
<thead>
<tr>
<th>Manufacturer</th>
<th>Card</th>
<th>Dual Screen</th>
<th>Price (SEK)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATI</td>
<td>Radeon 32MB DDR</td>
<td>No</td>
<td>1200</td>
</tr>
<tr>
<td>Point Of View</td>
<td>Radeon VE 64MB SDR</td>
<td>Yes</td>
<td>750</td>
</tr>
<tr>
<td>Asus</td>
<td>GForce 2 MX400</td>
<td>Yes</td>
<td>1100</td>
</tr>
<tr>
<td>Hercules</td>
<td>Kryo 2</td>
<td>No</td>
<td>1200</td>
</tr>
<tr>
<td>Matrox</td>
<td>G400DH</td>
<td>Yes</td>
<td>1600</td>
</tr>
</tbody>
</table>
The Matrox cards have good TV-out, with lots of features, and are among best 2D cards available for consumer use. They have, however, a very doubtful 3D performance even for this type of application [5]. Since they were the most expensive cards considered this shortcoming was unacceptable.

The Gforce 2 card family on the other hand, has relatively good 3D performance. The two most frequently used tv-out chip-sets on Gforce 2 cards, Chrontel 700x and Brooktree 869, produce lower TV-out quality than the other cards [6]. The picture does not utilize the entire available output area, which results in a big black frame around the picture. This can to some extent be solved with special programs like “TV-tool”, but makes the tv-out configuration both more complicated and expensive.

The ATI Radeon and Kryo cards perform well in all fields, but their price and lack of the dual screen feature made the choice fall on the POV Radeon VE card. The VE card does not deliver the same 3D power as the other two, but it is still sufficient. It has a low price, a good TV-out and passive cooling.

4.2.3 Sound

Almost all stand alone DVD players offer Dolby Digital (DD) output as well as decoded analogue signals. The digitally encoded signal is de-multiplexed from the MPEG2 video stream, and sent to a decoding circuit. This chip can be located in either the DVD itself, an external surround amplifier or an external dedicated signal processing unit.

There are two different standards for the transmission of the encoded signal to peripherals; fiber-optical TOSlink and coaxial SPDIF. Many DVD players offer both of these connection possibilities. Computer soundcards with digital output however, normally only offer the SPDIF connector. Since there are both add-on connectors with TOSlink and SPDIF to TOSlink converters available this does not offer any problem.

As not all consumers have a DD capable surround receiver, the sound card intended must have an on-board decoding chip for analogue low level 6 channel output. Even though DD decoding can be done in software, a hardware solution is preferred to ease the burden of the CPU.

The choice of sound card fell on SoundBlaster Live 5.1 which offers all necessary connectors and features for this project. There are

<table>
<thead>
<tr>
<th>Matrox</th>
<th>G550</th>
<th>Yes</th>
<th>1600</th>
</tr>
</thead>
</table>

The G550 is a 2D card with good TV-out, but with doubtful 3D performance.
other, more expensive, cards available with considerably higher analogue sound quality, but since it is primarily intended for digital output, the sound quality of the built-in amplifier was not a crucial issue.

4.3 Assembly of hardware

After completing the component selection, a suitable casing had to be chosen. None of the cases found at retailers were deemed suitable for this project. They were either too small, in which case the desired expansion cards and motherboard would not fit, or too big. A low profile, VCR-like, box layout would be preferable, as TV furniture often has shelves intended for such products.

4.3.1 Case specification and manufacturing

When deciding the case measurements, there were some limiting circumstances. The height and depth had to allow for a future PCI expansion with Digital video broadcast (DVB) equipment. When examining the specs of several different products it was found that all of them used at least the full height of the metal bracket. The length of the card also varied a lot, depending on manufacturer and the variant of DVB used. It was decided that, in order to preserve upgradeability, the box had to be able to accommodate at least one, standard compliant, full length PCI card. The maximal measurements of such a card is 312 x 106 mm (W x H) [7]. When taking this into consideration the case had to be made considerably larger than previously intended. After doing some simple modeling, simulating different possible component layouts, the final measures of 360 * 360 *123 mm (W*D*H), were reached.

The case itself was made out of 1.5 mm sheet metal, bent to the desired shape. The choice of material fell on plain steel sheets, because they were cheap and easy to machine. The two other options considered, stainless steel and aluminum were both more expensive and difficult to obtain.

The components and layout allows the system to be upgradable with new hardware, including, for instance, an additional harddrive, more memory and a faster CPU. Making the case easy to assemble and disassemble was not prioritized. The time needed to minimize the number of screws needed in the assembly was better spent on other, more crucial, tasks.

One of these tasks was to make robust fastenings. The first idea considered was to build sheet metal holders resembling those found
in most ordinary computer cases. This method would require too much space for the fastening of components on the bottom of the case. Therefore both the hard drive and PSU were instead suspended above the motherboard using a number of bended 20x2.5mm steel bars, as shown in figure 4.1. This construction was easy to manufacture and also provided greater stability to the outer casing.

![Diagram of internal support structures for large components.](image)

**Figure 4.1** Internal support structures for large components. There is an intentional gap between the DVD player and the PSU allowing the use of a PCI expansion card with a length of up to 350mm.

### 4.3.2 Cooling

Many parts, such as harddrive, graphic card and power supply, produce large amounts of heat. The Athlon processor alone can deliver up to 55 Watts of thermal power [8]. This excessive heat must be dispersed or else the lifetime of the parts will be reduced considerably. Keeping the components in a small enclosure makes it even more important to have an efficient airflow.

The biggest problem with cooling is that using fans tends to generate a lot of noise. There are passive cooling methods involving, for instance, large heat sinks to disperse heat and chimney-like tubes to create airflow. None of these methods were very well suited for this project because of the limited space.

A simple method of reducing noise from fans is to lower the supply voltage. The fans used in this project are specified to drop from 36 dBA to 20 dBA [9], when the voltage was reduced from 12V to 5.5V. This was achieved by using a serially connected high power resistor. Lowering the speed also affects the airflow. To compensate for this, a larger fan was used. Most Athlon heat sinks are equipped with 60mm fans, but by constructing a funnel shaped converter it was
possible to use a 92mm version instead. This is illustrated in figure 4.2.

To cool the PSU, a rectangular hole was cut in the top of the protective casing. Over this hole another 92mm fan was fastened.

![Illustration of the CPU cooling system.](image)

**Figure 4.2** Illustration of the CPU cooling system.

In this situation, simply having large volumes of air passing through the heat sinks was not enough. The air confined in the box must be vented to the outside. Adding additional fans to the construction would increase the noise level, which would not be acceptable. The solution was to use the already existing fans to perform two tasks.

Two large holes were drilled in the top of the case. When using the PSU and CPU fans to pull the air from the inside, past the heat sinks, most of the heated air would leave the enclosure instead of being reused. The low pressure created inside will cause new air to enter through several, small holes drilled on the sides. As long as the computer is kept in such an environment that the hot air can be allowed to continue rising, the amount of air that will be reused for cooling can be kept to a minimum.

A logical arrangement of the components will also contribute to efficient cooling. Like the CPU and PSU, the hard drive is also a quite temperature sensitive unit. The amount of heat generated is far less, so by simply placing it in the air flow close to the intake turned out to deliver sufficient cooling. Some hard drives are equipped with internal temperature sensors. Unfortunately the product used had not that functionality. However, by using a thermometer with a separate sensor, pressed against the hard drive, it was detected that, after hours of use, the casing temperature was only approx 35 degrees. Although using an imprecise measuring technique, the actual temperature would still be well below the maximum operation temperature of 55°C [10].
5 Software prototype

Having fulfilled the computational demands for decoding media files, equipping the device with suitable software became the next task. The amount of available software is vast and choosing a functioning mix of programs turned out to be very challenging.

5.1 Requirements

Before looking for suitable software, all desired requirements had to be established. By looking for inspiration in existing products and adding personal preferences, the list presented in table 5.1 was compiled. These basic requirements had to be possible to implement with the software chosen.

Table 5.1 Software requirements

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GUI requirements</td>
<td>The program must have a common GUI for all playback functionally</td>
</tr>
<tr>
<td></td>
<td>Well suited for TV display.</td>
</tr>
<tr>
<td></td>
<td>Remote control manoeuvred.</td>
</tr>
<tr>
<td></td>
<td>Exchangeable backgrounds, colours etc.</td>
</tr>
<tr>
<td>DVD requirements</td>
<td>All common playback control functions</td>
</tr>
<tr>
<td></td>
<td>Possibility to use digital output sound.</td>
</tr>
<tr>
<td></td>
<td>Support for subtitles</td>
</tr>
<tr>
<td>Movie requirements</td>
<td>A DVD-like navigational system</td>
</tr>
<tr>
<td></td>
<td>Support for Mpeg, Mpeg2, Xvid, DivX</td>
</tr>
<tr>
<td></td>
<td>Easily upgradeable with new codecs</td>
</tr>
<tr>
<td></td>
<td>Adjustable zoom and aspect ratio</td>
</tr>
<tr>
<td></td>
<td>Archive system for easy access.</td>
</tr>
<tr>
<td></td>
<td>On Screen Display (OSD) information</td>
</tr>
<tr>
<td></td>
<td>Support for subtitles</td>
</tr>
<tr>
<td>Audio requirements</td>
<td>Archive and play list system</td>
</tr>
<tr>
<td></td>
<td>Onscreen sound visualization</td>
</tr>
<tr>
<td></td>
<td>Mp3 support</td>
</tr>
</tbody>
</table>
After thorough scouting for suitable, existing, software, capable of playing all desired formats, the conclusion was drawn, that none of the programs found were ideal for this task. Especially navigational and archive functionalities were not living up to the expectations. This was probably due to the fact that most products, like Microsoft Media Player, Mplayer and MicroDVD were optimized for ordinary use on a computer. Some of them, like MicroDVD, had adaptations to be better suited for movie playback on a TV, but none of them were designed to be navigated entirely by remote. This made programming a brand new GUI menu system the only reasonable solution. The key issue of upgradeability further strengthened this line of approach.

5.2 Components
Although deciding to program the actual GUI from scratch, there were still a lot of other third party programs needed to achieve full functionality. The most basic hardware I/O had to be handled by an operating system. Additionally a lot of drivers and codecs were needed for media decoding.

5.2.1 Operating system
At first the free operating system Linux seemed like a natural choice for this kind of task. At the start of this project, however, examinations of newsgroups and mailing lists indicated that no unified multimedia playback API existed. Making this application would therefore require the rewriting and joining of existing source code from several other projects. This would occupy a large amount of time, just to get basic playback functionality. That time would be better spent on developing functionality in the actual application.

The choice of operating system fell on Microsoft Windows XP, mainly because of the DirectX API. Normally, programming all codecs (or modifying existing ones) needed for media playback would require an huge amount of time. DirectX offers a software development kit (SDK) that gives programmers a uniform, comprehensive interface to third party codecs. DirectX is available for a majority of the Microsoft operating systems. The reason for choosing Windows XP were partly because of it's short boot time, and partly that DirectX for XP contain additional multimedia functionality, useful in this kind of application. This issue is further discussed in the chapter Development Environment.
5.2.2 Third party software

In order to play movies and sounds, additional software was required. DVD playback was possible through Direct Show filters from the program WinDVD. Besides the movie Mpeg2 decoder, this program also included a software AC3 filter capable of converting the five channel digital surround sound into two channel wave data. This is necessary to get analogue audio when no hardware Dolby Digital decoder is available.

The decoding of Xvid and DivX movies was done with ffdshow, a Direct Show filter capable of generic mpeg4 decoding. The mpeg1 playback capability is handled with filters included in XP.

Regarding audio playback the conceptual thought of using Direct Show for all media formats had to be abandoned. Since audio visualization was one of the requirements, this would require the writing of a custom Direct Show filter, to be able to obtain Fast Fourier Transformation (FFT) data. Doing so would involve spending a significant amount of time would, just to learn the internals of DirectX. Instead, a free mp3 library, FMOD, with these capabilities was used. As a bonus this library also offered playback of older music formats such as FastTracker and ProTracker modules.

To handle IR codes from the remote, a free program called Girder was chosen. The program features a modular expansion system, with a large number of plug-ins. The program is under constant improvement and it has a large forum for developers[11], making it a suitable choice for this task.

5.3 Development environment

Since the project came to rely heavily on parts of Direct Show, the development kit of Direct X had to be used. Most of the available documentation presumed the use of Visual Basic or C/C++ as the programming language. C++, with its object orientation and good performance, became the preferable one in this development project.

Although using DirectX functionality for movie playback, it was not found the most appealing choice when it came to GUI presentation. The more simpler and efficient syntax of the competing standard, OpenGL, made the code more compact and easy to maintain.
Since OpenGL functionality are included with the Radeon drivers, as with virtually every graphic card driver, using it in this project would not require any additional software.

Despite making the program object oriented, it was decided to not include the commonly used object oriented framework Microsoft Foundation Classes (MFC). MFC offers the programmer access to a large number of functions for anything from file management to graphical manipulation. This project would only make use of a very small part of these functions, most of which has counterparts in the Win32 API. MFC is also primarily intended for use with windows style GUI with mouse driven menu systems, and not for this type of full screen application. Including it in this project would therefore only bring unwanted additional complexity to the code.

5.4 Basic program layout
By using C++ the different components of the program could be clearly separated into classes. In doing so, each part could be separately developed knowing that its content would not affect the rest of the program in a negative way. In figure 5.1, an overview of the component layout is given.

![Conceptual overview of the components.](image)

**Figure 5.1** Conceptual overview of the components.
Windows message handling is done in the winmain() main loop and interpreted in wndproc(). Winmain has two operating modes. Based on the required amount of redrawing, winmain uses either GetMessage or PeekMessage. The latter, used in graphic intense situations, creates a loop that runs at maximum speed, grabbing as much CPU time as it can. GetMessage on the other hand freezes the main loop and waits for incoming events. Normally GetMessage is used, but whenever animations are needed, the PeekMessage function is invoked instead.

User interaction is captured by monitoring WM_KEYDOWN events. Once this occurs, the message vParam containing the key code is stored in the globalPref object, where it can be accessed from all program modules.

To allow for remote control operation, external WM_APP messages sent from Girder are processed in the same way as keystrokes except that here the UINT message identifier, not VPARAM, is used to determine which key has been pressed.

Besides message handling, Winmain also requests a graphic redraw, and one message response dispatch, for each pass in the loop. This is done by a call to the object Gui and the methods DispatchRender() and DispatchClass(). From there the requests are directed to the currently active component of the system.

5.4.1 Audio player
The Audio player section is primarily intended for mp3 playback, although several of other file formats may be implemented with little effort.

Keeping the interface easy to use, even when using a remote control, was the most difficult task faced when constructing the player. An important feature was the use of play lists. Besides being able to open Winamp style (.m3u) play lists the player also has an internal list where files can be added. A schematic display of the layout can be seen in Figure 5.2.
The initial thought was to have the media archive and the play list on separate screens, and have the user switch between them whenever needed. It was later decided that a split view with both lists on the same screen were a better solution. The user could then immediately see the result of each command and thereby reduce the need for post-editing of the play list. The downside of this solution was that font size had to be reduced in order to fit all information onto one screen.

To further enhance usability, control of all selections and editing of the play list were limited to the use of only five buttons. Using a remote with four arrow keys located around a select button, made the remote control handling fast, with very limited need for visual orientation.

The Archive is arranged in a tree structure, making it easy for the user to navigate. Each file is a node in the tree, and each directory a branch. The tree can be collapsed to reduce the need for scrolling. Entire branches containing multiple files can be added to the playlist.

The tree is built by traversing and scanning a set of paths, specified in a configuration file for valid media files. This allows the archive to contain not only files stored on the local hard drive but also network resources. Any media present in the DVD player will also be scanned, and included as a directory in the tree.
When playing music for a long time and leaving the GUI screen unchanged for hours, some sort of graphical audio visualizing would be a good feature. It would then serve both as eye candy, and screen saver. A preliminary visualizer was constructed, based upon very simple beat detection algorithms and showing a 3D loudspeaker bumping in pace with the music. Screenshots from the visualizer component can be seen in figure 5.3.

![Figure 5.3](image)

**Figure 5.3** Two of the visualizers used in the audio player component.

### 5.4.2 CD player

Implementing a module with basic CD playing capabilities was given little priority as it was initially considered a simple task. This would turn out to be a false assumption.

The intended approach was to use CD playback control through the Media Control Interface (MCI). MCI is built into Windows XP so no additional software would be required. For the most basic commands, this method was no obstacle. It was later discovered that, for instance, track seeking required two way messages. Implementing this message passing in the present structure would require extensive rewriting.

After some research it was found that the Fmod component used in the Audio Player, also had CD playing routines. Unlike MCI, Fmod uses a one way communication which made it ideal for this application.

### 5.4.3 DVD player

The patent holder of the MPEG2, MPEG LA, demands licensing fees from anyone wishing to implement their compression technique [12]. This has reduced the number of DVD playing software available, especially those free of charge. Fortunately many graphic card
manufacturers choose to bundle software players with their products.

The player used, WinDVD, uses Direct Show compatible filters to render the Mpeg2 stream. This makes it possible to control these proprietary filters from a third party program. The downside is that the manufacturers have made the filters in such a way that they cannot be run in debugging mode. A reasonable explanation would be that it was done to conceal as much as possible of the internals of their product. This made the programming a lot more difficult. Some of the desired OSD functions had to be abandoned because of these debugging difficulties.

As seen in figure 5.1 this module has no Draw function. All GUI interaction with the user could be taken care of using the predefined menu functionalities included on each DVD title.

5.4.4 Movie player

Of all the modules, this was the one to offer the greatest challenge. In order to really make it something special, some, probably unique, functions were to be implemented.

The layout consists of two parts. The primary one is the archive function based on the same principles as the one in the Audio player. This consists of a tree system of directories were all movies can be categorized. If multiple instances with the same directory path and depth are found, these will be merged. This makes it possible to have for instance the sub directory \"cartoons\Henry the hedgehog\" on several separate drives, and their contents will still be merged into a common directory in the tree.

When selecting an item in the tree, an info area appears, showing amongst other things movie duration, current codec used and presence of subtitles. This information is gathered, the first time ever a specific movie is selected. A screenshot from the movie is also captured and stored to disk. The picture is then presented with the information each time this movie is chosen. All of this to make it easier to find the movie in mind.

During playback of a movie, the user can change the aspect ratio and zoom of the movie. This allows for maximal usage of the display unit at hand. If, for instance, presented on an ordinary 4:3 TV, movies using widescreen 16:9 can be zoomed and stretched to utilize more of the available display area.
When playing a movie files in a windowed player there is often a time-slide that can be used to instantly navigate to any desired point in the movie. Besides fast forward and rewind, these are the only common means of navigation within a file. Applying a user friendly, efficient technique in a remote controlled full screen environment would require looking at different approaches.

DVD titles often use chapter selection as the means of navigation. These chapters are fixed anchor points in the time line chosen by the movie company. From a chapter menu, the user can select a scene, and then playback will start at that anchor point. Out of this came the idea to create dynamic chapter selection menus, for any given movie media file.

The menu system was constructed by taking a certain number of screenshots from the active movie, at regular intervals. These were then used as menu items, allowing the user to easily navigate to roughly the right time. Generating the menu takes typically about 5-20 sec, depending on the type of file selected. Once this procedure is completed the images stays loaded in memory until another movie is chosen. This eliminates loading time when additional selections from the menu are needed. The layout of the index can be seen in figure 5.4.

**Figure 5.4** Example of auto generated movie index.

Another feature that was found very handy is continuous playback of segmented movie files. This means that if a movie is divided into two or more files, the program will automatically join them during
playback. The program identifies the files that are to be handled in this way by looking for a short id string to the end of the filename.
6 Prototype evaluation

To be able to properly evaluate the potentials of this prototype it is necessary to compare the achieved result with other similar products. Over the last year the number of products available in this genre has expanded tremendously. With that in mind, making a complete market survey to find the best competitor would be a huge undertaking. This report will therefore have its main focus on comparison with main stream market products produced in large series. Some of the more exotic and challenging new products available will also be included to offer a view of the technical frontline.

6.1 Competitors

As a first evaluation, technical specification comparisons of a few representative competitors were made. Since the prototype is to feature functionality using several different multimedia technologies, the products were chosen to represent different fields of use and price segments. From the DVD player category, a cheap player, Euroline 6632, and a mid price player, Pioneer DV-444S, were chosen. As representatives for mp3 and mpeg4 playback, the Terratec M3Po and Kiss DP-450, were chosen. The comparison data is taken from the manual of each unit [13,14,15,16] and can be found in table 6.1. This information will be further discussed in the evaluation of each prototype subsystem.
Table 6.1 Product capabilities.

<table>
<thead>
<tr>
<th></th>
<th>Euro.6632</th>
<th>DV-444S</th>
<th>M3Po</th>
<th>DP-450</th>
<th>Prototype</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approx price</td>
<td>1000 SEK</td>
<td>3500 SEK</td>
<td>4500 SEK</td>
<td>4000 SEK</td>
<td>5500 SEK</td>
</tr>
</tbody>
</table>

**Features**

<table>
<thead>
<tr>
<th></th>
<th>DVD</th>
<th>SVCD</th>
<th>VCD</th>
<th>Mp3</th>
<th>Mpeg4</th>
<th>Picture CD</th>
<th>CDA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Signal output**

<table>
<thead>
<tr>
<th></th>
<th>RGB</th>
<th>S-Video</th>
<th>Composite</th>
<th>Stereo output</th>
<th>5.1 Analogue</th>
<th>Digital optical</th>
<th>Digital coax</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal output</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>RGB</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>S-Video</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Composite</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Stereo output</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>5.1 Analogue</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Digital optical</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Digital coax</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Storage**

<table>
<thead>
<tr>
<th></th>
<th>CD</th>
<th>DVD</th>
<th>Hard drive</th>
<th>Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>CD</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>DVD</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Hard drive</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Network</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Upgradeability**

<table>
<thead>
<tr>
<th></th>
<th>Firmware</th>
<th>Software codec</th>
<th>Hardw. upgrade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upgradeability</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Firmware</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Software codec</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Hardw. upgrade</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

6.2 DVD player

When comparing this project with budget class DVD players, the benefits of mass production become obvious. For only about 1000 SEK it is possible to obtain a unit, like Euroline 6632, capable of, not only DVD but also, VCD, SVCD and Mp3 playback. When looking at these market products, how does the prototype DVD implementation handle the competition?

6.2.1 Hardware

The manufacturers of the cheaper products often choose to exclude useful features that might not be that obvious to the average consumer. All of this to keep production costs at a minimum. An example of this, taken from the Euroline DVD mentioned above, is the lack of RGB and component video output. The device has separate S-video and composite connectors, but in the SCART connector, only the composite and sound pins are utilized.
When looking at the prototype, there are some obvious drawbacks. The graphic card used, just like the cheaper DVD players, does not feature TV RGB output. Only S-video is available. On the other hand, the Radeon card has very good 2D VGA picture quality [17]. If used with a projector it is possible to achieve very high quality output. Another feature missing is the option to use optical digital output of AC3 and DTS sound, instead of the coaxial connector.

6.2.2 Software and functionality
Although the prototype does not implement all of the DVD functions, it demonstrates some the possibilities. All of the features specified in the DVD standard, like zooming and switching angles, are all included in the Direct Show interface, making them easy to include in the custom player. To give an example of the benefits of software decoding, one can observe the feature “progressive scan”, most commonly found in the high end price segment.

Normal DVD video output is interlaced, meaning that only every other line is drawn on each of the screen update passes. Since there is a time displacement of 1/50 sec between the two parts of a full screen update, there can be some negative effects in picture quality. Interlacing can result in a light flickering behaviour, and jagged edges will also appear when sudden movement occurs in the film [18]. If using a display device capable of handling progressive content, it is possible to circumvent these unwanted side effects, by letting the DVD player merge the two passes into one. If a movie being mastered into the DVD format is inherently interlaced, the deinterlacing process requires advanced algorithms for motion compensation [18].

When used in normal DVD players, this feature requires additional, or more advanced decoding circuits and digital/analogue converters. If relying on software decoding, on the other hand, implementing progressive scan is only a matter of installing free, publicly available, components.

In its present configuration, the box software uses 800*600 pixels as its GUI and movie presentation resolution. This is the preferred mode when using a VGA projector, but for s-video output it results in an unnecessary downscale procedure to get PAL values. It is possible to run some Radeon models in 2D mode at that specific resolution. While reducing, or even eliminating, the additional resize operation, this would in theory result in a higher picture quality. It has, however, not been tested in this project.
6.3 **Mp3 player**

For several years there has been hardware consumer electronics, capable of playing the mpeg layer 3 (mp3) compression format. Integrated functionality to play mp3 files has become more or less standard on recently produced stand-alone DVD units. Since the first stationary CD players with mp3 playback, the signal processing chips have developed, making it possible to also create very compact hand held units.

6.3.1 **Hardware**

The mp3 format is a well documented part of the ISO 11172-5 standard [19]. This strict specification on the encoded data makes it easy to create compatible, specialized integrated circuits with higher efficiency then that of a general CPU. With exception of some hardware decoding circuits having trouble handling high or variable bit rates (VBR), most units will play any mp3 file, as long as it is standard compliant. Software decoding can therefore be found to have very limited benefits in this case.

There are three common methods used by players to store mp3 files: optical storage medium, hard drive and memory chips. Mobile devices tend to prefer the latter because of the small format, low power consumption and superior chock resistance. The downside is that it offers low storage volume at a very high price.

Optical storage media, such as recordable CD’s and DVD’s, represents the direct opposite. They are big, require motors, consuming much more energy, and are sensitive to careless handling. On the other hand, a CD-R offer a price per megabyte that is only about 1/500 of that of compact flash memory cartridges.

Mp3 capable DVD players find their “archive” limited to the space available on a CD or DVD. The prototype box allows playback from both hard drive directories and exchangeable optical media. Hard drive with its significantly larger store space, offers continuous playback for longer periods of time, with no need for media replacement.

There are however commercial products using the same solution with both hard drive and CD’s. One example is Terratec M3Po, which also has internal mp3 encoding capabilities, making it possible to compress CD Audio and store it on the hard drive [15]. This type of specialized product is more uncommon, and has a higher price tag. In the case of M3Po it is approx 4300 SEK.
The hardware used in the box, well exceeds the computational demand of software mp3 decoding. In fact, when played on the prototype it only uses a few percent of the CPU time. The only hardware limitation in this construction is the mediocre quality of the cheap soundcard digital analogue converter (DAC). Using the digital output to an external DAC would therefore be recommended for greater signal accuracy.

6.3.2 Software and functionality

The solution with play lists, used in the prototype software, is also adopted in several DVD players and specialized mp3 players. A play list system offers a flexible way of arranging continuous playback. The simplest devices, however, offer only selective or sequential playback of the files, just as normal CD-audio players do.

Another benefit of using software decoding is that adding functionality, requiring specific types of information from the audio stream, is relatively easy. For the constructors of hardware based decoding, this is not all that easy. To implement, for instance, spectrum visualization, the integrated circuits has to be able to generate and deliver FFT data to the GUI program. Although not having any audio enhancing functionality, adding different kinds of visualisation adds an additional dimension to the multimedia experience.
6.4 Movie player

Movie player is the only software subsystem with very few competitors on the market. A handful of products, capable of playing Mpeg4 compatible video streams, are available. One of the first, Kiss DP-450, is based upon the highly integrated Sigma Designs chip EM8500. This single IC features hardware decoding of Mpeg1, Mpeg2 and Mpeg4 video. There is also built in 150Mhz CPU that can be used for GUI control, and IDE interface for direct connection to standard PC DVD units. All of these features in one chip makes it easy for hardware manufacturers to produce Mpeg4-capable DVD units. In a near future this functionality will probably become a standard feature in the same way as mp3 audio decoding.

As previously shown, hardware based decoding can not be upgraded with new functionality. This aspect becomes very clear in this specific field of use. There are several competing companies and programming groups, involved in development of Mpeg4. Four of the most commonly used codecs are: Microsoft Windows Media, Apple QuickTime, DivX and Xvid. All of these can generate Mpeg4 compliant output. Unfortunately that is not the case of some older software, like DivX 3.11. If the person compressing the video is not aware of this, the result might be an invalid file. There are also additional, quality enhancing, features added to recent versions of the encoding software. These are not supported by the EM8500 card, making the file impossible to play.

What is even worse is that there is a whole jungle of sound compression formats, all of which can be used together with an mpeg4 video stream. The AVI and MOV file formats both allow independent video and audio sources, only limited by the presence of compatible codecs. The hardware decoder expects Mp3 or AC3 encoded sound, but the audio part of the file might instead use, for instance, the popular open source format OGG. All of these problems are non existent on the prototype, as long as the proper software is installed.

Some of the prototype features, such as generated indexes and auto de-segmentation seems to be unique features for this software. In the search for competitive products, they have not been spotted in any other movie player of this kind.
6.5 User interface

Products using hardware based media decoding tend to use limiting techniques of generating OSD menus. Even in new chips like the EM8500, graphics is only allowed to have 8 bits of colour and 8 bits alpha transparency. Older video processing chips like ES4408F, offer 4 bits colour and 4 bits alpha [20]. This means that high quality graphic images can not be used in the menu system without down sampling of the colour information. When using software like DirectShow to implement OSD overlay, it is possible to use 24 bit graphics and 8 bit alpha, allowing 16.7 million colours instead of 16 or 256 [21].

Mixing 3D and 2D graphics in the way done in this program would be difficult to do on ordinary consumer electronics. Generating moving 3D graphics requires heavy use of floating-point precision calculations. This requires either 3D accelerating hardware or a quite powerful CPU, none of which are present in most products.

However, as long as the menu system contains the necessary functionality and is simple for the user to operate, all of these graphic capabilities are to be considered an aesthetic bonus. In the prototype software, the menu systems are all but complete. Lot’s of options and settings are to be implemented in order to reach full functionality.

In an attempt to get the user opinion of the user interface, a small user group evaluation was preformed. The details and results of this evaluation can be found in appendix 1. The overall opinion of the system was rather good, but the results also indicated a number of shortages. One of these was the remote handling not being smooth enough. The reason for this is uneven message parsing from the Girder remote program. Using another remote, with a different signal interval might produce better results. Another big flaw in the user interface was the shortage of OSD information in the movie and DVD subsystems, regarding run time of movies, chapter selection etc. This must be implemented in future versions in order to reach a satisfactory result. The overall speed of the automated index generation must also be improved.
6.6 General hardware and environmental aspects

One of the major hardware requirements of the prototype was high quality video output. To evaluate the actual results, the user group study, mentioned in 6.5 was also asked to give an opinion on the TV picture quality. The results were uneven. Compared to a normal stand-alone DVD player, the prototype produced a slightly less sharp image, but was instead more stable. The DVD player displayed video with a slight flickering behaviour. This was especially noticeable in the on screen display menus. The reason for this could not be established. Regarding the softer image, one probable cause would be the 800*600 to PAL downscale operation mentioned in 6.2.2. Implementing native PAL resolution in the software would minimize the quality loss.

Most of the major prototype drawbacks are related to the concept of only using standard PC components. Making stand-alone DVD players slim and light is easy because of highly integrated small PCBs. Making small cases for generic PC hardware is much more difficult. PCI expansion cards, heat sinks and storage devices all require a lot of space. The finished prototype is quite heavy and substantially larger than its competitors. Since it is intended for stationary use, the weight can be considered less important.

Because of the large amount of heat produced, it is very important that the unit is not put in confined space. The air from the fans has to be able to flow freely to keep the electronics at an acceptable temperature. Other products in this segment produce less heat and are therefore not as sensitive. An ordinary computer often has many sources of irritating noise. With a careful selection of hard drive and cooling techniques, the noise level could be kept at a minimum.
6.7 Economic aspects

When buying a piece of customer electronics it is easy to establish the cost. In the case of the prototype this turns out to be much harder. As seen in table 6.2, the total amount spent on hardware fall well within the established budget. These figures were easy to calculate, but how do one value the time needed to be spent on manufacturing, programming, installation and configuration? These parts of the project well exceed 600 hours of work. If considering that potential cost, the budget could never have been kept.

The development of a custom case and software stands for the vast majority of the time spent. It is therefore likely to believe that in the future, with more products already adapted for this field of use, the required amount of work would have been considerably less.

Table 6.2 Hardware costs.

<table>
<thead>
<tr>
<th>Product</th>
<th>Price (SEK)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maxtor 60 GB D540</td>
<td>998.00</td>
</tr>
<tr>
<td>POV Radeon VE</td>
<td>544.00</td>
</tr>
<tr>
<td>ASUS A7V133-VM</td>
<td>743.00</td>
</tr>
<tr>
<td>Pioneer DVD 106-S</td>
<td>727.00</td>
</tr>
<tr>
<td>AMD Thunderbird 1000 MHz</td>
<td>871.00</td>
</tr>
<tr>
<td>Creative SB Live</td>
<td>332.00</td>
</tr>
<tr>
<td>NIC 10/100</td>
<td>75.00</td>
</tr>
<tr>
<td>128MB SDRAM</td>
<td>204.00</td>
</tr>
<tr>
<td>ATX PSU</td>
<td>300.00</td>
</tr>
<tr>
<td>IR receiver</td>
<td>110.00</td>
</tr>
<tr>
<td>Case material</td>
<td>130.00</td>
</tr>
<tr>
<td>Paint</td>
<td>180.00</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>5214.00</strong></td>
</tr>
</tbody>
</table>
7 Conclusions

The question asked in this thesis is, whether or not, a custom made PC could offer the customer a higher value for the money then its mass produced commercial competitors. With the building of a prototype, both new possibilities and difficulties have been discovered.

The major advantage of this type of approach to digital home entertainment is the flexibility. The concept can be modified to support almost any digital media format imaginable. If the computational demands are too high, using modular hardware design based on PC components makes it cheap and simple to upgrade with new, improved hardware. The use of technologies like Direct Show also makes the software easily upgradeable with new playback codecs. With a combination of software, powerful GPU and CPU, it is also possible to implement more advanced and customizable GUI’s then the ones normally present in stand-alone DVD players.

The shortcomings found when making the prototype are not to be neglected. The most serious ones are economically related. With a total hardware cost of approx. SEK 5500, and a commercial operating system, the price tag on this type of product would be significantly higher then it’s competitors. The picture output quality is not quite as high as it should be, and there remains much to be done in order to achieve the same level of functionality as the other players.

Implementing all functions was, however, not the intent of this project. The main idea was to show the possibilities and disadvantages of working with PC components, and in doing so discover if this approach has potential to offer the customer more than ordinary products.

During the time of this project, the market for this type of products has virtually exploded. Now there is a variety of hardware and software, made exclusively for these type of applications. An example of this is the VIA ITX motherboard. It features an intergrated 800 MHz processor, on board video card with TV output and sound with SPDIF connector. The processor is x86 compatible and produces very little heat. All of this is squeezed onto a 170*170mm PCB. The prices for these components have also dropped considerably.
There are now several open source projects focusing on offering the same type of functionality as the prototype. With teams of experienced programmers, making software such as the one in this prototype, the result can be achieved very quickly. With several of these projects programming free software for Linux, it certainly makes that operating system a very attractive choice.

The practical result of this project must therefore be considered, to some extent, obsolete. However, the evaluation shows that the prototype is capable of offering the user more advanced features, not present in the commercial competitors. Some of them, like the auto generation of a chapter index in the movie subsystem, are probably unique in this context. So with the shortages reported in the user study fixed, the prototype would certainly have a technical upper hand. With the development of new software and further reduced prices on, even more suitable, components, the PC-approach will further strengthen its position.

Because of the amount of effort required for installation and configuration, it will still not be an alternative to the average consumer, but rather a bonus for those who have adequate computer knowledge. For those fitting that category, however, a PC, tailor made for digital entertainment, will be the better choice.

7.1 Continued work

From being a rather new idea, the software part of this project now got company with several other programs, like the open source “Freevo”. Many of them are under constant development. The prototype software is therefore not able to offer the users as much functionality as its competitors. Keeping the software up to date would require a large amount of work. Hopefully the concept of the best prototype features, like the 3D menus and auto indexing, will find its way into the competing projects, so that everyone can enjoy their benefits.
8 Reference

[1] DivX Minimum System Requirements
DivXNetworks, Inc
http://www.divx.com/divx/system_requirements.php
2003-06-03

[2] Processor Electrical Specifications
Chris Hare
http://users.erols.com/chare/elec.htm
2003-05-14

Page 9
Intel Corporation
http://www.formfactors.org/developer/specs/atx/atx2_1.pdf
2002-06-01

[4] microATX Motherboard Interface Specifications,
Page 9
Intel Corporation
http://www.formfactors.org/developer/specs/microatx/matrix1_1.pdf
2002-06-01

[5] Matrox Millennium G400 & G400MAX
Anandtech
1999-05-20

Anandtech http://www.anandtech.com/guides/viewfaq.html?id=29
2001-06-13

[7] PCI Compliance Checklist Rev 2.2
Page 13
PCI Special Interest Group
http://www.pcisig.com/data/specifications/2_2_checklist.doc
1999-08-15

[8] AMD Athlon Processor Model 4 Data sheet
Page 33
AMD Inc
http://www.amd.com/us-en/assets/content_type/white_papers_and_tech_docs/23792.pdf
2001-11-01

[9] NP Fan ZM-F2
Zalman Tech Co. Ltd
http://www.zalman.co.kr/english/product/cnpsfan-f2.htm
2003-06-12

Maxtor Inc
2002-11-11

2003-04-05

[12] An overview
MPEG LA
http://www.mpegla.com/mpeg2/m2_overview.htm
2003-04-04

http://www.leien.info/euroline/
2003-06-14

[14] Pioneer Europe, Online PDF manuals
Pioneer corp.
http://195.61.130.92/PEEServiceRegistrInstrMan.nsf/OnlineHomePage2?
2003-06-14

TerraTec
http://www.m3po.net/m3po_manual_eng.pdf
2000-05-18

Kiss-Technology A/S
2003-06-14

[17] ATI Radeon VE Review
Jupitermedia Corporation
http://www.sysopt.com/reviews/radeonve/index10.html
2003-06-10

[18] DVD Player Benchmarks, Progressive Scan DVD
Don Munsil, Brian Florian
2003-01-01

[19] MPEG-1 Description
Leonardo Chiariiglione
http://mpeg.telecomitalialab.com/standards/mpeg-1/mpeg-1.htm
1996-06-01
[20] ES4408 DVD Processor Product Brief
ESS Technology Inc.
2001-05-21

[21] Video Mixing Renderer Filter 9
Microsoft Corp.
2003-06-14
Appendix

User group evaluation

Technical setup

For DVD picture quality comparison, a Euroline 6620 DVD player was used. Both the DVD player and the prototype were then connected to a Sony KV-29X5E Trinitron TV using the same type of S-Video cable.

Realization

The purpose of the user group evaluation was to get user comments on both picture quality and GUI handling. The user group consisted of four persons. After the tests they were to give comments on both usability and picture quality. Only a basic orientation of the remote keys was given before the tests.

Each group member was given approx. five minutes to freely navigate the prototype software. After that a chosen section of the DVD title “Hälsoresan” (1999, AB Svensk Filmindustri) was played on both units, to evaluate picture quality. The menu system of the DVD player was also shown for quality comparison. The results were written down and compiled from oral comments given during the test. The written comments were given to the group member to make sure the transcription was correct.
## Results

<table>
<thead>
<tr>
<th></th>
<th>User 1</th>
<th>User 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General GUI</strong></td>
<td>Functional 3D layout in main menu. Easy to understand. The font size could be bigger.</td>
<td>The Icons could be more intuitive, but together with the text it works well.</td>
</tr>
<tr>
<td><strong>Audio player</strong></td>
<td>Slow remote handling. Expandable and collapsible tree is a good approach.</td>
<td>Jerky remote scrolling, and no scrollbar to indicate position in the list. Nice graphics.</td>
</tr>
<tr>
<td><strong>DVD player</strong></td>
<td>OSD indication of chapter and position would be appreciated.</td>
<td>The remote has limited DVD controls.</td>
</tr>
<tr>
<td><strong>Movie player</strong></td>
<td>Slow remote handling. Chapter selection is really good. Fast forward and rewind could be smoother. Lack of OSD movie position.</td>
<td>Like in the audio player, the remote scrolling is jerky. Nice chapter, preview and resize functions.</td>
</tr>
<tr>
<td><strong>Picture quality</strong></td>
<td>Not much difference. A bit crisper image on the Euroline.</td>
<td>The Euroline is crisper, but the prototype is more stable.</td>
</tr>
<tr>
<td><strong>User 3</strong></td>
<td>Nice GUI. 3D menus give it a unique touch.</td>
<td>No problem navigating the main menu. Nice graphics.</td>
</tr>
<tr>
<td><strong>Audio player</strong></td>
<td>A bit difficult to handle play list editing at first. Perhaps another larger font would be better. Scrolling will not stop when I want it to. Other then that it is quite easy to work with.</td>
<td>Works very well, except that the remote control commands are a bit slow.</td>
</tr>
<tr>
<td><strong>DVD player</strong></td>
<td>Navigation in the DVD menus works well. A few important remote commands missing. No playback time count.</td>
<td>No playback information. Limited remote control functions.</td>
</tr>
<tr>
<td><strong>Movie player</strong></td>
<td>Slow remote handling. Chapter selection is really good. Fast forward and rewind could be smoother. Lack of OSD movie position.</td>
<td>The same remote control problem as in the audio player. The index generation is a bit slow, but it is a really nice feature.</td>
</tr>
<tr>
<td><strong>Picture quality</strong></td>
<td>The Euroline picture is better, but the other one is not bad either.</td>
<td>Perhaps a bit softer image on the prototype. The Euroline unit sometimes has twitches in the picture, especially in the menus.</td>
</tr>
</tbody>
</table>