

Interface Design

LCD Touch Interface for ETRAX 100LX

By

Magnus Kling, Jerker Buud

SUBMITTED IN PARTIAL FULFILMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
AT
UMEÅ UNIVERSITY
UMEÅ, SWEDEN
SEPTEMBER 2003

Abstract

In this thesis a liquid-crystal-display (LCD) interface for the ETRAX 100LX processor is presented. The interface is made as an add-on circuit board to an AXIS 82 Developer Board. The LCD controller used on the add-on circuit board is the EPSON S1D13706 LCD controller. To have graphical support for the ETRAX 100LX the Linux framebuffer device was implemented for the CRIS Linux architecture. A touch screen is mounted on the LCD and controlled by a Texas Instrument ADS7846 A/D converter and Winbond 78E54 microprocessor. The touch point coordinates are sent to the AXIS 82 Developer Board through the serial port. An example application has been built to simulate a Document Server. The graphical user interface is built with EZFB API which uses the Linux framebuffer.

Sammanfattning

Rapporten tar upp hur ett liquid-crystal-display (LCD) gränssnitt har skapats till en ETRAX 100LX processor. Gränssnittet är tillverkat som ett externt kretskort som sammankopplas med ett AXIS 82 Developer Board. På det externa kretskortet finns en EPSON S1D13706 LCD kontrollert. För att få grafiskt stöd till ETRAX 100LX har Linux framebuffer enheten implementerats för CRIS Linux arkitekturen. En pekskärm var monterad på en LCD och pekskärmen kontrollerades med hjälp av en Texas Instrument ADS7846 A/D omvandlare och en Winbond 78E54 microprocessor. Peksärmskoordinaterna skickades till serieporten på AXIS 82 Developer Board. En exempelapplikation byggdes för att simulera en Dokument Server. Det grafiska användargränssnittet byggdes med EZFB API som utnyttjar Linux framebuffer.

ACKNOWLEDGEMENT

We would like to thank our two supervisors at Axis Communications, Bjarne Rosengren and Peter Johannezon for their support and valuable comments. We are grateful to the Hardware team who has been supporting us through the project and to all employees that have answered our different questions. We would also like to thank our supervisor at Umeå University, Ulf Brydsten.

Lund, Sweden
August 29, 2003

Jerker Buud, Magnus Kling

Contents

1	Introduction	1
1.1	Background	1
1.2	Motivation	2
1.3	Thesis Objectives	2
1.4	Thesis Outline	3
2	Theoretical background	5
2.1	LCD	5
2.1.1	Freedericksz Transition	6
2.1.2	Twisted Nematic Displays, TN	6
2.1.3	Super Twisted Nematic Displays, STN	8
2.1.4	Driving methods of LCD	8
2.1.5	Colour and monochrome Displays	9
2.2	Reflective, Transmissive, Transflective	10
2.3	OLED	11
2.4	The LCD module	12
2.5	LCD Controllers	13
2.5.1	Controlling the LCD panel	14
2.5.2	Techniques for colours/shades	15
2.6	Touch panel	15
2.6.1	Resistive 4-wire technology	16
2.6.2	Resistive 5-wire technology	16
2.6.3	Capacitive	17
2.6.4	IR	18
2.6.5	Surface Acoustic Wave	18
3	Software	19
3.1	Linux Framebuffer Device	19
3.1.1	Internal Structure	20
3.1.2	User access	20
3.2	Linux Virtual Consoles	20
3.2.1	Framebuffer Console	21
3.3	EZFB	22
3.4	Microwindows/Nano-X	22

4	Hardware Implementation	24
4.1	Prototype	24
4.2	Developer Board	24
4.3	LCD Module	25
4.3.1	LCD Module Interface	25
4.4	EPSON S1D13706 LCD Controller	26
4.4.1	S1D13706 Interface	27
4.5	Touch panel	30
4.5.1	ADS7846	30
4.5.2	Winbond 78E54	31
4.5.3	Serial Interface	32
4.6	Schematics and PCB layout	33
5	Software Implementation	34
5.1	Linux Kernel Version 2.4.20	34
5.1.1	Framebuffer Device	34
5.1.2	Epson S1D13706 Device Driver	36
5.1.3	Keyboard	37
5.2	Microwindows	37
5.3	Touch panel interface	38
5.4	Demonstration application	38
6	Results	40
6.1	Market Survey	40
6.2	Prototype	41
6.3	Performance	42
7	Touch LCD in Axis Products	43
7.1	Network Document Servers	43
7.2	Network Cameras	43
7.3	Video Servers	44
7.4	Network DVR	44
7.5	Developer Boards	44
7.6	New product: NetScreen	44
8	Discussion	45
A	Abbreviations and Acronyms	48
B	Market Survey	50
B.1	Acte	52
B.2	Egevo	52
B.3	Truly Semiconductors	53
B.4	TRG	54
B.5	Scancraft	54
B.6	Price examples	55

C	EPSON specification	56
C.1	Epson S1D13706: LCD support	56
C.2	Epson S1D13706: CPU Interface	56
C.3	Epson S1D13706: Display Modes	57
D	Measured interface timing	58
E	Bill of Materials	60
F	Circuit Diagram	63

List of Figures

2.1	90 degrees rotation of liquid crystal in TN LCD. Japanese Technology Evaluation Center.	7
2.2	Transmission (brightness) as a function of voltage. Japanese Technology Evaluation Center [17].	8
2.3	Direct vs multiplexed driving of LCD	9
2.4	Passive vs Active LCD design	10
2.5	Bias Voltage description	12
2.6	2x2 pixel dithering	15
2.7	4-wire resistive touch panel.	16
2.8	Capacitive touch panel.	17
2.9	Surface Acoustic Wave touch panel.	18
3.1	Linux framebuffer console	21
3.2	Nano-X implementation of Tetris.	23
4.1	Interface between EPSON S1D13706 and PMC39T01H	25
4.2	Generic bus interface	27
4.3	ETRAX 100LX Memory map	28
4.4	EPSON S1D13706 Generic 2 interface timing	29
4.5	Block diagram of touch construction	30
4.6	ADS7846 implementation	31
4.7	Serial Interface	33
5.1	The user is presented with this picture at the start up.	38
5.2	Picture, name and email address of one user when the upper arrow is pressed.	39
5.3	The scanned image is shown to the left and receivers email address is displayed on the top of the picture.	39
D.1	1=CS, 2=WAIT, 3=WE, 4=50 MHz Clock	58
D.2	1=CS, 2=WAIT, 3=RD, 4=50 MHz Clock	59
D.3	1=CS, 2=RD, 3=A0, 4=50 MHz Clock	59

Chapter 1

Introduction

1.1 Background

In 1984, Axis Communications AB was founded with a goal to become the leader in the IBM mainframe and midrange printer protocol converter market. They later widened their direction to become a developer for core technologies and products for network connectivity, including network video products, print, document and CD/DVD servers and wireless access points. To become the leader in network connectivity a well founded competence in ASIC design was built up.

During 1990-1993 Axis Communications focused on a new processor, ETRAX, that could be included in different embedded systems. ETRAX-1 (Ethernet, Token Ring, Axis -1) 32-bit RISC, Ethernet, Token Ring and I/O. Axis first network-ing system-on-a-chip solution was introduced on the market in 1993 and it was followed by three successors within the following three years; ETRAX-2 1994, ETRAX-3 1995 and ETRAX-4 1996.

In 1996 Axis Communications released the AXIS 200 Network Camera. A network camera built with the ETRAX-4.

In 1998 a new processor ETRAX 100 added 100 MBit Ethernet, ATA and Wide SCSI support and Token Ring support was removed. A version of the Linux operating system was used for the first time. This version depended on μ Clinux patches, which made it possible to run Linux on CPUs without a memory management unit (MMU).

In 2001 the processor ETRAX 100 got its successor; the ETRAX 100LX. ETRAX 100LX is a fully operating CPU with a MMU and can thereby run Linux 2.4 kernels without the need to rely on μ Clinux patches. Some of the specifications of the CPU are listed below:

- 32 bit RISC CPU core
- 10/100 MBit Ethernet controller

- 4 asynchronous serial ports
- 2 synchronous serial ports
- 2 USB ports
- 2 Parallel ports
- 4 ATA (IDE) ports
- 2 Narrow SCSI ports (or 1 Wide)
- Support for SDRAM, Flash, EEPROM, SRAM
- 100 MIPS

In the year 2002 Axis released ETRAX 100LX Multi Chip Module. Based on ETRAX 100LX, the MCM includes 2 Mbyte flash memory and 8 Mbyte RAM, making it essentially a Linux computer in a single 27x27mm chip.

1.2 Motivation

Axis Communications AB is now market leading in the network video market. To open new markets and continue to be in the frontier of the market, it would be an advantage if the ETRAX 100LX is capable of using a display to present information to an end user.

There is one product that incorporates an ETRAX and an LCD (Liquid Crystal Display) in Axis Communications product suit. The Network Document Server. The Network Document Server is based on the ETRAX 100 and is using a COG (Chip on Glass) display that has two rows with each row including 16 characters for displaying text. It can not show graphics, which is a disadvantage in todays consumer driven market.

On the Network Document Server there are a number of keypads for choosing name and document format. The keypads can be removed by applying a graphical LCD touchscreen instead of the 2x16 display that will take over the functionality from the keypads and the COG display. To connect a graphical LCD touchscreen to ETRAX 100LX an new interface must be implemented.

By making a graphical LCD touchscreen interface to the ETRAX, Axis Communications will be able to offer new products, which will help them to be in the cutting edge on the market. The potential for the OEM market is great with different network and storage products that can benefit from using a graphical touch LCD as presentation media.

1.3 Thesis Objectives

A prototype system shall be developed that illustrates the stated problem as well as the solution. Project cost optimisation will be in focus which limit the use of

advanced LCD and newer technologies such as OLED.

The thesis objectives are stated below:

- Circuit diagram design.
- Product optimisation.
- Prototyping of the solution.
- Application Programming.
- Software and design analysis.

To build a prototype, a product named AXIS 82 Developer Board will be used as platform. AXIS 82 Developer Board consist of an ETRAX 100LX MCM, which has 2 Mbyte flash memory and 8 Mbyte RAM. Upon that, AXIS 82 Developer Board has an additional 4 Mbyte flash memory and 8 Mbyte RAM, which concludes to a total of 6 Mbyte flash and 16 Mbyte RAM. The choice of platform is mainly based on the fact that the AXIS 82 Developer Board has a number of external connectors (Network interface, USB, system bus, serial ports) and that it runs Linux.

1.4 Thesis Outline

Chapter 2 Theoretical Background: An introduction to how a liquid crystal display is working, how LCDs can be controlled and how colour is being displayed on the screen. It also explains the theory behind touch panels and briefly about OLED.

Chapter 3 Software: Presents the used software in the project.

Chapter 4 Hardware Implementation: Explains how the hardware for the prototype was built and a description of the components used.

Chapter 5 Software Implementation: Presents the implementation of the software in the prototype.

Chapter 6 Results: Introduces the results of the thesis.

Chapter 7 Touch LCD in Axis Products: How could products from Axis Communication benefit from using a graphical LCD touchscreen?

Chapter 8 Discussion: Makes a quick brief over further improvements to the prototype and our own reflections.

Appendix A Abbreviations and Acronyms: Short explanation on concepts that occur in the report.

Appendix B Market Survey: Market survey of LCDs and what five different supplier could offer.

Appendix C EPSON S1D13706 specification: Specification of the LCD controller.

Appendix D Measured interface timing: Shows pictures of the captured timings of the interface.

Appendix E Bill of Materials: List of all components used in the prototype.

Appendix F Circuit Diagram: Pictures of the circuit diagram for the prototype.

Chapter 2

Theoretical background

This chapter provides the reader with basic theory regarding liquid crystal displays, touch panels, OLED and basic colour algorithms on an LCD. It also includes a section that explains how an LCD subpixel get addressed.

2.1 LCD

LCD stands for Liquid Crystal Display and is used in many different applications such as calculators, watches and computer screens. LCDs are common, since they offer real advantages over other technologies; they are lighter, thinner and usually consumes less power than a cathode ray tube (CRT).

Most of us think of crystals as rock like substances. In 1888 an Austrian botanist Friedrich Reinitzer [18] observed that when he melted cholesteryl benzoate it first became a liquid that was murky, but cleared up when its temperature rose. Upon cooling, the liquid turned murky before it crystallised. The cholesteryl benzoate had two melting points. When the cholesteryl benzoate is murky it is between a solid phase and a liquid phase. This phase is named liquid crystalline state.

It is well known that a material can be solid, liquid or gaseous, which means that the molecules are either stationary or moving. Liquid crystal can behave both as a liquid and a solid material. A fair amount of heat is needed for a material, in liquid crystalline state, to go to a liquid phase. This explains why liquid crystals are very sensitive to temperature changes, and why a notebook display can act strange during a hot or cold day.

Most liquid crystal molecules are categorised as either thermotropic or lyotropic. Thermotropic molecules react on temperature and pressure. The reaction of lyotropic liquid crystals, which are used in the manufacture of soaps and special LCDs, depends on the type of solvent they are mixed with. In the lyotropic phase the molecules tend to align themselves in planes or layers. The molecules can move within the plane, and the plane can move around other planes. But the molecules

can not move from one plane to another.

The thermotropic liquid crystals are either isotropic (random arrangement) or nematic (definite order of pattern). The most commonly used molecules in LCD are of type nematic.

The distinguishing characteristic of the liquid crystalline state is the tendency of the molecules to point along a common axis, called the director. This response of the liquid crystals to an electric field is utilised in industrial applications.

If an electric field is applied along the director, the liquid crystal molecules, which are dipoles, tries to align themselves in the direction of the electric field.

If there is no external field, the director is free to point in any direction. There is a solution to get the director to point in a special way, without applying external fields.

If a polymer is spread evenly on a piece of glass and then rubbed with a cloth in one direction, creating small microscopic grooves in the surface of the polymer, the molecules will align themselves in the direction of the rubbing motion.

2.1.1 Freedericksz Transition

If the molecules of a liquid crystal is anchored to the surface between two polymer coated glass, parallel with the glass, and an electronic field is applied perpendicular to the molecules they will start to rotate to align with the electronic field. This phenomena is called Freedericksz Transition [23]. Freedericksz Transition is used to make LCDs and without the understanding of this phenomena, one can not make an LCD.

2.1.2 Twisted Nematic Displays, TN

To understand how a display works some theory about polarisation is needed. An optimal screen should be able to let light pass through a certain pixel in the ON state and block light in the OFF state. To make this happen, one could use a theory of polarisation. A non-polarised light have no common direction of the light beams. They are emitted from the center of the source and spread in every direction. Linear polarisers absorb (or block) light that is not polarised exactly in their orientation and pass all light that is in their orientation. When two polarisers are put parallel, in the same orientation and on top of each other they pass on all light that is in their orientation. If one of the filters is rotated 90 degrees, it will block the light that is polarised and coming from the other filter. Rotated ones more 90 degrees, it will let the light pass through. Considering this, the light output would be proportional to the mechanical rotation of the second polariser. Due to mechanical difficulties, this is not a practical solution when making an LCD. Instead, twisted nematic molecules are used in order to make the light follow the director and get through the filters.

A display is built up by two glass plates, which has polarisers attached to the outside of the glass. On the inside the glass plates have an electrode plane made of indium-tin oxide. The electrode plane covers the entire area of the display. The electrode cover is then coated with a polymer and rubbed with a cloth in the direction of the polarised film, creating small grooves in the polymer. The glass plates, with the electrode plane and the polymer coating, are then rotated 90 degrees from each other, which makes the light totally blocked between the two plates. Between the plates, liquid crystals are inserted and the package is then sealed. The liquid crystals will then align themselves in the direction of the grooves in the polymer. This will make the molecules twist 90 degrees and the light will follow the, now, twisted molecules and thereby go through the second polariser. If an electric field is sent from the bottom plate to the top plate, the molecules will untwist and make the light travel within the crystals, without rotation. That will cause the, already polarised, light to hit the second polariser, which will block the light and the screen will appear dark. The behaviour is for a positive LCD type. A negative LCD type behaves the other way around, meaning that the screen is dark when no electric field is applied and if an electric field is applied, the molecules will twist and light will go through. See figure 2.1 for an image of TN layout in positive mode.

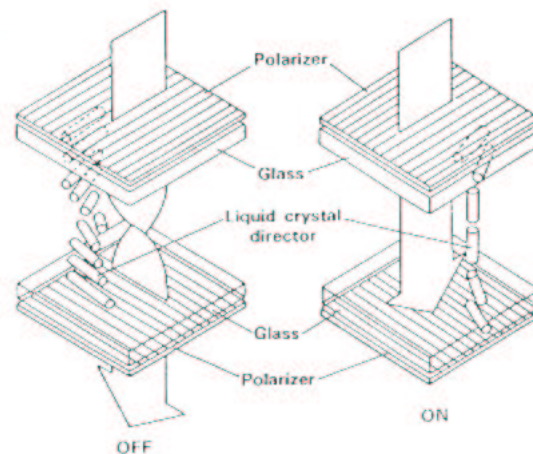


Figure 2.1: 90 degrees rotation of liquid crystal in TN LCD. Japanese Technology Evaluation Center.

The transmission of the LCD as a function of applied voltage is shown in figure 2.2 on page 8. There is a threshold behaviour for most LCDs and no change in transmission occurs until a threshold voltage, V_{th} , is reached. Transmission then decreases as the voltage increases until saturation is reached. Threshold voltage is typically 1.5-2.5 volt, and saturation occurs at about 4-5 volts.

For direct-drive LCDs, which are used in simple indicators, high contrast can be achieved by driving the liquid crystal into saturation. Contrast ratios in excess of 100:1 can be achieved in this mode. To address multiple lines, as is typical in computer or TV screens, multiplexed addressing is used. Information is applied to column electrodes one row at a time. The number of rows that can be multiplexed depends on the steepness of the transfer characteristic. The ratio of the voltage in

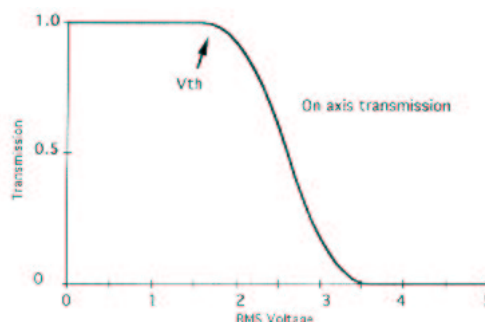


Figure 2.2: Transmission (brightness) as a function of voltage. Japanese Technology Evaluation Center [17].

the selected state, V_s , and the non selected state, V_{ns} , is given by the formula:

$$\frac{V_s}{V_{ns}} = \sqrt{\frac{\sqrt{N} + 1}{\sqrt{N} - 1}}$$

where N is the number of rows multiplexed. For example, if N is 200, the difference between on and off states is only 7%. To achieve reasonable contrast, a very steep electro-optic transfer characteristic is required. The limit for TN LCDs is about 64:1 multiplexing. After that the contrast ratio drops rapidly [17].

2.1.3 Super Twisted Nematic Displays, STN

Super Twisted Nematic Displays has the same function as the Twisted Nematic Display but is even more twisted. The twist extends to 270 degrees, which makes it superior to the normal Twisted Nematic Display when comparing the contrast ratio, since the 270 degrees twist blocks a lot more light than the 90 degrees twist. Due to the large twist angle, the actual alignment of the polarisers for STN LCDs is not perpendicular, but it is adjusted to find the best direction for optimum display characteristics. The STN material is rotated in a way so the change from transmission to totally dispersion is very abrupt and therefore it can respond quickly to small changes in voltage. The technology is widely spread and more and more companies are using it in order to build cheap LCDs that have good contrast.

2.1.4 Driving methods of LCD

There are two different methods to produce an image on an LCD. Segmented driving method, which creates images from segments, and the matrix driving method, which creates images with dots. The simplest of the two is the segmented driving. The segmented driving method produces characters and pictures with cells defined by patterned electrodes. By hardwiring a segment of a character or image, it is

very easy to address each segment that forms the image if there are a small number of segments. If the picture has more pixels, the wiring becomes complexed.

An alternative method of driving the LCD is by using the multiplex drive method, in which the pixels are arranged and wired in a matrix format. This method involves driving several pixels at the same time by time division. Each row gets a specific amount of time and during this interval the column driver lights the selected column segment. A current goes from the column segment to the intersected row driver and the selected pixel is lit.

The figure 2.3 shows the differens between direct driving and matrix driving methods.

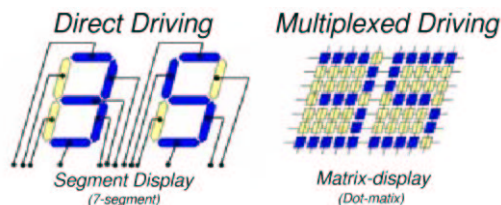


Figure 2.3: Direct vs multiplexed driving of LCD

Matrix driving methods can be divided in two categories, passive and active matrix. In the passive mode the pixel gets litted by applying a voltage between the intersected row and column. The pixel must be litted for a specific time to allow the liquid crystal to twist so the light can be blocked. Due to the significant voltage which is applied over the liquid crystal during a long time, crosstalk between lines can occur and this can create blurred images.

In active-matrix LCD, also known as TFT (Thin Film Transistor) LCD, a switching device (transistor) and a storage capacitor are integrated at each cross point of the electrodes. This generates less crosstalk and sharper images. On the other hand a switching device and a capacitor take up a certain amount of space and if a screen has a resolution of 320x240 pixels there needs to be $320 \times 240 = 76800$ capacitors and 76800 switching devices. A total of 153600 units are attached to the screen. If one of them fails there will be a black or a white pixel on that screen. This is one of the biggest disadvantages of the active matrix construction. See figure 2.4 on page 10, for a visualisation of the differens between passive and active LCDs. The cons are that they have a faster response time, meaning that the time from a non-selected state to a selected state and back to the non-selected state is smaller than for a passive matrix LCD. This produces higher quality in the pictures on the active matrix LCD.

2.1.5 Colour and monochrome Displays

All the methods described until now are for monochrome displays. If the voltage goes low, the crystal is twisted, creating a white pixel. If the voltage goes high,

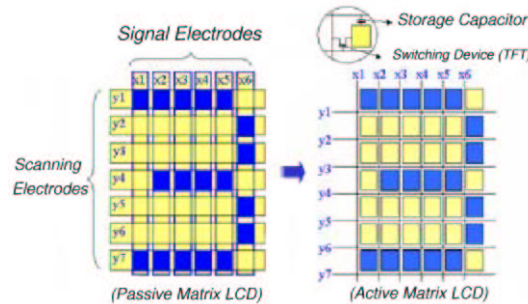


Figure 2.4: Passive vs Active LCD design

the crystal untwist, creating a black pixel. To make a colour display, three different colour filters must be added side by side in each pixel. The colours of the filters are red, green and blue. Often referred to as RGB colours. The three filters in each pixel are called subpixels and must be addressed the same way as in a monochrome display. This creates three times more electrodes in passive LCDs and three times the total amount of units (transistors and capacitors) in active matrix LCDs.

For example, if a red pixel should be created, the voltage goes high on the green and on the blue subpixel, letting only light pass through the red filter. With three colour filters that can be either on or off eight colours can be created; black, white, green, blue, yellow, magenta and cyan.

To make additional colours and to create a grey scale other methods must be presented. This is discussed in section 2.5.2.

2.2 Reflective, Transmissive, Transflective

All LCDs need some sort of light source to show the pixels on the screen. There exist three light transmission modes for the LCDs.

The cheapest and most commonly used mode is reflective mode. It can be found in phones, watches, displays and many other consumer electronics. It works by having ambient light reflected in the display. This is achieved by combining a reflector with the rear polariser. This viewing mode works best in well lit office or outdoors.

Transmissive mode works by having an external light source mounted on the back of the LCD, for example a cold-cathode-fluorescent-light (CCFL). The light from the CCFL goes through a diffuser, that spreads the light evenly on the back of the LCD and onto the transparent rear polariser. This construction is used in applications where the best quality is a critical factor and where there is no or little ambient light. The technique is commonly used in computer screens and LCD TV.

In very bright light, as in direct sunlight, the transmissive mode can make the text be "washed out". "Washed out" means that the text on the LCD becomes almost

white and the colours fades away.

Transflective mode is combining transmissive and reflective modes. By using both a reflective rear polariser and a backlight, the LCD can operate in most situations. In direct sunlight the CCFL can be switch off, saving power, and switched on again when needed. The transflective mode also reduces the "washed out" effect that transmissive LCDs have in bright light.

To use a CCFL, an inverter connected to the CCFL is needed. The inverter will have a small voltage, around $+3V \sim +5V$, as input and then the voltage is increased by the inverter to around $400V$. That ignites the CCFL and makes the gases inside the CCFL glow.

Instead of using a CCFL, LEDs can be used. They give not as bright light as the CCFL, but draws less power and an inverter is not needed.

2.3 OLED

Organic Light Emitting Diode (OLED) is a new technology that can be used in displays. The technique differ from the LCD technology by not needing a light source and by not having a liquid crystal between two glass substrate. Unlike traditional LCDs, OLEDs are self-luminous and do not require back lighting. This eliminates the need for bulky CCFL and yields a thinner, more compact display.

The basic OLED cell structure consists of thin organic layers, sandwiched between a transparent anode and a metallic cathode. When a voltage is applied to the cell, the emitting layer produces light (electroluminescence). This will give a bright green light that can be compared to old monochrome displays colour range. To apply the voltage, the same technology used in LCDs can be used in OLED technology. As described above, in section 2.1.4, there are passive and active techniques for addressing the right cell. Kodak [20] is the leading manufacturer for OLED and have several patents. They where the first to make colour OLEDs and is driving the research further than any other company.

The disadvantage of OLED is that the organic material will decrease its luminescence by 50 % [19] after 3.4 years of normal usage (based on eight hour usage per day). Because no backlight is needed they have a low power consumption, which provides for maximum efficiency and helps minimise heat and electric interference in electronic devices.

2.4 The LCD module

To control an active or passive dot-matrix LCD panel, external components are needed. The LCD module usually consist of an LCD panel, column driver, row driver and a bias voltage circuit. For a transmissive LCD panel the module can include a light source (backlight) to show the pixels on the screen.

The panel is the visual part of the LCD module, which contains the pixels, as explained in section 2.1.4. The panel is driven by the column and row driver to address the specific subpixel. The bias voltage circuit supplies the drivers with the power needed to rotate the liquid crystals.

The bias voltage levels are decided by an internal bias voltage circuit. Normally, the internal bias voltage circuit (VEE) is supplied from an external power source. This voltage is spread on four signals typically called V1~V4. GND is V5 and VEE is V0. See figure 2.5 The levels for the signals follows this equation:

$$|V0-V1| = |V1-V2| = |V3-V4| = |V4-V5|$$

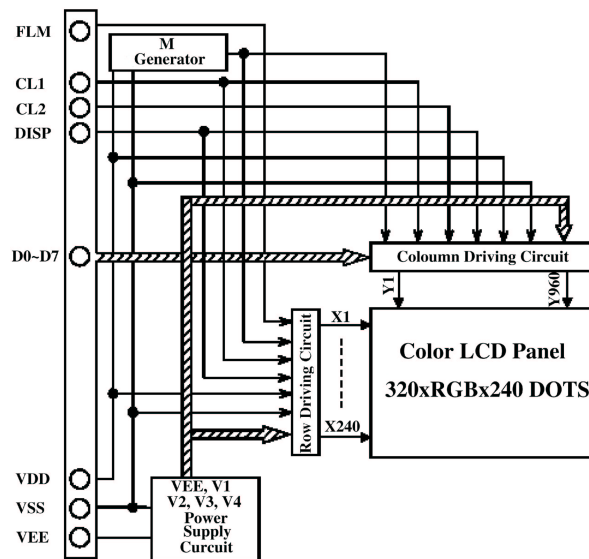


Figure 2.5: Bias Voltage description

For example:

$$V0 = 25.00V, V1 = 23.44V$$

$$V2 = 21.87V, V3 = 3.13V$$

$$V4 = 1.56V, V5 = 0V$$

This gives:

$$|V0-V1| = 1.56V, |V1-V2| = 1.57V = |V3-V4| = 1.57V = |V4-V5| = 1.56V.$$

V0 and V5 goes to both the column driver and row driver ICs of the LCD. V1

and V4 goes to the row driver IC. V2 and V3 goes to the column driver IC. At normal operation on a positive LCD, with no data displayed, the voltage over a single pixel is $1.57V \sim 1.56V$. This means that it is less or equal to the V_{th} and light will pass through. By adjusting the VEE, the differences in voltage between V1,V2 and V2,V4 can be extended, thus going over the V_{th} level and the LCD starts to get dark. This adjusts the contrast. The curve in figure 2.2 shifts to the left as temperature increases, and the VEE must be decreased in order to maintain the same visibility.

2.5 LCD Controllers

The LCD controller can be integrated in the LCD module, but in most case it is an external component. There exists a number of different LCD controllers and they have all different interface timing, LCD support and functionality. The purpose of the LCD controller is to control the driver ICs named row and column driver in the LCD module, which displays a picture on the LCD.

The method to address the subpixels are mentioned in section 2.5.1. But how do the controller actually make a picture on the LCD? One example with a 3.9" passive 320x240 colour LCD PMC39TO1H from Palm Technology [29] will be explained below, but the same method can be applied on all LCDs. The differences between LCDs are often how many data signals there are and how the LCD is built. A controller that supports the LCD module and the CPU interface must be chosen according to the manufacturer's specification.

The control signals, that the controller sends to the LCDs row and column driver, are often called "Data Shift Clock", "Input Data Latch signal" and "Scan start up". The LCD controllers often have an internal memory which is called video ram (VRAM). The CPU puts the data, that should be displayed on the LCD, in the VRAM of the controller. The screen image must then be re-displayed over 150 times per second (frames) by the LCD controller. This puts the workload of sending data to the LCD module on the controller and the CPU can handle other important functions.

Large manufactures of LCD controllers are EPSON [13] and Asilant [4]. Epson controllers are mostly used in embedded systems and Asilants controllers are used in desktop computers.

An LCD controller can be built with an FPGA, but there are drawbacks in the development of the FPGA code; an LCD module can easily be destroyed if the liquid crystal drive voltage is allowed to stay with the same polarity too long. The polarity must be reversed between 300 and 800 times per second. The development must therefore use a simulator that acts as an LCD module.

There exist different LCD controllers built using an FPGA. Examples are Triscend [38] and their series of microprocessors with a built in FPGA. On OpenCore [15], a project have built an LCD/VGA controller with an FPGA.

2.5.1 Controlling the LCD panel

A pixel is typically a square dot of colour, but it actually consists of red, blue and green subpixels. These subpixels are arranged side by side within a pixel. The pixel layout of the LCD is stored in a VRAM that is accessed by the LCD controller. The amount of VRAM depends on the amount of bit/pixel. For example an LCD with 320x240 pixels to display a image with a resolution of 16-bit/pixel needs an 150Kbyte large VRAM.

In this example, with the PMC39T01H, the screen format is described as 320x240 pixels, meaning 320 pixels across and 240 rows, and system software and LCD controllers would be configured for 320x240. At the interface level, each row requires 960 bits if a colour LCD module is used, for monochrome displays, 320 bits is required.

The PMC39T01H has three different control signals and eight data signals. The control signals are: CL1, which is the “Input Data Latch signal” and causes the data to be latched into place and displayed on the next row, CL2, which is the “Data Shift Clock” makes the data signal address eight bits more and FLM, which is the “Scan start up signal”. CL1, CL2 and data signals go to the column driver. FLM and CL1 goes to the row driver.

The actual placement of data to the pixels of the LCD panel is made by the column and row driver of the LCD module. Data is sent to the LCDs colour driver one row at a time, followed by a pulse on the CL1 signal, which causes the data to be latched into place and displayed on the next row. When the FLM signal, which goes to the row driver, is asserted high, CL1 causes the data to be displayed on the top row. The latch action happens when CL1 goes from high to low.

The data lines are named D0~D7 and data is placed on these lines and clocked in by a raising pulse on the CL2 line. For 320 pixels, there will be $320 \times 3 = 960$ bits per row. With 8 data lines there will be $960/8 = 120$ clock pulses on CL2 per row. Bits are placed on the data lines as table 2.1 shows.

CL2 Pulse =>	1	2	3	—	120
Signal					
D7	R0	B2	G5		G317
D6	G0	R3	B5		B317
D5	B0	G3	R6		R318
D4	R1	B3	G6		G318
D3	G1	R4	B6		B318
D2	B1	G4	R7		R319
D1	R2	B4	G7		G319
D0	G2	R5	B7		B319

Table 2.1: Addressing sub pixels in the PMC39T01G LCD module. R, G, B are abbreviations for Red, Green, Blue followed by the pixel number.

2.5.2 Techniques for colours/shades

The number of colours that can be displayed on a colour STN or TFT display is eight, since the LCD only has 3 bits per pixel (red, green and blue). But since eight (2^3) colours/pixel does not generate good looking pictures, further techniques must be applied in the LCD controller in order to display more colours or shades. By applying these techniques the eye can be manipulated to believe that there are more colours than there really are. Two different methods [33] that can be used are presented below:

Dithering is a method to extend the number of visual shades that can be seen on an LCD by using a group of pixels. Within this group the different pixels can be on vs. off according to the shade that the group should represent. The number of shades that can be displayed depends on the size of the pixel group. When the group gets larger the graininess increases and therefore the group size is kept to a minimum, typically 2 x 2 pixels. The figure 2.6 illustrates how a 2x2 pixel group can be dithered to 5 shades. Dithering can be used in both colour and monochrome LCDs.

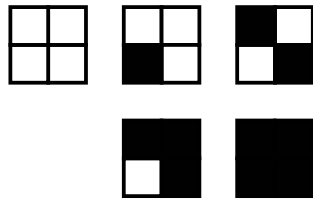


Figure 2.6: 2x2 pixel dithering

Frame Rate Modulation (FRM) is a technique for modulating a pixel's on vs off time over multiple frames, typical 16 frames. The advantage of using this technique is that it does not generate graininess like the dithering technique. The drawback of FRM is that the more shades generated, the longer the modulation period. When the modulation period increases the number of frames needed for each visible frame increases. This can make the display to flicker or roll when using fluorescent backlight [28]. This technique works better with passive displays (STN) than with active displays (TFT), because they are slower and will average out the visual effects when switching on and off pixels rapidly.

Frame rate modulation and dithering can be used together to extend the colour palette without pushing either technique beyond its limits.

2.6 Touch panel

A touch panel is an interface between a user and a device. Touch interface have been used in held personal digital assistant (PDA) for many years and more appli-

cations are developed; ATMs, mobile phones, ticket machines and other machines where the user should interact in a simple way with an application. The industry can benefit from using a touch screen when a mouse and keyboard can not be used due to dirt or other

There are a number of different techniques for making a touchscreen; resistive, capacitive, IR and by sound waves. All of them have both advantages and drawbacks. A quick explanation of each technique is presented below.

2.6.1 Resistive 4-wire technology

A 4-wire resistive touch screen consists of three layers [34]. Two layers that consist of a flexible membrane with a transparent metal oxide coating. Between the two membrane layers a third layer, which consist of a grid of spacers, is placed in order to avoid contact between the two membranes. The metal oxide coating and the spacers may reduce the picture quality and brightness of the screen. The key advantage of resistive touch technology is the wide range of pointing devices e.g. a gloved finger, finger nails, credit card pens, etc. that can be used. The two metal oxide coated membranes are referred to as x and y planes. See figure 2.7.

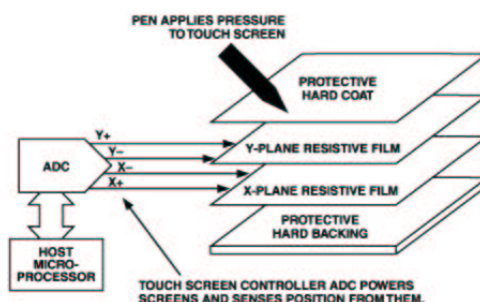


Figure 2.7: 4-wire resistive touch panel.

To detect a touch point on the x plane, the touch controller sends a small voltage over the x plane and makes a read out on the y plane, which acts as a probe. This value is the X position and then the same step is repeated but with the voltage on the y plane and the read out on the x plane. The values are converted with an A/D converter and passed on to a microprocessor for further calculations. The 4-wire technology is the most used technology for developing a touch panel.

2.6.2 Resistive 5-wire technology

5-wire technology [34] is built up by connecting four wires to the corners of a conductor (bottom plane) made of indium-tin oxide. A conductive plane (probe) is placed on top of the bottom plane and spacers are added between them. By applying voltage to the four corners of the bottom plane, and measure the vertical and

horizontal resistive network with the probe, X and Y coordinates can be calculated. For example, the Y coordinate is determined by connecting the upper left and right corner to VCC and connecting the lower left and right corner to GND. When a touch point is made the probe measure the voltage at the touch point and the A/D converts it and passes the digital signals to the microprocessor. The voltage measured is determined by the voltage divider developed at the touch point. The 5-wire technology is more accurate than 4-wire technology, but more expensive.

2.6.3 Capacitive

A capacitive system consists of an electrical charged layer, placed on a glass screen. See figure 2.8. When the user touches the monitor with his finger, some of the charge is transferred to the user, so the charge on the capacitive layer decreases. In the four corners of the monitor, circuits are located and the decrease is detected by these circuits. A touch point is calculated from the relative differences in charge at each corner where the touch event took place.

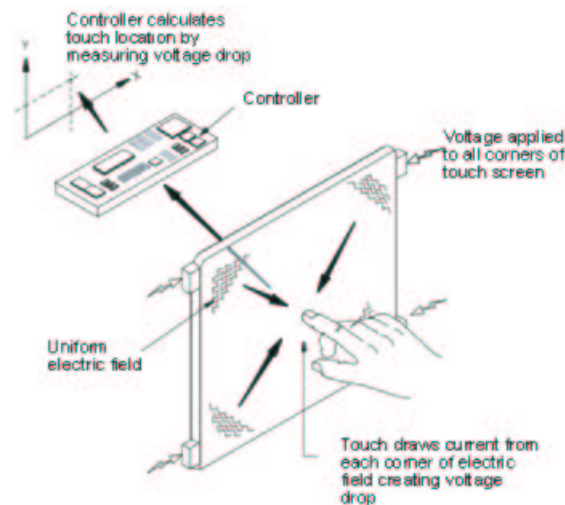


Figure 2.8: Capacitive touch panel.

The great advantage with this technique is that it transmit almost 90 % of the light from the monitor, whereas a resistive system only transmit about 75 %. Another effect is that a capacitive system is more accurate than the resistive system.

A negative effect with this type of system is that it will not work if the pointing device is not conductive. Another disadvantage is that it is more expensive than resistive touch screens.

2.6.4 IR

By using infrared (IR) light a touch on a screen can be detected. A grid of IR beams is transmitted over the screen and by picking up the disturbances in the grid a touch point can be calculated. The drawback with this method is that a touch can be recognised before a touch is physically made on the surface. Dirt can also disturb the IR light, causing false touch points.

2.6.5 Surface Acoustic Wave

Surface Acoustic Wave (SAW) systems have all a glass overlay with transmitting and receiving piezoelectric transducers for both the x and y axes. See figure 2.9. The touchscreen controller sends a signal to the transmitting transducer, which converts the signal into waves within the glass. These waves are directed across the front surface of the screen by an array of reflectors. Reflectors on the opposite side gather and direct the waves to the receiving transducer, which transforms the wave back to an electric signal.

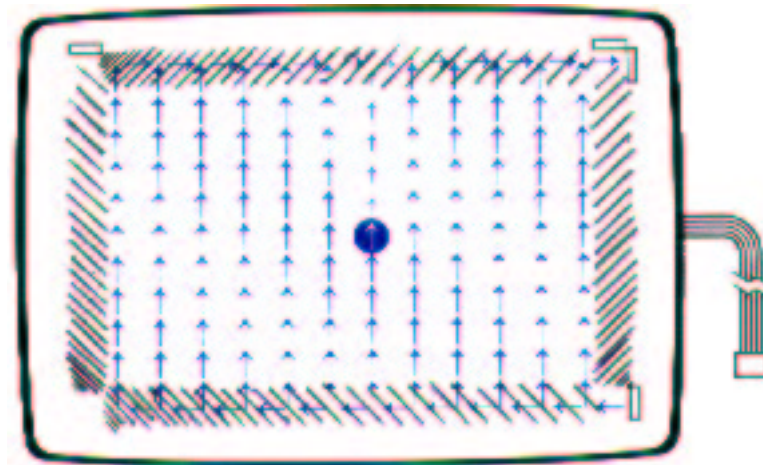


Figure 2.9: Surface Acoustic Wave touch panel.

When one touches the screen, a small portion of the wave absorbs into the pointing device. This disturbance is collected and compared to the stored digital map over the glass surface. This gives the x and y coordinate of the touch point.

The advantages of using SAW systems are that they are very accurate and one do not need to have special pointing devices. The disadvantage is that they must be clean to function correctly.

Chapter 3

Software

This chapter provides the reader with information about the software used by the prototype. After reading the chapter the reader will be familiar with Linux framebuffer, Linux Virtual Console and some framebuffer applications.

In AXIS Document Servers a realtime kernel software called OSYS is used. The operating system on the prototype, and through this thesis, will be the Linux. Linux is used more and more at Axis Communications and the prototype platform, the AXIS 82 Developer Board, uses Linux as its native operating system. The fact that the writers of this thesis are familiar with Linux and not OSYS have also been considered when choosing operating system.

3.1 Linux Framebuffer Device

The Linux framebuffer device is an abstraction for the graphics hardware. It was introduced in the Linux kernel version 0.9.x (m86k) and was supported by i386 architecture in version 2.1.107. Since the release of the 2.2 Linux kernel, the Linux framebuffer device has been the most commonly used console driver for non i386 architectures.

The reason that the Linux framebuffer device was introduced in the Linux kernel, was to have support for the Linux virtual console layer for m68k. The m68k architecture uses a graphical console and not a text console as the i386 architecture. In the early m68k version of the Linux kernel the Linux framebuffer device was built into the virtual console layer. This made further developing of the virtual console layer difficult since it was hardware dependent. The graphical hardware specific code was then moved out from the virtual console layer and the Linux framebuffer device was born. The Linux framebuffer device is designed to be platform independent and support many different graphical devices and architectures.

3.1.1 Internal Structure

The Linux framebuffer device store all information that can be used by the system or the software in different structures. These structures are located in the file *linux/include/linux/fb.h*. A short explanation of the main structures follows:

fb_var_screeninfo

holds the information that normally can be changed by the user, e.g resolution and colour depth.

fb_fix_screeninfo

contains the non changeable properties of a graphics card, e.g the physical address to the video memory.

fb_info

information about the current hardware independent state of the framebuffer device is stored here.

fb_monospecs

holds the information about the monitor display and its supported video modes.

3.1.2 User access

To access the Linux framebuffer device the user can use the specific device nodes, which are located at */dev/fb[0-7]*. These nodes act like a normal memory device, and any written data will be drawn by the graphical device to the screen. The normal way to access the framebuffer is to map the video memory, using the *mmap* system call. This will return an address that represents the physical address for the video memory in the graphical hardware.

To change different video modes, resolution etc. different *ioctl* calls can be used.

3.2 Linux Virtual Consoles

Linux virtual consoles (also called virtual terminals) are full screen terminal displays on the system video monitor. They are named virtual since there can exist more Linux virtual consoles than the number of physical displays connected to the computer. The virtual console layer in the Linux kernel must have some kind of keyboard input device to be functional for the user.

Virtual consoles are named */dev/tty**, with numbering starting at */dev/tty1* (*/dev/tty0* is the current virtual console). TTY devices, are used to control the full screen terminal display for a session like the virtual console, serial ports and the pseudo terminal. To access the full screen terminal on a system that does not have support for the Linux framebuffer console device, the */dev/tty0* device should be used. Systems with support for the Linux framebuffer device console should instead use the

`/dev/fb*` nodes, to get the terminal contents.

The console device, `/dev/console`, is the device that system messages should be sent to, and on which logins should be permitted in single user mode. This device is since the kernel version 2.1.71 managed by the kernel. For older versions it should be a symbolic link to either `/dev/tty0`, a specific virtual console, or to a serial port [2].

The Linux virtual console layer starts with an abstract console driver as shown in figure 3.1 The abstract console driver was developed by the early m64k Linux versions and was integrated with the original Linux console driver to support the Linux framebuffer console. The different console drivers can be changed during runtime. The console device is usually integrated with some kind of device that can be a kernel module, like the Linux framebuffer device drivers. When the module has been initialised and loaded into the kernel, it can make a console switch to the newly installed module and use its console driver. To make a console switch the driver calls the function `take_over_console`.

The Linux kernel needs a console at startup, and for system that has its console integrated in kernel module devices, a dummy console is used at startup.

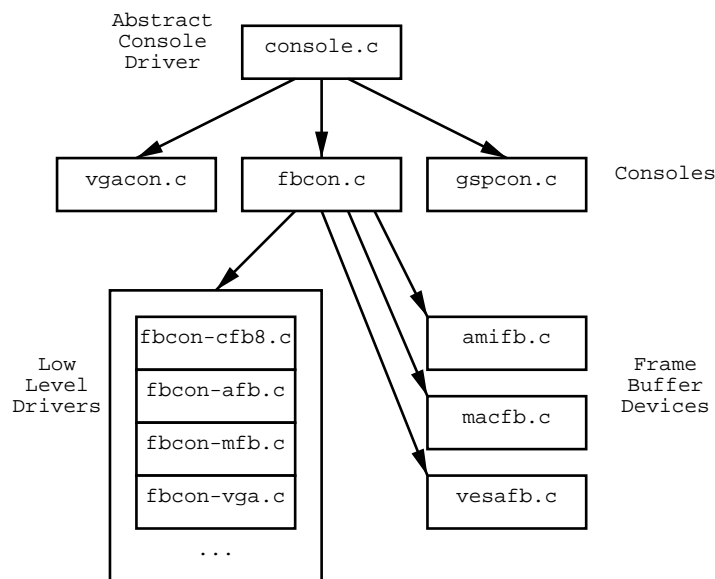


Figure 3.1: Linux framebuffer console

3.2.1 Framebuffer Console

The Linux framebuffer console device is designed to work with the Linux virtual console layer and the Linux framebuffer device. Its main purpose is to handle the standard console functionality that is defined in the `consw` struct located at `linux/include/linux/console.h`. These functions are for example the `con_putc` that handle all character output to the screen, and `con_cursor` that draws the cursor.

The interaction between the Linux framebuffer device, and the low level console drivers, are shown in figure 3.1 on page 21.

The character font layout is not located in the hardware for graphical consoles. Each character must exist in software, and be drawn pixel by pixel to the console when selected. This will make the framebuffer console slower than the VGA-text consoles, that uses a font table stored in the video hardware memory.

3.3 EZFB

EZFB [22] is a free Linux framebuffer API written in C by James Lehman. EZFB will run with any properly configured Linux framebuffer device. EZFB includes the ability to display all or portions of bitmap files anywhere on the screen, automatic colour depth conversion, the ability to capture to bitmap files, an 8x8 bitmapped font, routines to draw points, lines, outlined and filled rectangles.

EZFB is bundled with several applications that use the API:

- `./show [path/bitmap]` Display the specified bitmap in the center of the screen.
- `./tell` Show the current framebuffer configure.
- `./grab [path/bitmap]` Make a framebuffer screenshot.
- `./j1` The Screen Shredder demonstration.
- `./demo` Shows what your framebuffer can do.
- `./tsc` Touch screen calibration program.

EZFB is very easy to use and has the most common functions that one can expect from an API. The `grab` program was very useful when the prototype was made since the program takes memory dumps from the framebuffer device video memory to a BMP file. This gave a preview of what the screen should display, without needing an LCD connected to the AXIS 82 Developer Board.

3.4 Microwindows/Nano-X

The graphical user interface (GUI) has four essential elements; Windows, Icons, Menus and Pointing device (WIMP). The GUI software can be implemented in either user-level code, as is done in UNIX systems, or in the operating system itself, as in Microsoft Windows.

Microwindows [8] is a project that includes both a Win32/WinCE API and a Xlib-like API called Nano-X. It is a free open source project aimed to have the features

of a modern graphical windowing environment to smaller devices and platforms. Microwindows can be configured to work with the Linux framebuffer device. It is small and effective, as wanted in embedded systems and singelboard computers with reduced hardware resources. It can easily be crossed compiled for different architectures.

Microwindows has support for input devices such as mice, keyboards and touch panels.

Nano-X was designed for a client/server environment, as no pointers to structures are passed to the API routines. Instead, a call is made to the server to get an ID, which is passed to the API functions and is used to reference the data on the server. In addition, Nano-X is not message-oriented, instead modelled after the network client/server X protocol.

Some example applications are supplied to show the ability of Microwindows. For example there is a window manager, a tetris game and a terminal application. See figure 3.2 for a screenshot of Nano-X Tetris.

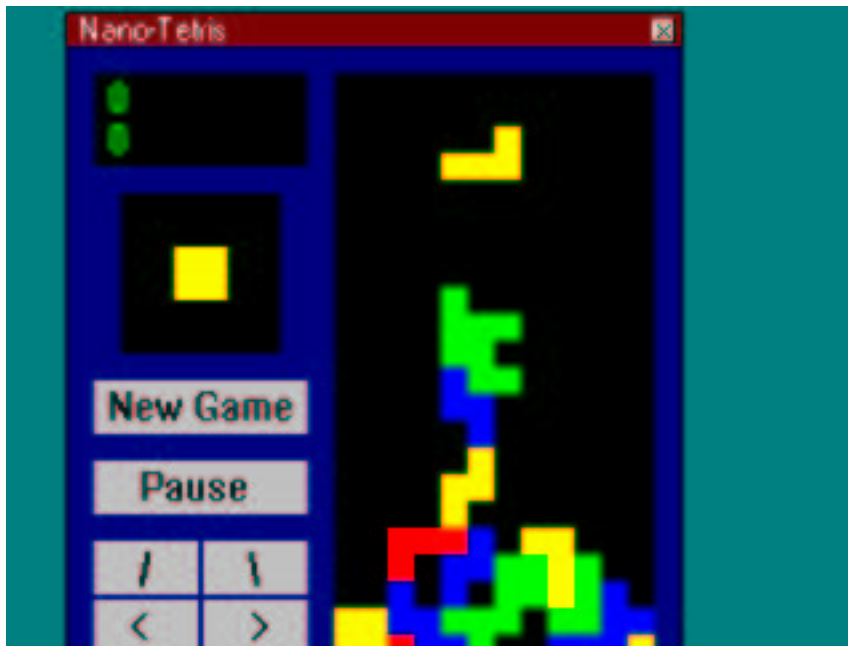


Figure 3.2: Nano-X implementation of Tetris.

Chapter 4

Hardware Implementation

The chapter is a description of how the prototype hardware where constructed. It will give a full explanation of the interface between AXIS ETRAX 100LX MCM, EPSON S1D13706 LCD controller, Palm Technology PMC39T01H LCD and how a touch point is recognised with the use of a Texas Instrument ADS7846 and a Winbond 78E54.

4.1 Prototype

The hardware implementation, described in this chapter, consist of a graphical LCD with a touch screen. The hardware implementation is interfaced to the AXIS 82 Developer Board as an add-on circuit board. The prototype tries to simulate the AXIS Document Server. The AXIS Document Server used today uses a character LCD which can display 2x16 characters. A keypad on the AXIS Document Server is used as user interface, but for this prototype the touch screen is used instead. The prototype is used to show how a graphical LCD with touch screen interface, can be used in future Axis products.

4.2 Developer Board

The AXIS 82 Developer Board [6] was chosen as the prototype platform since it is the most economical solution for this project. The reason that the AXIS 82 Developer Board was used as the prototype platform for this implementation where that the system bus is available on the developer board as a connector. The AXIS 82 Developer Board includes the ETRAX 100LX MCM with additional 4Mbyte flash and 8Mbyte RAM.

4.3 LCD Module

The chosen LCD module for the prototype was the Palm Technology PMC39T01H [28]. The PMC39T01H is a relatively inexpensive LCD module and was ordered from ACTE [1] as a result of the market survey in appendix B. One example of the PMC39T01H costs 840 SEK, but the cost could be reduced to 500 SEK if the order is 5000 units.

The module is a 3.9" colour STN LCD with a resolution of 320x240 pixels. It has a cold cathode fluorescent lamp (CCFL) as backlight and a 4-wire resistive touch screen mounted on top of the LCD panel. The voltage that the module needs are: one +3.3V for the logic ICs and a contrast voltage (VEE) around +25V at 25 C°. The typical frame frequency is 150 Hz.

4.3.1 LCD Module Interface

The PMC39T01H LCD module from Palm Technology uses a standard 8Bit colour STN interface. The interface consists of 16 inputs as described in figure 4.1. The cable connector mounted on the LCD module is made by Molex¹. A matching SMT 2.0mm bottom connector and cable was ordered from Farnell [14].

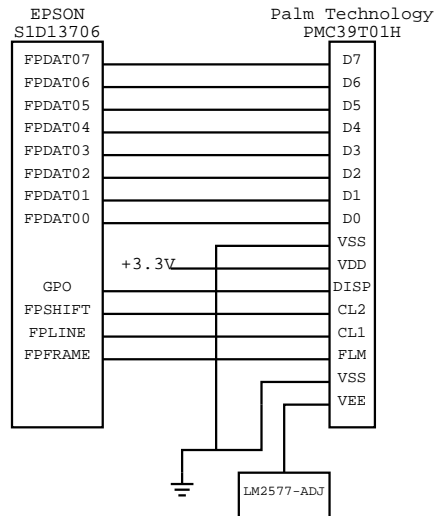


Figure 4.1: Interface between EPSON S1D13706 and PMC39T01H

The control signals to the PMC39T01H comes from the EPSON S1D13706 LCD controller. The signals from the EPSON S1D13706 are:

- FPDAT[0-7]: Data signals that goes to the matching D[0-7] on the PMC39T01H.

¹Molex [25] part 52893-1690.

- FPSHIFT: Is the "data shift clock" that clock pixel data to the PMC39T01H.
- FPLINE: Latch a line of data into place.
- FPFAME: Starts a new frame.
- GPO: Turns the LCD on or off.

Section 2.4 gives more information about the LCD control signals.

Further more, a contrast voltage called VEE is needed by the PMC39T01H. This voltage controls the bias driving voltage inside the LCD module as described in section 2.4. The voltage level of VEE makes the crystal twist and this adjusts how much light which should be transmitted. The VEE needed for the PMC39T01H to have a proper contrast at +25 degrees Celsius is +25V. See section 2.1.2 for more information on contrast adjustment. The AXIS 82 Developer Board does not have a +25V voltage output. To avoid an extra AC/AC power supply unit a step-up regulator is added to the prototype.

There exist a +16V output directly after the rectifier, which perhaps could be used as a input for the step-up regulator. It is easier to transform +16V to +25V than +5V to +25V, but the +16V is not a stable voltage since it depends on the output level from the AC/AC adapter. The output voltage of the AC/AC adapter used to power the AXIS 82 Developer Board can differ. More suitable is the +5V and +3.3V, which has a more reliable voltage thanks to the step-down regulators mounted on the AXIS 82 Developer Board. The +5V output was chosen as input to the step-up regulator.

National Semiconductors [27] delivers a step-up regulator called LM2577-ADJ. ADJ means that it can have an adjustable output as long as the output (V_{out}) is not over $10 \times V_{in}$ and $V_{out} < +60V$. The AXIS 82 Developer Board +5V output is used as V_{in} and V_{out} as the VEE at +25V. The maximal supply current used by the VEE would be 10.0mA. V_{out} should also be adjustable by a user to change the contrast when the temperature changes. See [26] for a full description of how the LM2577-ADJ works. The web site at National Semiconductors [27] has an excellent application to help designing the schematics for different output voltage levels. On the circuit layout, see appendix F, there is a potentiometer on which the user can change the resistant and thereby adjusting the V_{out} from LM2577-ADJ.

4.4 EPSON S1D13706 LCD Controller

The EPSON S1D13706 LCD controller support both monochrome and colour LCD modules (active and passive), with resolutions up to 320x240 at 8bpp (bit-per-pixel). The controller is equipped with an embedded 80kbyte SRAM display buffer, which can be direct accessed by the CPU. See appendix C for more information about the EPSON S1D13706 LCD controller.

One option that this thesis had was if an FPGA should be used and be programmed to control an LCD. However, this was not an option when the EPSON S1D13706 was found on the market. The EPSON S1D13706 costs only 140 SEK when ordered as a single unit and 95 SEK in orders of 1000 units. This is an advanced LCD controller for embedded systems at a low price. Its main drawback is its slow read/write access time, but all LCD controllers on the market has this drawback.

The EPSON S1D13706 LCD controller has a direct interface to a number of different CPUs, for example Motorola M68K, Hitachi SH-4 and Motorola REDCAP2. A simple drawing of how the EPSON S1D13706 will interface the ETRAX 100LX is shown in figure 4.2.

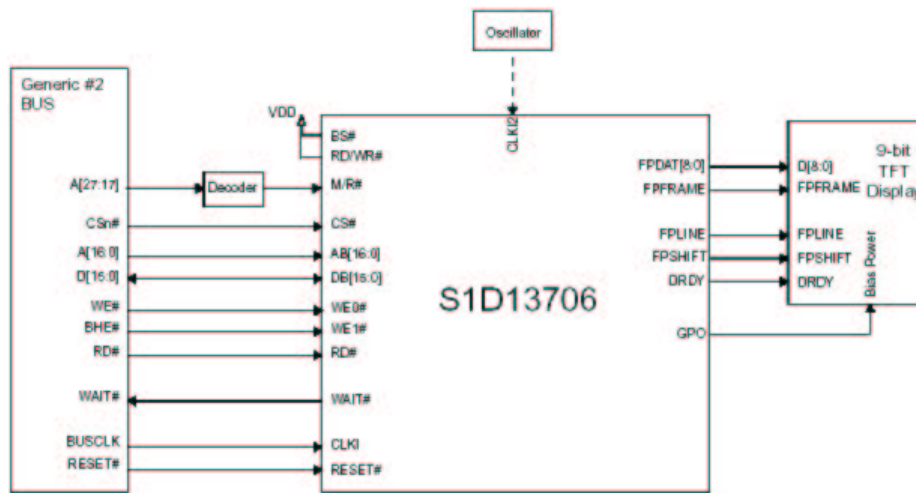


Figure 4.2: Generic bus interface

4.4.1 S1D13706 Interface

The EPSON S1D13706 LCD controller is directly connected to the system bus (AXIS 82 Developer Boards pin headers X14, X15, X18 and X21) like the typical system implementation diagram for Generic #2 Bus in the S1D13706 Technical Manual [12](figure 4.2).

Interfacing the ETRAX 100LX MCM on the AXIS 82 Developer Board like the Generic #2 Bus interface will result in some minor problems.

The first problem is that the address bus on the ETRAX 100LX is designed without A0. A0 is used to address odd or even byte in the 16bit word. To address the bytes on the ETRAX 100LX, the byte enable strobes [7] in the SRAM group is used. To make the byte enable strobes have the same timing as the adress bus, common write enable mode (*cwe*) is used. The byte enable strobes in *cwe* are active during both read and write cycles. The *cwe* can be configured to 16bit or 32bit but since the S1D13706 has a 16bit data bus, *cwe* was set to 16bit.

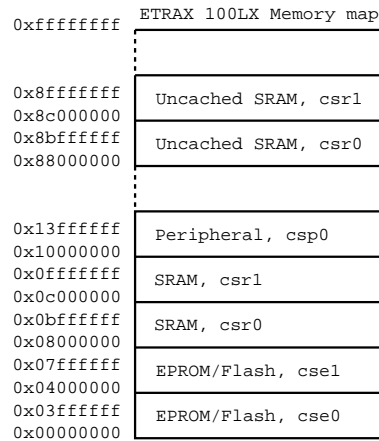


Figure 4.3: ETRAX 100LX Memory map

The S1D13706 LCD-controller is connected to the SRAM chip select 0 (*csr0#*). The *csr0#* is active in the address space 0x08000000-0x0BFFFFFF, see figure 4.5. This gives room for the 80Kbyte video memory and the internal registers in the S1D13706. To get an A0 on the ETRAX 100LX, in this implementation, the lower byte enable strobe (*be0*) on the ETRAX 100LX is used.

The second minor problem is to control when the ETRAX 100LX should be given access to the video memory or the register address space of the EPSON S1D13706. A simple solution to this is to use A17 on the system bus. This gives a block size of 128Kbyte, the lower block (0x08000000) is for the register address space and the upper (0x08020000) for the 80Kbyte video memory.

The third problem is that the LCD controller requires at least one clock to work. This clock is normally, for synchronous interface the bus clock, but since the system bus on the ETRAX 100LX is asynchronous and does not have a bus clock, one external clock is needed to drive the S1D13706. Clocks that already exist on the AXIS 82 Developer Board are one 20MHz clock, which is used by the ETRAX 100LX and one 50MHz clock², used for the SDRAM memory. The maximum clock frequency, which can be used with the LCD controller is 50MHz. Since the clock frequency is proportional to the read/write cycle time, the SDRAM clock at 50MHz will be used to minimise the cycle time. This makes faster access to the LCD controller chip.

The last problem is to get the S1D13706 synchronised with the ETRAX 100LX bus timing. The S1D13706 Generic #2 Bus, that is used for this interface, is designed to activate WAIT when enabled. The WAIT input is active, the ETRAX 100LX is forced to insert extra wait states. When using WAIT with a synchronous bus clock at 50MHz the active time for read/write on a Generic #2 Bus interface, is maximum 160ns. The SDRAM clock that is used for this implementation is not synchronous with the ETRAX 100LX bus timing. The read/write active time for this implementation can therefore not be precise.

²Can be configured to 100MHz, this will not work with the S1D13706

To make the system bus interface to work with WAIT, the number of internal wait states ($ew + lw$) must be three or more. The reason is that WAIT is sampled three clock cycles ($30ns$) before the end of the bus cycle.

For this implementation, when WAIT is used, the bus is configured with one ew , two zw , five lw and 16bit cwe with extended write mode for the SRAM memory area.

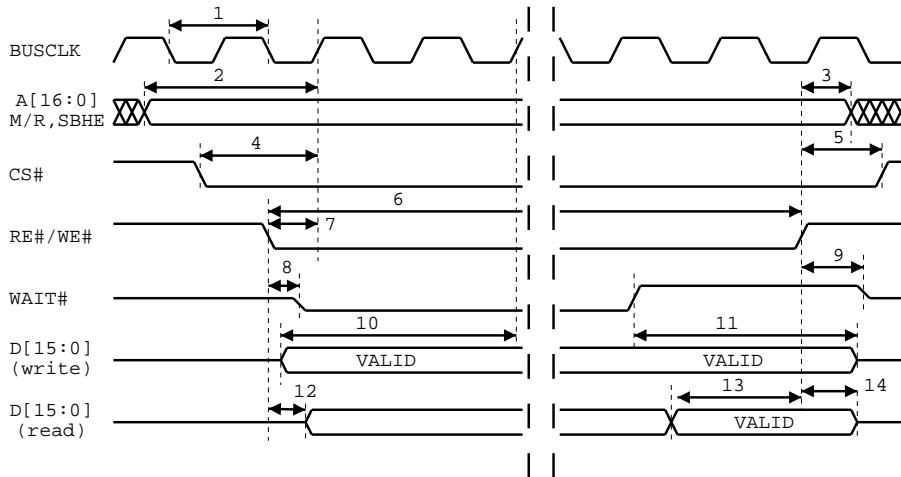


Figure 4.4: EPSON S1D13706 Generic 2 interface timing

No	Explanation	Min	Max	Unit
1	Bus clock frequency	-	50	<i>MHz</i>
2	Setup to first BUSCLK rising edge	1	-	<i>ns</i>
3	Hold from RE# or WE# rising edge	0	-	<i>ns</i>
4	CS# setup to first BUSCLK rising edge	1	-	<i>ns</i>
5	CS# hold from RE# or WE# rising edge	0	-	<i>ns</i>
6	RE# or WE# active width	-	8	BUSCLK
7	RE# or WE# setup to BUSCLK rising edge	1	-	<i>ns</i>
8	Falling edge of RE# or WE# to WAIT# low	3	15	<i>ns</i>
9	Rising edge of RE# or WE# to WAIT# high	3	13	<i>ns</i>
10	Setup to third BUSCLK rising edge	0	-	<i>ns</i>
11	Hold from WAIT# rising edge	0	-	<i>ns</i>
12	RE# falling edge to D[15:0] driven	3	13	<i>ns</i>
13	WAIT# rising edge to D[15:0] valid	-	2	<i>ns</i>
14	Rising edge of RE# to D[15:0] inactive	3	12	<i>ns</i>

Table 4.1: S1D13706 Generic 2 interface timing

To get the cycle time as low as possible, the BUSCLK, for this implementation, is kept as high as possible. With a BUSCLK at 50MHz the RE# or WE# active width³ is maximal 160ns as seen in table 4.1. This will result in reduced performance when

³No 6 in figure 4.4

accessing the LCD controller. When using DMA, buffer overrun fault can occur if the bus is locked when the DMA buffer is becoming full. The internal DMA in ETRAX 100LX, which is used for the Ethernet device, can buffer data for at least 500ns, but when using external DMA this LCD controller interface can result in failure.

4.5 Touch panel

To make the LCD interact with a user a touch panel is included on the PMC39T01H. It is a resistive 4-wire touch panel. More information about different touch panels can be found in section 2.6.1.

There exists a cable at the bottom side of the PMC39T01H, which can be connected to a Molex⁴ FFC/FPC connector. The touch interface consists of an ADS7846, a Winbond 78E54 microprocessor, EEPROM 93C46 and a MAX3232. The ADS7846 that is used in the prototype is connected to a Winbond 78E54 microprocessor which takes the signal from the ADS7846 and calculates it according to calibration values that the Winbond 78E54 has in its memory. The coordinates that has been calculated is sent to the MAX3232 circuit that adjusts the signal to RS232 levels and puts it on Tx. See figure 4.5 on page 30 for block diagram.

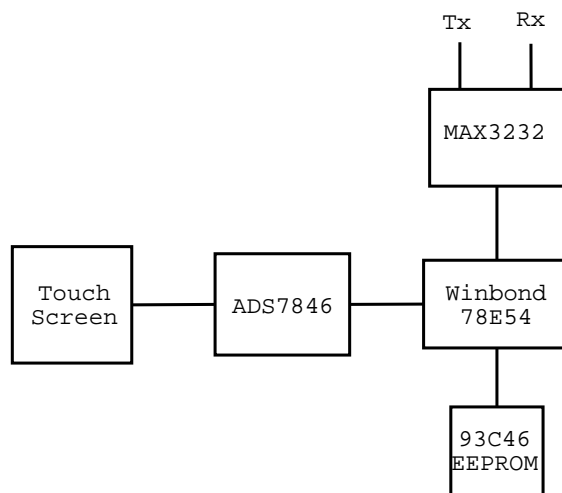


Figure 4.5: Block diagram of touch construction

4.5.1 ADS7846

To use a touch panel, a decoding unit is required for touch point detection. A commonly used device is the ADS7846 manufactured by Texas Instrument [35]. The device is a successive approximation register analog to digital converter. The

⁴Molex [25] part number 5597-NCPB

ADS7846 is connected to the PMC39T01H's touch panel and also to the Winbond 78E54. A touch point is detected by applying a voltage from Y+ on the ADS7846 through the Y plane in the touch screen and then back to Y- on the ADS7846 and then measure the X+ on the ADS7846. This is the Y coordinate⁵. Then it applies a voltage from X+ on the ADS7846 and through the X plane in the touch screen and back to the X- on the ADS7846. A read out is performed on the Y+ on the ADS7846. This is the X coordinate. See figure 4.6 for a detailed circuit diagram.

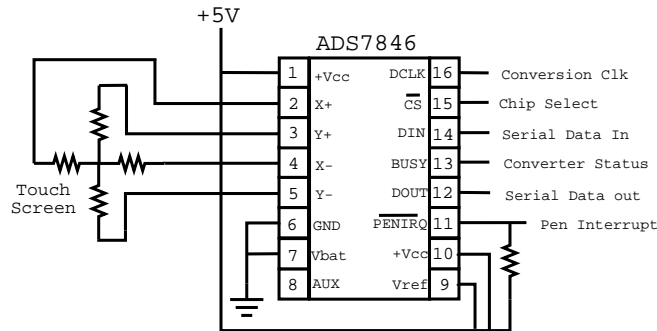


Figure 4.6: ADS7846 implementation

4.5.2 Winbond 78E54

The Winbond 78E54 is a microprocessor, which contains a 16K flash and 256 bytes RAM. Its instruction set is fully compatible with the 8051 microprocessor. The device is connected to the ADS7846 and an EEPROM (93C46) of 1Kb. The EEPROM is for storing calibration data that is needed to give accuracy when calculating touch points. When purchased the 78E54 was shipped with firmware version 1.03.00 by Touch International [36]. The 78E54 calculates a touch point based on the values it gets from the ADS7846. The 78E54 sends them to a MAX3232 IC, which adjusts the signal to RS232 levels. The protocol that the 78E54 sends information with is constructed as a five byte sequence. See table 4.2 for details.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Sync	1	0	0	Pen Down	0	B	B	B
X1	0	X10	X9	X8	X7	X6	X5	X4
X2	0	X3	X2	X1	X0	0	0	0
Y1	0	Y10	Y9	Y8	Y7	Y6	Y5	Y4
Y2	0	Y3	Y2	Y1	Y0	0	0	0

Table 4.2: Report protocol from Winbond 78E54

The packet for point report is defined as:

Packet = Sync (1 byte) + X1X2 (2 bytes) + Y1Y2 (2 bytes) where:

⁵Note that on the prototype the Y+ and Y- are swapped, resulting in that the upper left corner has coordinate 0,0.

Sync = 1 R R S R B B B

S (Pendown) = Status for the point: 1 = Touch Down, 0 = Lift Off

R = Reserved bits.

B = 3 bits counts to indicate the reported point's resolution. (if BBB=0 8 bit resolution is used, if BBB=1 9 bit resolution is used and so on.

X1 = MSB of X axis

X2 = LSB of X axis

Y1 = MSB of X axis

Y2 = LSB of Y axis

Bit four in Sync byte is useful to check if a touch has been made. If the first byte equals to '93' it means that S is 1 and the resolution is B B B is 0 1 1. The touch controller keeps sending data as long as the touch panel is being pressed. Data is '93' on the first byte and the second to the fifth byte is coordinates. When the touch on the touch panel is released the first byte will be '83' instead of '93' and the last coordinate is sent.

The firmware in the Winbond 78E54 version 1.03.0, supplied by Touch International, has instructions for a 25-points, five-points and two points calibration. This means that 25 points, five points and two points on the touch screen is being pressed at a certain pattern and the coordinate in each touch point is corrected to match the relative location on the screen. This calibration is made in a Windows program supplied by Touch International [36]. There exists Linux versions of the program but they have not been tested. The 25 point calibration stores it values in the EEPROM and therefore a 25 points calibration does not need to be made every time the prototype is turned on. The two and five point calibration is stored in RAM (two point) and in the flash memory (five point). The five point calibration is kept in flash memory, which means that the device can use the values after a reboot without a new calibration being performed, but the five point calibration gives less accuracy then a 25 point calibration and we recommend that a 25 point calibration is performed when needed.

4.5.3 Serial Interface

The data from the Winbond 78E54 is sent to a MAX3232 IC that raises the signal to RS232 signal levels. It is then sent out to the Tx pin on a 9-pin serial interface. There exists only two signals from the MAX3232 and they are Rx (receive data) and Tx (transmit data). The Tx and Rx signals are then connected to the RS232 9-pin interface on the AXIS 82 Developer Board.

The 9-pin RS232 interface is defined to have 9 signals. They are as follows:

- Pin 1: Data Carrier Detect
- Pin 2: Received Data
- Pin 3: Transmit Data

- Pin 4: Data Terminal Ready
- Pin 5: Signal Ground
- Pin 6: Data Set Ready
- Pin 7: Request to send
- Pin 8: Clear To Send
- Pin 9: Ring Indicator

If the Tx signal from the MAX3232 is sent directly to the Rx pin on the 9-pin DSUB on AXIS 82 Developer Board some driver software could wait forever for one of the handshaking lines to go to the correct level. It could work, but the most reliable solution is to use a loop back of the handshaking signals and thereby accomplish a handshake. This loop back is done by connecting pin 1,4,6 and pins 7 and 8 on the AXIS 82 Developer Board 9-pin DSUB. See figure 4.7.

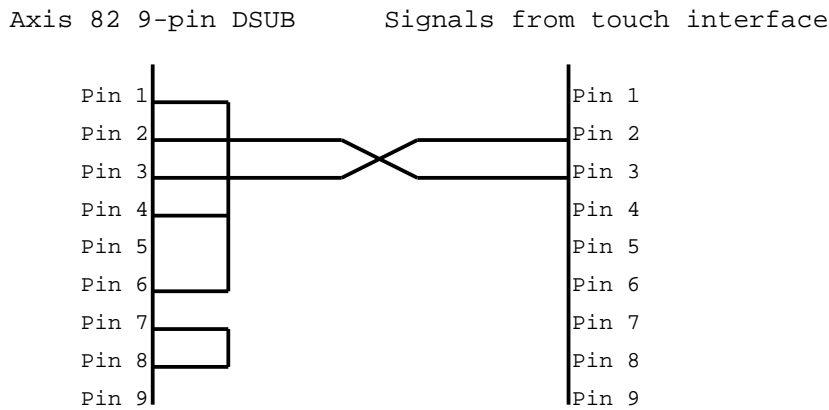


Figure 4.7: Serial Interface

4.6 Schematics and PCB layout

All logic schematics were made in PADS [24] PowerLogic. The schematics were then imported to PowerPCB and PCB routing were done by Peter Johannezon. The add-on card was made with double side, 1 mm thick, epoxy laminate with a size of 100mmx160mm.

Chapter 5

Software Implementation

After reading this chapter the reader should be able to understand the implementation of software in the prototype. The chapter will explain how an example application is built with the EZFB API and Linux framebuffer.

5.1 Linux Kernel Version 2.4.20

The product prototype was developed with the Linux kernel version 2.4.20. The Linux kernel was patched with *linux-2_4_20.diff* that is included in the Axis 82 Developer Board Release 1.90. This patch does not give support for the Linux framebuffer device for the ETRAX 100LX MCM, but with some minor changes in the kernel this can be achieved. These changes has been inspired by the ARM Linux port.

5.1.1 Framebuffer Device

The Linux framebuffer device is used in this implementation as the graphical software layer. On top of the Linux framebuffer device the Linux virtual terminal (virtual console) is used to have a physical user interface to the prototype.

The Linux virtual terminal is enabled in the Linux kernel when defining the virtual terminal option in the Linux config. The Linux virtual console is added to the terminal to get the terminal information to the console screen driver, that in this case is the Linux framebuffer device console driver. To get a working kernel with the terminal console the Linux kernel is configured as shown below:

```
CONFIG_VT=y
CONFIG_VT_CONSOLE=y
CONFIG_UNIX98_PTYS=y
CONFIG_UNIX98_PTY_COUNT=256
```

To build the Linux kernel with the virtual terminal, but without native hardware support for an input device as a keyboard, a dummy keyboard is added to the Linux kernel. The keyboard driver in the Linux kernel uses the function `show_regs` that is called from the `show_ptregs` function in `linux/drivers/char/keyboard.c`. The `show_regs` function is used to make a software interrupt with the keyboard, which displays all the registers in the CPU on the console.

The `show_regs` function is system dependent and must be implemented to the CRIS architecture to get a dummy keyboard interface. The `show_regs` and the function `show_registers` that is used by the CRIS Linux has the same functionality, but the function `show_regs` has not been tested and is implemented as a dummy function.

The dummy keyboard is added to the Linux kernel as shown below:

```
CONFIG_DUMMY_KEYB=y
CONFIG_PC_KEYMAP=y
Added a dummy file: linux/include/asm-cris/keyboard.h
```

Changes in file `linux/arch/cris/kernel/process.c`:

```
void show_regs(struct pt_regs * regs)
{
    printk("The function show_regs is not
           implemented for this architecture\n");
}
```

The Linux framebuffer device is added to the kernel by configuring `CONFIG_FB` and adding the framebuffer logo file as shown below. The logo file contains the image that is displayed on the graphical screen when the Linux framebuffer device is loaded in the kernel. The image is normally Tux, the Linux penguin, located at `linux/include/linux/linux_logo.h`

```
CONFIG_FB=y
Framebuffer logo file: linux/include/asm-cris/linux_logo.h
```

To build the kernel with the Linux framebuffer device enabled, the function `io_remap_page_range()` must be implemented. The function is used by the Linux framebuffer device to add the video memory area to the Linux address space. Like all other architectures, except SPARC, CRIS Linux uses the `remap_page_range()` to create a set of page table entries covering a given physical address range. On SPARC systems, `io_remap_page_range()` use the systems I/O mapping hardware to provide access to I/O memory. The `io_remap_page_range()` function is added to the CRIS Linux kernel as shown below:

Changes in file `linux/include/asm-cris/pgtable.h`:

```
#define io_remap_page_range remap_page_range
```

In order to get the Linux virtual console working, its console driver must be added

in the kernel initialisation. This implementation use the dummy console driver until the framebuffer device is loaded into the kernel. When the framebuffer device is ready it will take over the dummy console driver with the framebuffer console driver. This will display all the console output on the graphical LCD connected to the LCD controller on system bus.

How to enable the dummy console for the Linux kernel is shown below:

CONFIG_DUMMY_CONSOLE=y
Changes in file: *linux/arch/cris/kernel/setup.c*

```
#include <linux/console.h>

void __init
setup_arch(char **cmdline_p)
{
    ...

    #ifdef CONFIG_DUMMY_CONSOLE
    conswitchp = &dummy_con;
    #endif
}
```

5.1.2 Epson S1D13706 Device Driver

The Linux framebuffer device driver for the LCD controller can be added to the Linux kernel as described in section Linux Console Driver in the S1D13706 Technical Manual [12].

The device driver for the LCD controller depends on a configuration file that contains all the hardware specific information about the LCD controller and the LCD module. To generate a working configuration file for the S1D13706 device driver the Windows program *13706cfg* was used.

The generated configuration file was changed since a bit-shift in the *S1D_WRITE_PALETTE()* macro divided every colour in the palette by 16, resulting in a dark palette.

Changes in the file *linux/drivers/video/epson/s1d13706.h*, which was generated from *13706cfg* are shown below:

```
#define S1D_WRITE_PALETTE(p,i,r,g,b)
{
    ... = (S1D_VALUE)((r));
    ... = (S1D_VALUE)((g));
    ... = (S1D_VALUE)((b));
    ... = (S1D_VALUE)(i);
}
```

The video memory and registers are accessed as a noncached memory. To speed up the device, cached video memory can be used. The cost of using cached video memory is that the screen will not be updated until the cache is written to the video memory. This happens when the cache is full or when a content switch accrues. The original framebuffer device driver for the S1D13706 use the function *ioremap_nocache()*, since noncache is accomplished in ETRAX 100LX by adding 0x80000000 to the physical address the function *ioremap()* is used instead.

To control the LCD module power-on sequence a delay in the initialisation code for the S1D13706 device driver is added before the GPO control¹ is turned on.

5.1.3 Keyboard

The keyboard is used to get a user interface for the console. The simplest way to get this is to enable the USB HID (Human Interface Devices) and use a USB keyboard. The configuration and changes in the Linux kernel is shown below:

```
CONFIG_USB_HID=y
CONFIG_USB_HIDINPUT=y
CONFIG_INPUT=y
CONFIG_INPUT_KEYBDEV=y
```

Changes in file: *linux/drivers/input/keyboard.c*

```
#if ...
...
defined(__arm__) || defined(__cris__)
```

5.2 Microwindows

For the prototype the X-like API Nano-X was used and build for the CRIS architecture using the Linux framebuffer as graphical device. Microwindows is designed to be cross compiled for different systems, like the SuperH or StrongARM architecture. To get Microwindows built for CRIS some changes in the files *config* and *Arch.rules* was made. This was done to get the “cris” prefix for the gcc tools and the path to the cross-compiled Linux kernel.

The touch panel driver for Microwindows was done by making changes in the MicroTouch touch panel driver, which is included in the Microwindows source code, see section 5.3 for more information.

¹Bit-7 in register 0xad in S1D13706

5.3 Touch panel interface

The touch panel interface is accessed by the software through the RS232 serial device node at `/dev/ttyS2`. The device node is set to no parity, 8 data bits and one stop bit at 9600 baud. To read data from the touch panel a non-blocking read is used. This is to achieve a continuous flow in the program. The data is transmitted from the touch panel to the AXIS 82 Developer Board when a touch has been made, as described in section 4.5.2.

To transform the reported data from the touch panel into a X,Y coordinate that represents the physical pixel coordinate on the touch panel these formulas are used.

$$PressedPixelX = \frac{ReportedX}{MaxXReportVaule} * MaxPixelX$$

$$PressedPixelY = \frac{ReportedY}{MaxYReportVaule} * MaxPixelY$$

The maximal report value for both X and Y in 11bit resolution is 2048 (2^{11}), and the LCD panel has 320,240 pixel as maximal X,Y. When touching the upper left corner 0,0 is calculated and 320,240 when pressing bottom right. The result is inverted compared to the expected result, since the Y axis has been swapped in hardware. This to get a result that can be used in the application without having to make extra calculations.

5.4 Demonstration application

A demonstration application was built to show how the interaction between the LCD and ETRAX 100LX can be done. The application is developed as an AXIS Document Server. The application has no scanner attached to it and the scanned image already exist in the memory of the ETRAX 100LX.

EZFB API has been used to display the images on the LCD. The touch point coordinates is read from the serial port.

The application is build up as a user interface with two arrows and a green OK button (upper button) and a red Cancel button (lower button). These elements of the picture are clickable. See figure 5.1

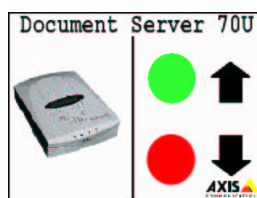


Figure 5.1: The user is presented with this picture at the start up.

The up and down arrows should act like a scroll list for all users that are in the

user list. For each touch on the arrow a new person will show up on the left of the screen. In this application there exist two users. The first user shows up when the upper arrow is touched. The picture looks like figure 5.2.



Figure 5.2: Picture, name and email address of one user when the upper arrow is pressed.

After a receiver has been chosen an image should be scanned. This is accomplished by pressing the upper green circle. When this is done an image as shown in figure 5.3 will be displayed.

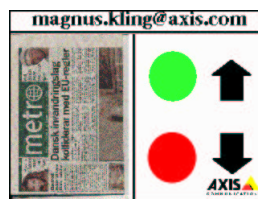


Figure 5.3: The scanned image is shown to the left and receivers email address is displayed on the top of the picture.

When the scanned image looks OK a touch at the upper circle will "send" the image to the receiver. This is simulated and is not happening. If the lower circle is pressed at any time the application will return to its initial state as showed in figure 5.1.

Chapter 6

Results

6.1 Market Survey

The prototype implementation is not cost effective due to the high cost of buying a single LCD module. The used LCD module, Palm Technology PMC39T01H cost 840 SEK in quantities of one unit. If ordered in quantities of 5000 units, the cost per unit drops to 500 SEK. PMC39T01H is a colour LCD with an integrated touch panel. If a less advanced LCD module is used, as the Tianma TM320240ACIWLG (FSTN) from Egevo, the cost can be reduced to around 160~180 SEK including a touch panel. The drawback with the TM320240ACIWLG is that it is monochrome. This has the effect that the GUI must show different symbols instead of different colours. For example the Cancel button can not be red because the red colour will only show as grey on the LCD module. The TM320240ACIWLG is of transfective type. It is delivered without backlight. A transfective LCD module works well in normal office ambient light and if needed there is a possibility to add backlight to the TM320240ACIWLG. The TM320240ACIWLG has no internal power supply circuit for the bias voltage and the needed bias voltage must be supplied according to the specifications of the LCD module.

Our recommendation is to consider if a colour LCD module is needed for the application or if a cost effective monochrome LCD module could do the work. This has a big effect on the total cost for the application. Consider that the Document Servers COG and buttons have a price near 150 SEK. A TM320240ACIWLG with EPSON S1D13706 and connector cable have the price of ~300 SEK. In appendix ?? is three price examples for a low, mid and high price product(LCD module, LCD controller, inverter, touch controller and cables).

The chosen LCD controller, EPSON S1D13706, supports a lot of LCD modules as seen in the specification in appendix C. This gives the freedom to choose different LCD modules and still use the same LCD controller. All companies in the market survey recommended us to use an EPSON controller. EPSON controllers seem to have a good reputation on the market and they are affordable. Our own research on

the Internet showed that EPSON is the only manufacturer that makes low power LCD controllers for the embedded market.

6.2 Prototype

The prototype, which was build, works as expected. The EPSON S1D13706 can communicate with the ETRAX 100LX directly over the system bus. The measured bus timing between the EPSON S1D13706 and the ETRAX 100LX can be found in the pictures in appendix D. This should be compared to the estimated value in section 4.4.1. The bus interface, in the prototype, will use the bus for about 200ns when accessing the video memory in the EPSON S1D13706. 200ns is a relatively long time, compared to 110ns when accessing the flash memory on the AXIS 82 Developer Board. When using external DMA this 200ns bus lock can case a buffer overrun in the external device.

The add-on card is connected to the AXIS 82 Developer Board with a pin list on the system bus. This pin list can be damaged when the add-on card is disassembled too often. This damage can result in disturbance in the interface, which can be seen as pixel error on the LCD module or in that the AXIS 82 Developer Board can not be flashed. The solution, for reducing this disturbance, is to built a card that includes both the LCD controller and ETRAX 100LX.

The built-in touch panel works well after finding out that the +Y axis had a connection problem. This was solved by applying a gentle pressure on that specific line. The ADS7846, Winbond 78E54 and the MAX3232 work together as calculated with only one problem, which was that the connector from the touch panel to the ADS7846 needs a four pin connector from MOLEX. This connector could not be found so a ten pin connector where fitted to the add-on card. The four pin connector from the touch panel fits in the ten pin connector from MOLEX. However, this is not an optimal solution.

The LCD contrast voltage (VEE) should be +25V and this is achieved by stepping up the +5V on the AXIS 82 Developer Board with a LM2577-ADJ voltage regulator. This circuit needs a coil that is too big to be used in a commercial applications. A more suitable solution should be to use a special IC, as the MAX629, for driving the +5V to +25V.

The inverter used in the prototype is not the most suitable for the LCD module. The inverter is normally driven by +5V, but with +5V the inverter leaves too much voltage on the output (500V), which could damage the CCFL on the LCD module. By driving the inverter with +3.3V an output voltage near the specifications of the PMC39T01H is reached (350V). This solution is submitted by Palm Technology's engineers.

The total cost for the whole prototype is difficult to calculate, because the price for individual resistors and capacitors are not found but should end around a total cost of 100 SEK. The cost for the PMC39T01H, EPSON S1D13706, inverter,

ADS7846, Winbond 78E54, PE-53115 inductor, connectors, cabel and the LM2577-ADJ is 1641 SEK. The AXIS 82 Developer Board is not included in any sums given.

The framebuffer device in the Linux kernel was enabled and works without any limitations on the CRIS Linux. The framebuffer applications, used in the prototype, can be used in both the prototype and on the host computer.

6.3 Performance

The test, described below, should be considered as a guideline of the performance, which could be expected from the prototype.

The performance was tested by using the tool "tcp_perf", which can be found in the software package included with the AXIS 82 Developer Board. This program simulates network traffic on UDP and TCP. By stressing the network device to its maximum with "tcp_perf" and at the same time updating the video memory the number of frames per second can be calculated. The network that was used had a maximum speed of 100Mbits/s.

Without a network load we reached 93.8 frames/s. When simulating network traffic using TCP with package size of 2000 bytes, we reached 46.1 frames/s and 1.6 Mbyte/s.

Chapter 7

Touch LCD in Axis Products

7.1 Network Document Servers

The AXIS Network Document Server works by providing users the ability to send scanned documents instantly over the network to e-mail addresses, file servers, printers, URLs and the Web. The network document servers provides a faster, cheaper and safer method of distributing documents than by fax, traditional mail or courier.

The AXIS Network Document Server already has a user interface which consist of a character LCD, with a keypad and a external PC keyboard. With a touch LCD the character LCD and the keypad can be eliminated. The user will instead use the touch functionality to make his selection. This product uses the OSYS operation system and to have the same support for scanners in Linux, a new set of drivers must be implemented.

7.2 Network Cameras

A network camera can be described as a camera and computer combined in one intelligent unit. It captures and transmits live images directly over an IP network (e.g. LAN/intranet/Internet), enabling users to remotely view and/or manage the camera from a Web browser on any computer, anywhere, at any time.

The AXIS Network Cameras use Linux and the ETRAX 100LX. The product works by taking RGB signals from a camera to the ARTPEC chip for JPEG transformation. The JPEG data stream is then sent to the ETRAX 100LX that sends the stream on the network. The CPU load, while operational, is low since it is mainly used to put the JPEG stream on the network. To put the JPEG image on the LCD module, image transformation must be performed. This operation will require CPU load and therefore only display a minimum of images. A touch LCD interface can be implemented, but the practical usage for an LCD on a network camera is limited.

7.3 Video Servers

The AXIS Video Servers is used to convert analog live video signals to digital. The digital data is then transmitted over the network and received as a live image from a Web browser on any networked computer. This product is used when old analog cameras is integrated with a new digital video system like the AXIS Network Cameras.

Video servers work in a similar way as the AXIS Network Cameras, it only needs a external RGB signal. An LCD user interface, for the video servers, can be more or less useful, it can be used to display the images from the cameras but a live image from a Web browser on a PC is more efficient.

7.4 Network DVR

The network digital video recorder is used to record data from analog cameras, and simultaneous gives access to live viewing, playback and administrative functions from anywhere using an Internet Explorer Web browser. The data is stored on its internal hard drives.

A Touch LCD user interface for the AXIS Network DVR can be useful, but the product is made to be controlled by a web browser and does not need a physical contact with the user.

7.5 Developer Boards

The AXIS 82 Developer Board is the developer board for ETRAX 100LX MCM 2+8. It also serves as a higher-end deployment platform for system integrators.

An LCD interface could be implemented to the developer board, which could mean that the system integrators or developers could benefit from using an LCD.

7.6 New product: NetScreen

One new invention for Axis Communications is a network display, the NetScreen. An ETRAX 100LX MCM is connected to an LCD controller and an LCD is connected to the controller. This is built together to form an attractive solution.

With this unit, media can be transferred to the display directly over a network. Plug the NetScreen to the network and it will discovery its surroundings, show video, act as a videophone, as a terminal, as a PDA, a cook book or whatever you want it to be.

Chapter 8

Discussion

The objectives for this thesis was to find a cost effective solution for usage of touch displays in new Axis products, using an ETRAX CPU. A prototype system to illustrate the problem, as well as the solution had to be made.

To get an overview of the LCD market, a market survey was performed in the beginning of the project. Five companies, which exist on the Swedish market for LCD and touch screens, was contacted and a request for a tender on LCD, LCD controller and touch panel controller was sent to them. The market survey resulted in what components that had to be included to construct a prototype, which could be used to illustrate the thesis objectives. We choose to order all the specialised components; LCD module, LCD controller, inverter and touch controller from ACTE since they were able to deliver one sample unit of each component. The most cost effective LCD module was found at Egevo, but they required an order of 20 units minimum. Components needed to build the prototype, were taken from Axis Communications hardware laboratory. Components not found there were ordered from Elfa and Farnell. The prototype was built in the Axis Communications hardware laboratory with the help of the people there.

To get an overview of the ETRAX 100LX bus interface, we asked Per Zander (Expert Engineer, Axis Communications) for help. After Zander looked at the timing diagrams for the EPSON S1D13706 LCD controller, Zander considered that it was possible to interface the EPSON S1D13706 with the ETRAX 100LX MCM. To get more information about an asynchronous interface, to the EPSON S1D13706, we contacted the Epson Research and Development and got helpful support from their engineers.

The prototype consist of the AXIS 82 Developer Board with an add-on card, which is connected to the system bus on the AXIS 82 Developer Board. On the add-on card the EPSON S1D13706, contrast power module, touch controller and connectors are fitted. To get an 50MHz clock for the LCD controller a wire were soldered from the SDRAM on the AXIS 82 Developer Board to the EPSON S1D13706. Consider, that this was made as a prototype solution. Taking a 50 MHz clock over a ten centimeter cable should be considered as a less good alternative when making a production example.

The LCD controller is accessed from the Linux kernel as a framebuffer device. The reason for using Linux in this project instead of OSYS, is that we had good knowledge of Linux and that OSYS is not used in any new product lines at Axis Communications. The documentation of OSYS is very limited and hard to find. We could not find a single document until Jens-Henrik Lindskov mailed us a link to the OSYS USERS GUIDE. Reading the document provided a bit of information. OSYS does not include any graphical implementations such as a framebuffer device. The Linux framebuffer device is a standard component in the Linux kernel, but it has not been used for the CRIS Linux port (we can not find any information stating that it has been used before at Axis Communications). We thought this would result in problems or reduced functionality in the framebuffer device, but after a week of tracking in the kernel the framebuffer device was enabled, using the virtual framebuffer device on the CRIS Linux architecture.

The thesis problem is stated as "...investigate the possibility of using touch displays in new Axis products...". If read, word by word this would mean the use of only a touch screen without an graphical LCD module attached to an Axis product. That problem could be solved easily. This is possible because a simple touch screen uses the serial interface to communicate with a CPU. Most of Axis products includes a serial port. It could also be read as that an LCD module interface should be implemented to an Axis product. We stated the problem as an LCD interface should be implemented to an Axis product.

The result for this thesis is that touch LCD modules can be used in future Axis products. The touch LCD modules can not be considered as a cost effective solution for the Axis document servers, when comparing to the keypad and the character display used today. The main reason for an graphical LCD module, with a touch panel, on the document servers is to have an attractive user interface and be able to show previews of the scanned image.

The EPSON S1D13706 LCD controller uses WAIT to control the bus timing. This will result in reduced performance when accessing the LCD controller, since it stalls the bus for about 200ns every time it is accessed. A solutions to the problem is perhaps to implement an LCD controller in the next generation of the ETRAX CPU. The Winbond 78E54 microprocessor, which is used in the touch interface from Touch International, is not a cost effective solution. A better solution is to access the ADS7846 direct to the ETRAX 100LX through the general I/O port. The reason that we did not do this is that we thought, in the beginning, that the LCD interface should be hard to implement. This resulted in that the touch interface got a low priority.

The knowledge that we have learned during our thesis work includes LCD, Linux framebuffer device, design of a circuit board and CRIS architecture.

Axis Communications have constructed a good working environment at Edison Park, well suited for their needs. The employees have all high competences and they are always willing to help at any time. We have had a great opportunity to do our master thesis here at Axis Communications and we would like to thank Axis Communications and their employees and wish them all the best in the future.

Appendix A

Abbreviations and Acronyms

ATA	Advanced Technology Attachment
API	Application Programming Interface
ARM	Advanced RISC Machines, CPU design by Digital.
CCFL	Cold Cathode Fluorescent Light
CD	Compact Disc
COG	Chip On Glas
CPU	Central Processing Unit
CRIS	Code Reduced Instruction Set
CRT	Cathode Ray Tube
CSTN	Colour Super Twisted Nematic
DMA	Direct Memory Access
DVD	Digital Versatile Disk
EEPROM	Electrically Erasable Programmable Read-Only Memory
ETRAX	Ethernet Token Ring AXis
FPGA	Field Programmable Gate Arrays
FRM	Frame Rate Modulation
GUI	Graphical User Interface
IC	Integrated Curcuit
IR	Infra Red
LCD	Liquid Crystal Display
MCM	Multi Chip Module
MIPS	Million Instructions Per Second
MMU	Memory Management Unit
OEM	Original Equipment Manufacturer
OLED	Organic Light Emitting Diode
PTY	Pseudo Terminal
RAM	Random Access Memory
RGB	Red, Green, Blue
RISC	Reduced Instruction Set Computer
RS232	Serial communications protocol
SAW	Surface Acoustic Wave
SCSI	Small Computer System Interface

SPARC	Scalable Processor Architecture
SRAM	Static Random Access Memory
STN	Super Twisted Nematic
StrongARM	CPU architecture from Intel(built on ARM by Digital)
TFT	Thin Film Transistor
TN	Twisted Nematic
Token Ring	Local area network, with a ring or star topology.
TTY	TeleTYpewriter. In unix, all sort of terminals.
μClinux	Linux for systems without a MMU
USB	Universal Serial Bus
VGA	Video Graphics Array
VRAM	Video Random Access Memory

Appendix B

Market Survey

During March to April a market survey was performed to find an answer on which products the market had to offer, according to our specifications for the thesis.

As one of the thesis objective were to find a cost efficient solutions for implementation of an LCD interface to ETRAX 100LX, the large savings could be made by finding an cheap LCD module. That would be the most expensive part in the solution.

The maximum cost for an LCD module was set to 500 SEK. To get such a low price on an LCD module, one have to either buy a lot of LCDs or to find an LCD module that perhaps have less good quality but a better price. The survey did not try to favour either of the two alternatives.

Five contractors were sent an email with a request to write back with what they had to offer for this specification:

- LCD module within the size 2" to 7", with a resolution of 320x240 pixels, colour or monochrome, backlighting, transmissive or reflective, that cost less or max 500 SEK.
- An LCD controller module.
- A touch panel that could be fitted to the recommended LCD module.
- A touch controller that could control the recommended touch panel.
- Inverter, if needed for backlighting.

The small number of contractors was based on that we where not able to find more than five suppliers in Sweden, that could deliver small size LCDs. The contractors where Acte, Egevo, Truly, Scancraft and TRG. All of the contractors have worked with Axis Communications before this date, except TRG that is new to Axis Communications.

Acte [1] represents large manufactures such as Epson [13], Kyocera [21], Atmel [5] and TDK [11]. They are part of a group of technical distributors with sister companies in Sweden, Norway, Denmark, Finland, Poland, The United Kingdom and Germany. They work with electronics components, electromechanics and industrial automation.

Egevo [10] is part of an international company called OEM International AB. They work with electronic components and power supplies.

Truly Semiconductors [39] is a Hong Kong based company that produces custom made LCD modules. In Sweden they are represented by Anders Swedin.

TRG [37] is a joint venture between BB Avionic System AB and TRG Components USA. TRG supplies electronic components in Scandinavia and Finland.

Scancraft [32] is a supplier that have specialised in LCD, displays and electronics. They represent companies such as Kyocera, Powertip [30] and RiTek [31].

All contractors have replied and meetings with Egevo and Truly have been made. Following is a short description of the items asked for and a summary of what the companies recommended.

LCD Modules: There exist mainly three types of LCD modules on the market. STN (Super Twisted Nematic), CSTN (Colour Super Twisted Nematic) and TFT (Thin Film Transistor). STN and CSTN is based on the same technique but the difference is that CSTN is a colour monitor. TFT is a colour monitor that have a faster response time than STN or CSTN, resulting in better performance when showing movie or other moving pictures on the LCD. Large manufactures of LCDs are Kyocera, Hitachi [16], Arima [3], Citizen [9] and Varitronix [40]. These manufactures have all quality LCD modules. Kyocera has the best quality and as a result the most expensive LCD module.

LCD controller: To use an LCD module the ETRAX 100LX must communicate with the module by sending data and control signals. That is not a suitable workload for a processor that has more important things to do, eg. taking care of user inputs and network traffic. The solution is to let the ETRAX 100LX communicate with a controller chip used for controlling the LCD module. The most commonly used controller chips are manufactured by Epson and Asilant (former known as C&T). The LCD controller has its own video memory (VRAM) and the ETRAX 100LX writes data to the controller, which sends the data to the LCD module.

Touch panel: A touch panel is typically made by a glasplate and an electrical grid placed on top of the glasplate. The function of a touch panel is that, if the glasplate is being pressed a intersection in the grid causes an external device to recognise the touch point. This touch point can be detected by the operating system as a mouse click or mouse movement. The touch panel is mounted on top of the LCD module. There exists different techniques to recognise a touch point. The most cost effective one is the resistive model.

Touch panel controller: A touch panel controller is basically an analog to digital switch, which sends out a pulse on the touch panel and reads the output from the touch panel. Then it converts the analog signal to digital signals and sends them to a microprocessor, which calculates a touch point. The microprocessor can have different programs, which calibrates the touch panel.

B.1 Acte

Acte gave us four alternatives for an LCD module, all from the manufacturer Palm Technologies. Two of them had touch panel integrated in the module. See table B.1 for their specifications and price.

Model	Size	Res	B/L	T/P	SEK	Qty	Tech
PMC39T01H	3.9"	320x240	CCFL	yes	500	5k	CSTN
PMC28T01B	2.8"	320x240	CCFL	yes	370	5k	CSTN
PMG2432D	4"	320x240	EL	no	300	5k	FSTN
PMG3224A2-SBF	5.7"	320x240	CCFL	no	430	5k	STN

Table B.1: Res= resolution, B/L = type of backlight, T/P = included touch panel, Tech =technology the module uses.

As for LCD controller chip we where recommended the EPSON S1D13705 or EPSON S1D13706. The difference between the controllers is that S1D13706 is newer and a bit faster which could be of great importance when making an interface to ETRAX 100LX. The price was 88 SEK/pcs for S1D13705 and 95 SEK/pcs for S1D13706, both in quantities of 1000 units. They also recommended a touch panel controller chip packages from eTurbotouch that included a Winbond W78E54 microprocessor and an ADS-7846 for a price of 120 SEK/pcs in quantities of 1000 units. The price could drop a few SEK if larger quantities of LCD controller and touch panel controller are ordered. They also recommended a TDK inverter that could be used together with the backlight.

B.2 Egevo

In the beginning of April we had a meeting with Marcus Ånhammar, a sales representative from Egevo. He explained different ideas and technologies behind LCD

modules. He also suggested that in an office environment a backlight is preferred but not necessary needed. The week after the meeting he mailed Egevos alternatives for our specifications. See table B.2.

Model	Size	Res	B/L	T/P	SEK	Qty	Tech
TM320240ACIWLG	3.9"	320x240	REFL	yes	180	5k	FSTN
MC39T01B	3.9"	320x240	CCFL	no	484	5k	CSTN
MC28G01B	2.8"	320x240	CCFL	no	370	5k	CSTN

Table B.2: Res= resolution, B/L = type of backlight, T/P = included touch panel, Tech =technology the module uses.

As can be seen in the table B.2 Egevo recommends a cheap LCD module, which uses FSTN and includes a touch panel. At a price of 180 SEK/pcs at a quantity of 5000 units, this is an affordable price for an LCD that is reflective. Egevo recommended the LCD controller EPSON SED1335 at a price of 80 SEK in quantities of 1000 units. They did not recommend any specific touch panel controller but gave an indicator on 200 SEK for an complete module that could decode a touch panel. An inverter was not named but a price of 100 SEK in orders of 1000 units was given. Smallest order for LCD modules are 20 units.

B.3 Truly Semiconductors

A meeting with Anders Swedin from Truly Semiconductors were held on the 24:th April 2003. He explained what Truly Semiconductors had to offer. They are specialised in making custom LCD modules. They have standard modules that they can make but often the customer makes a specification and sends it to Truly, which then makes an LCD module according to the specification provided. They can match an LCD controller and touch controller to the specification. This results in very cheap LCD panels but with the disadvantage that it must be ordered in great quantities to get a low price

The standard LCD modules, which they can offer without making a custom model, is limited but Anders Swedin showed one CSTN module. The module is named MCG-G128160DFSW-3W and is a dual LCD module. Dual means, in this product, that it has two displays. One large, 128x160 pixel, and one small, 96x64 pixels. They are fitted with a LED backlight and the large screen is capable of showing 65k colours. The cost for this module was 30\$ plus a touch panel 4\$. On top of that cost comes a initial crafts cost of 1500~2000\$.

Truly Semiconductors can make customised LCD modules, but the price is a bit too high when making small series of LCD modules.

B.4 TRG

TRG recommended an LCD module by Gunze USA. It has a built-in touch panel and the LCD module is of STN type. See table B.3 for comparison. TRG have recommended a touch panel controller by the name CRS1 from Gunze USA. It is a controller that has a built in A/D converter and is capable of communicating with PS/2 interface. The cost for a quantity of 5000 units is 21 SEK each. As comparison a module with the CRS1 chip, different connectors and ready to use, costs 439 SEK in quantities of 5000 units.

Model	Size	Res	B/L	T/P	SEK	Qty	Tech
G-22-7	5.7"	320x240	CCFL	yes	846	10	FSTN
G-22-7	5.7"	320x240	CCFL	yes	792	500	FSTN
G-22-7	5.7"	320x240	CCFL	yes	746	1k	FSTN

Table B.3: Res= resolution, B/L = type of backlight, T/P = included touch panel, Tech =technology the module uses.

Minimum order for LCD modules from TRG is 20 units.

B.5 Scancraft

Scancraft had a comment to our specifications and that was to consider the technology called OLED (Organic Light Emitting Diode) [20] which is a technology that don't need a backlight because the organic material is self luminous. We received a test LCD module with a resolution of 128x60 pixels. It could not show multiple colours only green. Apparently green is the only available colour at the moment. Colour OLEDs will be more common in a few years. The disadvantage with OLED is its lifespan. It only has a lifetime of three years. After three years the light from the organic material is reduced. We became clear that this is a very interesting technology, but for now it is not suited according to our specifications.

Scancraft recommended a Kyocera CSTN 5.7" LCD module with a price of 1370 SEK/pcs at a quantity of 10 000 units. They also recommended a non specified inverter at a cost of 150 SEK.

B.6 Price examples

Three different price examples will be displayed to show the wide price range, which a complete LCD module, with all components included, will cost. The range is divided in three levels. A low, mid and high price range. The components are picked from the result of the market survey. The high price is in 10 000 units. The low and mid price is for 5000 units.

Component	Type	Price	Qty
LCD module	TM320240ACIWLG 3.9"	180 SEK	5K
LCD controller	EPSON S1D13706	95 SEK	5K
Touch controller	Touch International Kit	100 SEK	5K
Cables and needed components		100 SEK	
Total cost:		470 SEK	

Table B.4: Low price example for all components needed to build an LCD interface.

Component	Type	Price	Qty
LCD module	PMC39T01H CSTN 3.9"	500 SEK	5K
LCD controller	EPSON S1D13706	95 SEK	5K
Touch controller	Touch International Kit	100 SEK	5K
Inverter	TDK NJL	85 SEK	5K
Cables and needed components		100 SEK	
Total cost:		880 SEK	

Table B.5: Mid price example for all components needed to build an LCD interface.

Component	Type	Price	Qty
LCD module	Kyocera CSTN 5.7"	1370 SEK	10K
LCD controller	EPSON S1D13706	90 SEK	10K
Touch controller	Touch International Kit	80 SEK	10K
Inverter	TDK NJL	80 SEK	10K
Cables and needed components		80 SEK	
Total cost:		1700 SEK	

Table B.6: High price example for all components needed to build an LCD interface.

Appendix C

EPSON specification

C.1 Epson S1D13706: LCD support

The S1D13706 supports a wide versatility of different panel types:

- STN 4/8-bit monochrome LCD interface
- STN 4/8/16-bit colour LCD interface
- TFT 9/12/18-bit interface
- 18-bit Epson D-TFD interface
- 18-bit Sharp HR-TFT interface

C.2 Epson S1D13706: CPU Interface

- Direct support of the following interfaces:
 - Generic MPU bus interface using WAIT# signal.
 - Hitachi SH-3.
 - Hitachi SH-4.
 - Motorola M68K.
 - Motorola MC68EZ328/MC68VZ328 DragonBall.
 - Motorola “REDCAP2”-no WAIT# signal.
- 8-bit processor support with “glue logic”.
- “Fixed” low-latency CPU access times.
- Registers are memory-mapped - M/R# input selects between memory and register address space.
- The complete 80K byte buffer is directly and contiguously available through the 17-bit address bus.

- Single level CPU write buffer.

C.3 Epson S1D13706: Display Modes

- 1/2/4/8/16-bpp colour depths.
- Up to 64 gray shades using Frame Rate Modulation (FRM) and dithering on monochrome passive LCD panels.
- Up to 64K colours passive STN panels.
- Up to 64K colours on active matrix LCD panels.
- Example resolutions:
 - 320x240 at a colour depth of 8 bpp
 - 160x160 at a colour depth of 16 bpp
 - 160x240 at a colour depth of 16 bpp

Appendix D

Measured interface timing

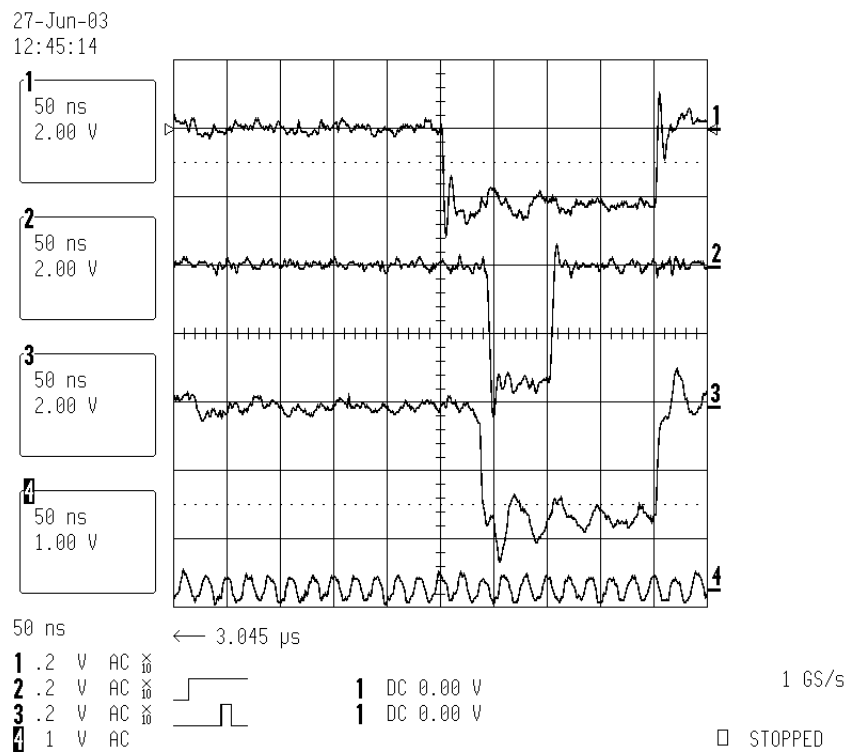


Figure D.1: 1=CS, 2=WAIT, 3=WE, 4=50 MHz Clock

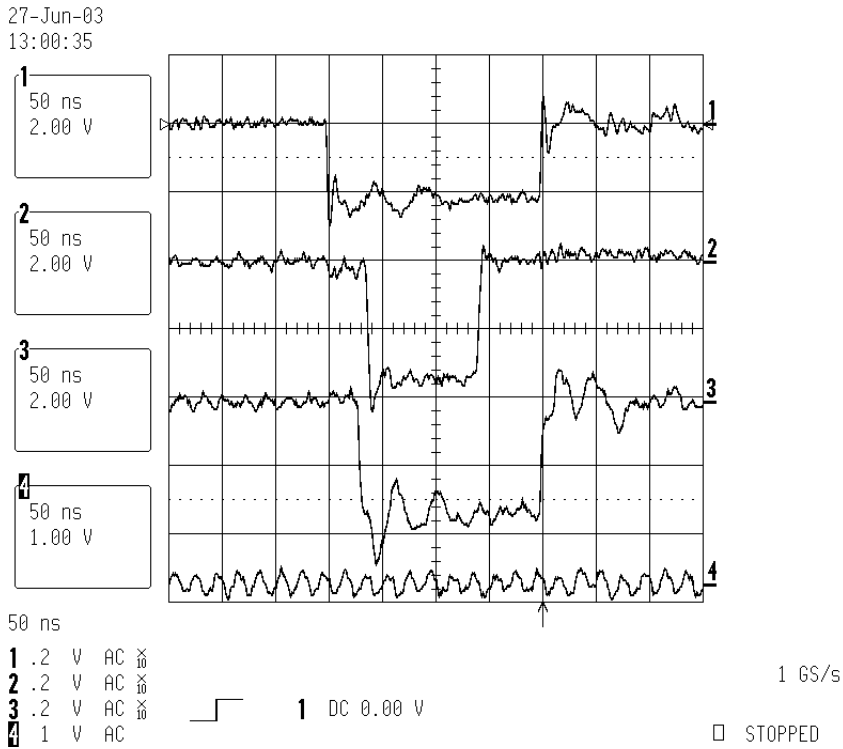


Figure D.2: 1=CS, 2=WAIT, 3=RD, 4=50 MHz Clock

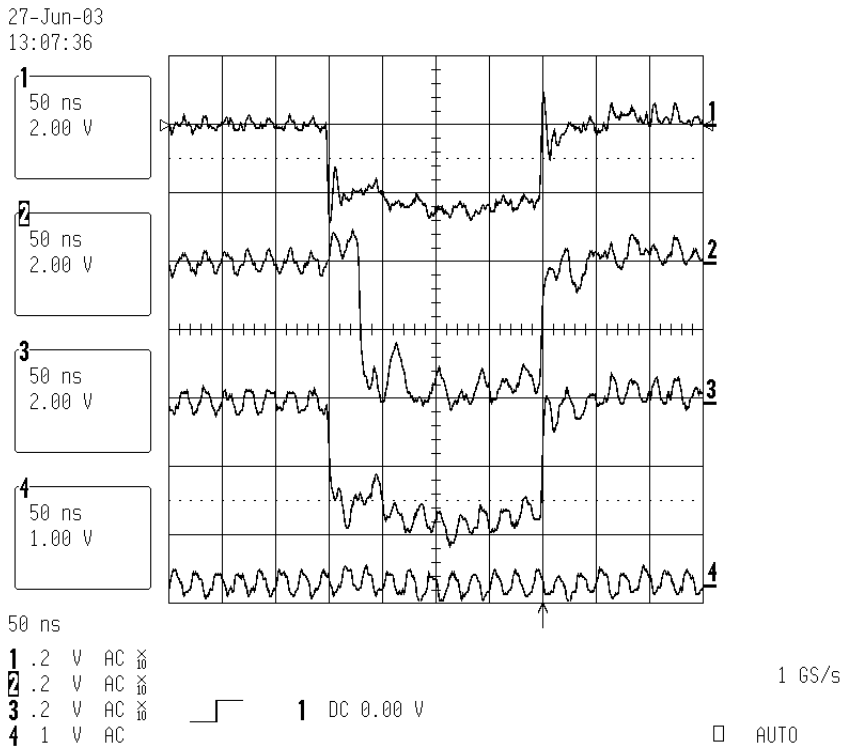


Figure D.3: 1=CS, 2=RD, 3=A0, 4=50 MHz Clock

Appendix E

Bill of Materials

Bill Of Materials for nibbler_R2_CAD_A.sch

Item	Qty	Ref-Des	Part Name	Manufacturer	Description
1	1	D9	6CWQ03FN		6CWQ03FN sch diod D-Pak
2	1	U3	8051-1	Winbond	Winbond 78E54 Firmware, 1.03.00 by Touch Int.
3	1	U4	ADS7846	Texas Instruments	Touch controller
4	1	C4	C+3528-2U2- 35V-18066		CAP 2.2UF 35V TAN LESR +-10% SMD B
5	1	C12	C+3528-2U2- 35V-18066		CAP 2.2UF 35V TAN LESR +-10% SMD B
6	1	C16	C+6032-10U- 20V-13596		CAP 10UF 20V TAN +-20% SMD C
7	1	C1	C+7343-10U- 35V-18070		CAP 10UF 35V TAN LESR +-10% SMD D
8	1	C2	C+7343-10U- 35V-18070		CAP 10UF 35V TAN LESR +-10% SMD D
9	1	C3	C+7343-6U8- 35V-18069		CAP 6.8UF 35V TAN LESR +-10% SMD D
10	1	C13	C0603-DUMMY		CAP SMD 0603 22pF
11	1	C14	C0603-DUMMY		CAP SMD 0603 22pF
12	1	L5	C0603-DUMMY		IND SMD 0603 BLM21A
13	1	L10	C0603-DUMMY		IND SMD 0603 BLM21A
14	1	L11	C0603-DUMMY		IND SMD 0603 BLM21A
15	1	L12	C0603-DUMMY		IND SMD 0603 BLM21A
16	1	C7	C1206-100N- 11397		CAP 100NF 50V CER X7R +-10% 1206
17	1	C8	C1206-100N- 11397		CAP 100NF 50V CER X7R +-10% 1206

18	1	C9	C1206-100N-11397	CAP 100NF 50V CER X7R +-10% 1206
19	1	C10	C1206-100N-11397	CAP 100NF 50V CER X7R +-10% 1206
20	1	C5	C1206-10N-1737	CAP 10NF 50V CER X7R +-10% 1206
21	1	C17	C1206-10N-1737	CAP 10NF 50V CER X7R +-10% 1206
22	1	C6	C1206-DUMMY	CAP SMD 1206 100nF
23	1	C11	C1206-DUMMY	CAP SMD 1206 100nF
24	1	R3	C1206-DUMMY	CAP SMD 1206 100
25	1	R4	C1206-DUMMY	CAP SMD 1206 100K
26	1	X1	CON-1X2STR-2.54MM-1094 1	PIN HEADER 1X2 STRAIGHT 2.54MM
27	1	X7	CON-1X2STR-2.54MM-1094 1	PIN HEADER 1X2 STRAIGHT 2.54MM
28	1	X8	CON-1X2STR-2.54MM-1094 1	PIN HEADER 1X2 STRAIGHT 2.54MM
29	1	X9	CON-1X2STR-2.54MM-1094 1	PIN HEADER 1X2 STRAIGHT 2.54MM
30	1	X10	CON-1X2STR-2.54MM-1094 1	PIN HEADER 1X2 STRAIGHT 2.54MM
31	1	X11	CON-1X2STR-2.54MM-1094 1	PIN HEADER 1X2 STRAIGHT 2.54MM
32	1	X4	CON-1X3STR-2.54MM-1062 1	PIN HEADER 1X3 STRAIGHT 2.54MM
33	1	X14	CON-2X13STR-F-2MM-XXXX	PIN HEADER 2X13 STRAIGHT 2MM
34	1	X15	CON-2X13STR-F-2MM-XXXX	PIN HEADER 2X13 STRAIGHT 2MM
35	1	X18	CON-2X13STR-F-2MM-XXXX	PIN HEADER 2X13 STRAIGHT 2MM
36	1	X21	CON-2X13STR-F-2MM-XXXX	PIN HEADER 2X13 STRAIGHT 2MM
37	1	X5	CON-2X3STR-2MM-17065	PIN HEADER 2X3 STRAIGHT 2MM
38	1	X3	CON-2X5STR-2.54MM-1089 6	PIN HEADER 2X5 STRAIGHT 2.54MM
39	1	X6	CON-DSUB9-ANG-18352	CONN DSUB 9P MALE R ANGLE 8.08MM

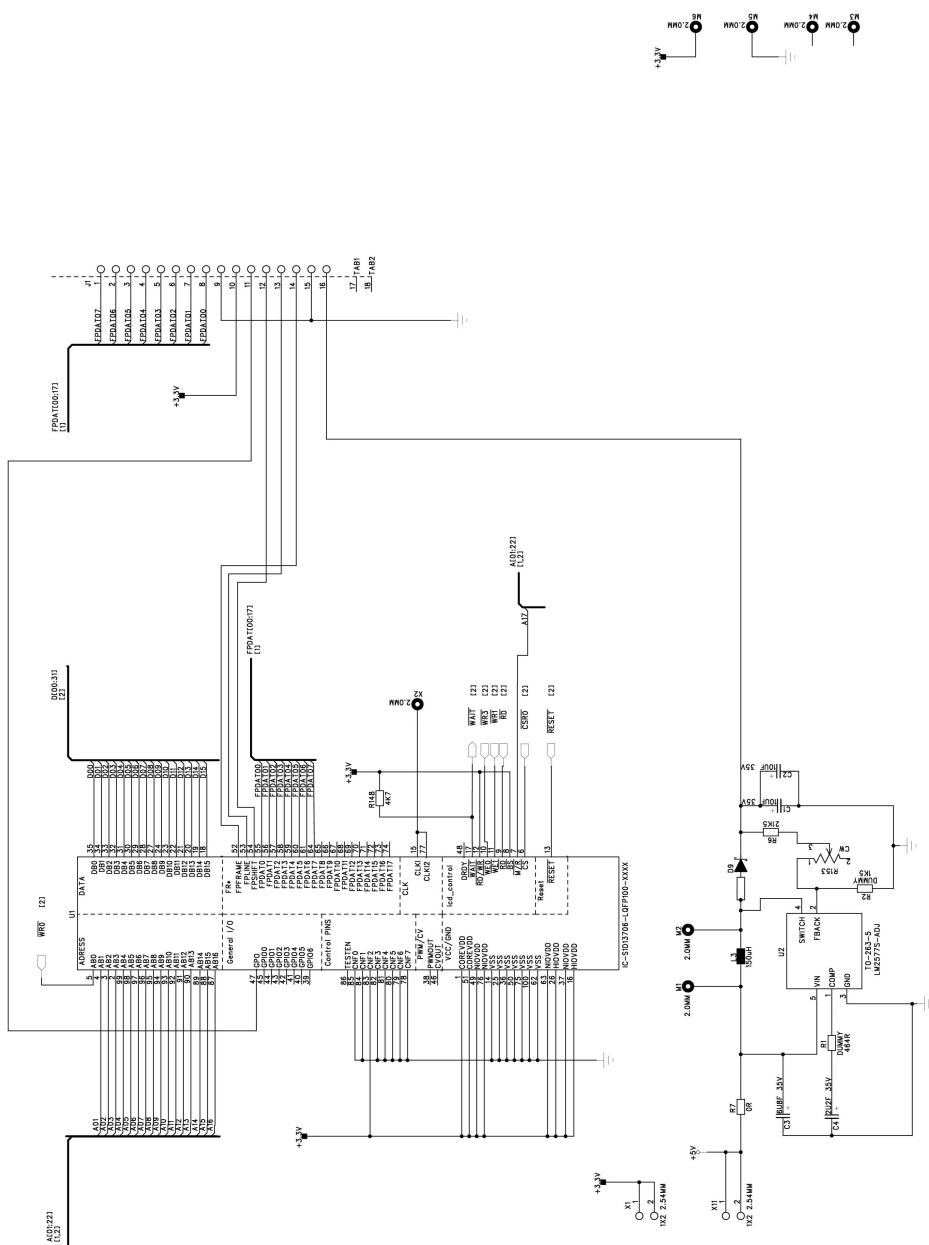
40	1	J1	CON-MOLEX-1 6P-ZIF	MOLEX	SMT,ZIF Connector 16 PIN
41	1	7	CON2X10- MOLEX	MOLEX	SMT,ZIF Connector 2X10 PIN
42	1	U5	IC-93C46-SO 8-19461		IC EEPROM 1024-BIT 93C46 8PIN SO
43	1	U8	IC-MAX3232- SO16-16000		IC 3232 DRIVER (RS232) SO16
44	1	U1	IC-S1D13706 -LQFP100-XX XX	EPSON	Epson S1D13706 LCD controller
45	1	L3	KM-4		150uH Inductor for Nationals 50Khz switch
46	1	LED1	LED-GREEN-S MD-ANG-1482 7		LED GREEN SMD
47	1	M1	PAD2_0		Pad with 2.0mm hole
48	1	M2	PAD2_0		Pad with 2.0mm hole
49	1	M3	PAD2_0		Pad with 2.0mm hole
50	1	M4	PAD2_0		Pad with 2.0mm hole
51	1	M5	PAD2_0		Pad with 2.0mm hole
52	1	M6	PAD2_0		Pad with 2.0mm hole
53	1	X2	PAD2_0		Pad with 2.0mm hole
54	1	R5	R0805-2K2-1 3116		RES 2K2 1/10W 1% 0805
55	1	R148	R0805-4K7-1 5612		RES 4K7 1/10W 1% 0805
56	1	R7	R1206-0R-12 035		RES 0R 1/4W 1% 1206
57	1	R6	R1206-21K5- 11608		RES 21K5 1/4W 1% 1206
58	1	R1	R1206-DUMMY		RES SMD 1206 464 ohm
59	1	R2	R1206-DUMMY		RES SMD 1206 1.5K
60	1	U2	TO-263-5	NATIONAL Sem.	LM2577S-ADJ switchreg National Semiconductor
61	1	R153	VRES-10K		VARIABLE RESISTOR (POT)
62	1	Y1	XTAL12		XTAL 16MHz crystal
63	1		Axis 82		Axis 82 developer board R1
64	1		LCD module	PalmTech	PMC39T01H
65	1		Inverter	TDK	CXA-L0505-NJL
66	1		Epoxy laminare		THICHNESS 1MM Double sided 100x160 MM

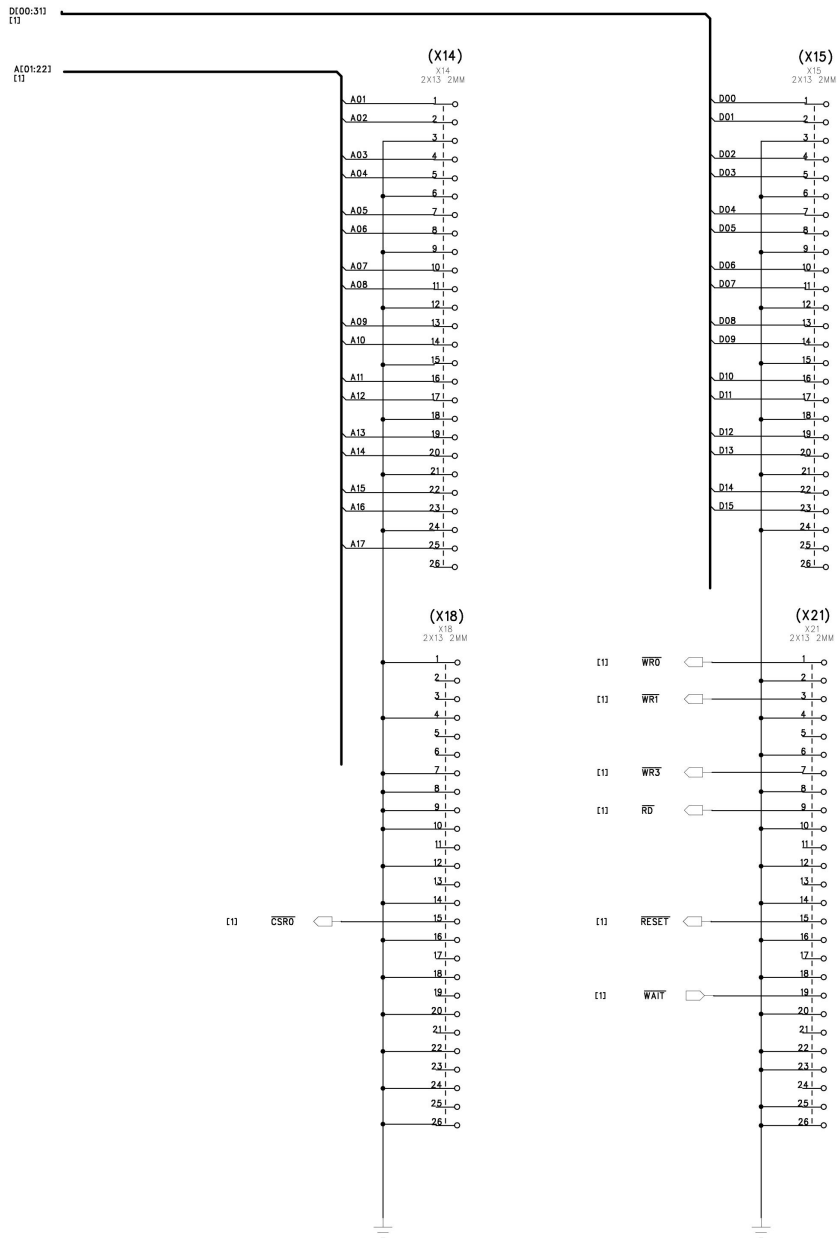
Appendix F

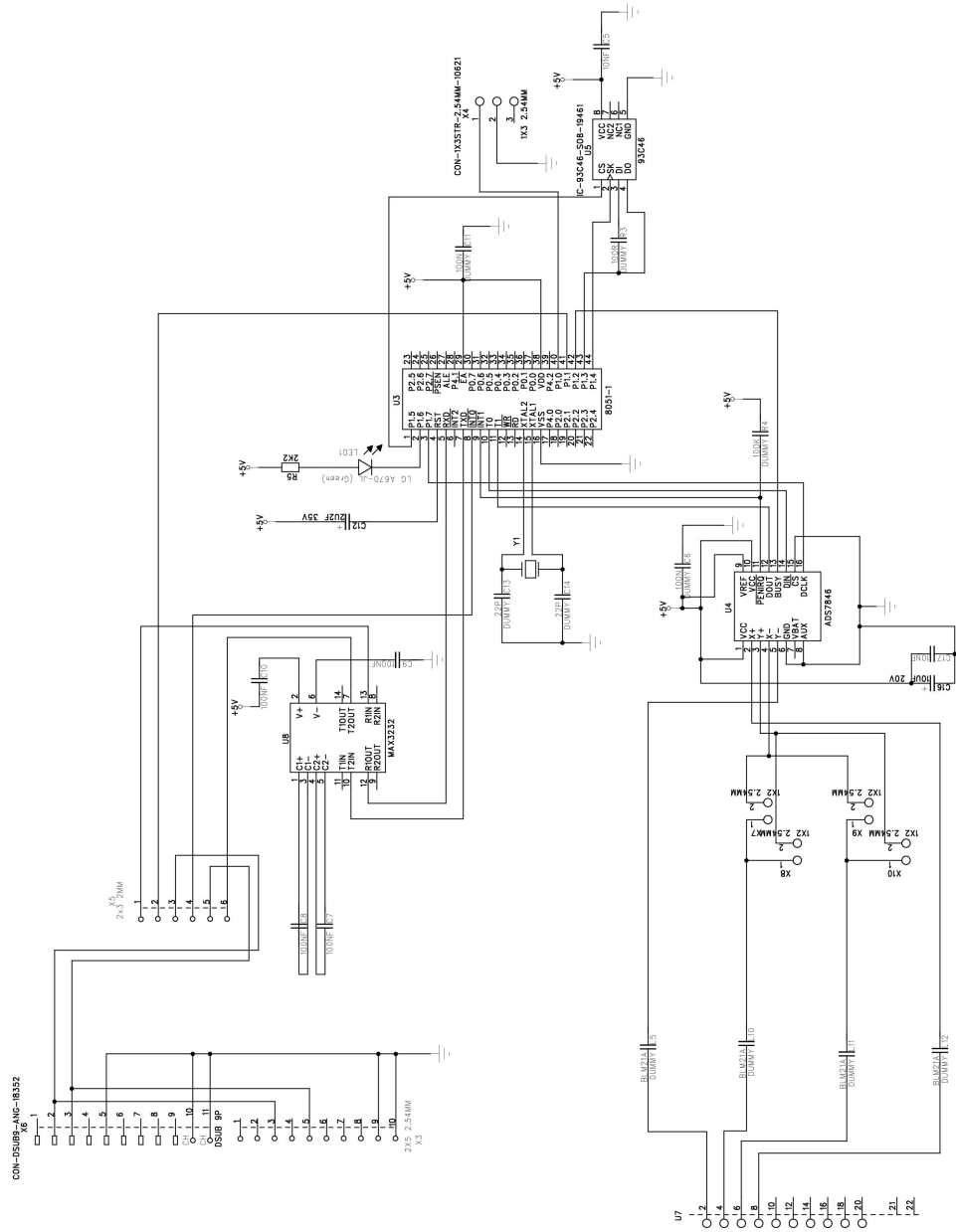
Circuit Diagram

Explanations to the circuit diagram on the following pages:

- The circuit board is connected to the system bus pin-list on the AXIS 82 Developer Board (s-side)
- R153 is the variable resistor for the contrast voltage (VEE).
- X4 - RS232 (1,3) - PS/2 (1,2)
- X5 - RS232 (1,3) and (5,6) - PS/2 (2,3) and (4,5)
- U7 - Touch panel interface contact
- J1 - LCD module interface connector







Bibliography

- [1] ACTE AB. Acte ab. Webpage, 2 July 2003.
<http://www.acte.se>.
- [2] ANVIN, P. Linux allocated devices. Webpage, 3 June 2001.
<http://www.lanana.org/docs/device-list/>.
- [3] ARIMA GROUP. Arima. Websida, 21 Aug. 2003.
<http://www.arima.com.tw/>.
- [4] ASILIAN TECHNOLOGIES. Asilient technologies. Webpage, 7 Aug. 2003.
<http://www.asilient.com/>.
- [5] ATMEL. Atmel. Webpage, 25 Aug. 2003.
<http://www.atmel.com/>.
- [6] AXIS COMMUNICATIONS. Axis 82 - developer board. Tech. rep., 13 Aug. 2003.
http://developer.axis.com/products/axis_82/index.html.
- [7] AXIS COMMUNICATIONS. Etrax 100lx designer's reference. Webpage, 13 Aug. 2003.
http://developer.axis.com/doc/hardware/etrax100lx/des_ref.html.
- [8] BELL, D. I. Nano-x programming tutorial. Websida, 3 Oct. 2000.
<http://www.microwindows.org/>.
- [9] CITIZEN. Citizen. Websida, 21 Aug. 2003.
<http://www.citizen.de/>.
- [10] EGEVO AB. Egevo ab. Webpage, 2 July 2003.
<http://www.egevo.se>.
- [11] ELECTRONICS, T. Tdk. Webpage, 2 July 2003.
<http://www.tdk.com/>.
- [12] EPSON RESEARCH AND DEVELOPMENT, INC. S1d13706 technical manual. Technical Manual, 17 Apr. 2001.
<http://www.erd.epson.com/vdc/html/contents/S1D13706.htm>.
- [13] EPSON RESEARCH AND DEVELOPMENT, INC. Epson research and development, inc. Webpage, 7 Aug. 2003.
<http://www.erd.epson.com/>.

-
- [14] FARNELL. Farnell inone. Webpage, 2 July 2003.
<http://www.farnell.se>.
- [15] HERVEILLE, R. Vga/lcd controller. Webpage, 7 Aug. 2003.
http://www.opencores.org/projects/vga_lcd/.
- [16] HITACHI. Hitachi. Webpage, 7 Aug. 2003.
<http://www.asiliant.com/>.
- [17] JAPANESE TECHNOLOGY EVALUATION CENTER. Display technologies in japan. Webpage, 20 Aug. 2003.
http://www.wtec.org/loyola/dsply_jp/toc.htm.
- [18] KGAA, M. Die geschichte der flussigkristalle. Webpage, 6 June 2001.
<http://pb.merck.de/servlet/PB/menu/1113370/index.html>.
- [19] KODAK. Webpage, 15 Oct. 2001. <http://www.kodak.com/US/en/corp/-researchDevelopment/technologyFeatures/display.shtml>.
- [20] KODAK. Oled. Webpage, 4 June 2003.
<http://www.kodak.com/US/en/corp/display/gen3Material.jhtml>.
- [21] KYOCERA INDUSTRIAL CERAMICS CORP. Kyocera industrial ceramics corp. Webpage, 13 June 2003.
<http://www.kyocera.com/>.
- [22] LEHMAN, J. Ezfb ~ linux frame buffer api. Webpage, 4 Aug. 2003.
<http://www.akrobiz.com/ezfb/>.
- [23] LIQUID CRYSTAL GROUP. Liquid crystal glossary. Webpage, 20 Aug. 2003.
<http://www.elis.ugent.be/ELISgroups/lcd/lc/glossary.html#EEE>.
- [24] MENTOR GRAPHICS. Mentor graphics. Webpage, 12 Aug. 2003.
<http://www.mentor.com/pads/>.
- [25] MOLEX INC. Molex. Webpage, 28 July 2003.
<http://www.molex.com/>.
- [26] NATIONAL SEMICONDUCTORS. Lm2577-adj. Webpage, 28 July 2003.
<http://www.national.com/ds/LM/LM1577.pdf>.
- [27] NATIONAL SEMICONDUCTORS. National semiconductors. Webpage, 28 July 2003.
<http://www.national.com/>.
- [28] PALM TECHNOLOGY CO., LTD. Pmc39t01h technical manual. Technical Manual, 09 Sept. 2002.
- [29] PALM TECHNOLOGY CO., LTD. Palm technology. Webpage, 22 Aug. 2003.
<http://www.palmtech.com.tw/>.
- [30] POWERTIP. Powertip. Websida, 25 Aug. 2003.
<http://www.powertip.com.tw/>.

- [31] RITEK. Ritek. Websida, 25 Aug. 2003.
<http://www.ritek.com.tw/>.
- [32] SCANCRAFT. Scancraft ab. Webpage, 2 July 2003.
<http://www.scancraft.se>.
- [33] SHARP. Lcd color palette generation. Note, 2 July 2003.
<http://sharpworld.com>.
- [34] SKIP OSGOOD, CK ONG, R. D. Touch screen controller tips. White Paper, 26 June 2000. <http://www-s.ti.com/sc/psheets/sbaa036/-sbaa036.pdf>.
- [35] TEXAS INSTRUMENT. Texas instrument. Webpage, 28 July 2003.
<http://www.ti.com/>.
- [36] TOUCH INTERNATIONAL. Touch international. Webpage, 29 July 2003.
<http://www.touchinternational.com/>.
- [37] TRG. Trg components. Webpage, 2 July 2003.
<http://www.trg.se>.
- [38] TRISCEND. Triscend. Webpage, 7 Aug. 2003.
<http://www.triscend.com/>.
- [39] TRULY. Truly semiconductors gmbh. Webpage, 2 July 2003.
<http://www.truly.de>.
- [40] VARITRONIX. Varitronix. Websida, 22 Aug. 2003.
<http://www.varitronix.com>.