NTNU
Innovation and Creativity

Open Digital Canvas

Nicolas Mendoza

Master of Science in Computer Science
Submission date: June 2007
Supervisor: Maria Letizia Jaccheri, IDI

Norwegian University of Science and Technology
Department of Computer and Information Science

Problem Description

The candidate will participate in a specific artistic project. The project consists of designing and implementing a system that will ensure that roughly 120 boards with 25 LEDS each together will form an artistic product. The project will be carried out by programming the specific boards, setting up a system to be able to control the boards individually and as a whole. The system should be controllable by third parties using an online input system, for intance a web page. The project work varies from programming low-level hardware to communication protocols on a TCP/IP level, and finally a high level system for third parties to communicate with the system. The project will be carried out together with selected parties at IDI already involved in the project. The result will be an electronic installation at ITV-054 for public display.

In addition, the candidate will have to ground his work and compare it to state of the art projects. Finally he will have to reflect on his own development process as well as on other challenges that will arise at the intersection between art and software, artists and technical parties.

Assignment given: 22. January 2007
Supervisor: Maria Letizia Jaccheri, IDI

Abstract

The candidate will participate in a specific artistic project. The project consists of designing and implementing a system that will ensure that roughly 120 boards with 25 LEDS each together will form an artistic product. The project will be carried out by programming the specific boards, setting up a system to be able to control the boards individually and as a whole. The system should be controllable by third parties using an online input system, for intance a web page. The project work varies from programming low-level hardware to communication protocols on a TCP/IP level, and finally a high level system for third parties to communicate with the system. The project will be carried out together with selected parties at IDI already involved in the project. The result will be an electronic installation at ITV-054 for public display.

In addition, the candidate will have to ground his work and compare it to state of the art projects. Finally he will have to reflect on his own development process as well as on other challenges that will arise at the intersection between art and software, artists and technical parties.

Table of contents

1. Acknowledgements

- **Åsmund Gamlesæter:** the initiator of the first collaborative project between the Faculty of Architecture and Fine Art and the Department of Computer and Information Science at NTNU.
- **Gunnar Tufte:** supervised students during Summer 2005. Gives good advice now.
- **Espen Gangvik:** Aesthetic consultant.
- **Anders Christensen:** co-supervisor of master thesis work. Technical responsible at IDI.
- **Letizia Jaccheri:** supervisor of this thesis' author.
- **Anna Trifonova:** researcher.
- **Salah Uddin Ahmed:** researcher.
- **Dag Svanæs:** gives god advices.
- **Hallvard Trætteberg:** advice on GUI ideas.
- **Olav Gulling:** developed PCB circuits and schematics for the HW at IDI.
- **Magnus Andersen**, **Guttorm Aase** and **Marius Grannaes** made some substantial efforts after Olav finished his summerjob and went to Africa.
- **Norvald Ryeng:** technical consultant, maintainer of Samfundet installation
- **Gunnar Lien:** lending out equipment to program boards
- **Øystein Lysholm** built prototype mounter in wood

2. Introduction

Open digital canvas will be a canvas covering an entire wall for which hardware, software, and behavior will be as open as possible. We wish to push the concept of openness in software a step further.

The goal of this project is to embellish a white wall with something that will tell us about our identity (hardware and software) and that will make us discuss and reflect about it. This project starts with the aim to reuse the hardware and software produced for the installation http://www.livingwall.org/.

It will hopefully bring us together and create curiosity and admiration among students and visitors. The digital canvas will be used in the context of Jaccheri's Experts in Team village and will evolve through student projects and master theses.

There are a set of artistic, technological, and research challenges which will contribute to render the open digital canvas development process a source of inspiration and reflection around interdisciplinary education and research.

The canvas will consist of a number of mainboards with LEDs on forming a big matrix of light pixels. The work to get the project done involves building the physical installation, designing software and hardware solutions to communicate with the lights on the boards from a server and for third party users.

The wall will be one of the electronic installations at NTNU. This work will contribute to make explicit other technology installations.

3. Goal of the project

The goals of the project vary from contestant to contestant, but mainly a physical installation with corresponding software and harwarde installation that ensure it is accesible by students and staff at IDI, NTNU is the immediate goal. Further research is possible for instance by using the system to visualize input, explore input and communication methods or investigate reuse and extensions of the system.

In addition the process itself of carrying out the project, designing the software, investigating possibilities and gaining knowledge by designing the system and comparing it with other similar projects are of main interested to us.

3.1. Research goals

By carrying out this project we want to investigate the methods of designing an open source project, learning by doing in regards of designing the system, software and hardware-wise and compare it to existing similar art-projects.

Lastly the project is intended to be open, both in openness of use of software licenses and in making the system accessible and in the degree of incluence third parties, developers and users can influence the system

4. Open Source Software

Open Source Software is the term coined for software utilizing and licensed under an Open Source License. The OSI (Open Source Initiative) works as the governing body of deciding which licenses are considered to fit under the category OSS. There are a plethora of Open Source licenses, among those licenses like the MIT, BSD, Apache, GNU GPL, GNU LGPL, Mozilla, Creative Commons, Mozilla, Qt and many more.

The purpose of Open Source Software and free software licensing is to permit and encourage the involvement by licensees in improvement, modification, and distribution of the licensed work. The adoption and the movement based on this effort is driven by its development methods, like distributed work, cooperation, public source repositories, iterative development and the like. In addition different kinds of compensations and non-traditional economics based for example on other value than monetary are common.

4.1. History

The origin of the Open Source movement can be traced as far back as the 60s and 70s. When AT&T ended up developing the MUTLTICS succesor, named Unix. Source code was exchanged between the original developers, the Berkley University and other interested parties, like DARPA, CGSR, BSDI and regular developers. Bugfixes and enhancements knew little boundaries as it was handed back and forth to improve the various systems that grew out of the original operating system. The BSD license originates from this era, and it permitted licensees, which meant anyone aquiring the software, to do whatever they wanted with the software as long as credit was due, in addition at this time there was a clause that said that the original source of the derived work had to be mentioned in advertising material. This clause was later removed in 1999.

It wasn't till the late 70s and 80s that the code and the derived products started to seem lucrative for AT&T and their lawyers started to act accordingly. The growing differences culminated in the 90s where AT&T, Berkley University and BSDI took matters thru court.

In the meantime Richard M. Stallman experienced personally the annoyances of not being able to alter source code of the products his work acquired, supposedly a driver for a paper jam ridden Xerox printer was refused to be disclosed by Xerox, meaning that they could not be able to fix the problem themselves -- like they were used to in many other areas. He left MIT in 1984 to form the FSF (Free Software Foundation), trying to formalise certain rights, or needs, a piece of software would come with to be able to be really "free". That means how software could come with as the liberty given to the user to use it in any way she seeked. It should be possible to run the software for any purpose, it should allow you to see how it works and change it to fit your needs and it should be freely re-distributable, for a fee or none, verbatim or in a modified form.

It wasn't uncommon that software came with all these criteria fulfilled, but he saw that such an enforcement couldn't last, ie. nothing prevented a person to use free software inside his own products

without publishing the changes or additions he had made. To be able to license software in a way that made the software itself, also in modified form, retain it's original license, a "viral" clause was added. This was the foundation of the GPL and the ideas of making an operating system that was truly free, GNU (GNU's not UNIX).

The Open Source Initiative (OSI) was started as a reaction to the Free Software Foundation and its problems reaching out to the conservative business. Raymond and Perens set out to form OSI defining Open Source as a new term bsaed on FSF's initiatives. They created a guideline that open source licenses should adhere to be labeled as such. The guidelines were based on the previously defined Debian Free Software Guidelines [PERENS1999]. They were criticized for using the term open source which was watered out, but one of the arguments in favour was that the term free" used by FSF had a much worse duality in that it could mean free of charge in addition to the meaning that is used in free software.

The guidelines set out contained a set of criteria that an open source license should meet. They focused on that the licenses should allow redistribution, modification, be non-discriminatory, source code should be available, licenses must not be restricted to aproduct or a brand, licenses should not contaminate other software and more. [OSIDefinition].

4.2. Methodology

The Open Source movement has not only influenced the world of software development with its licenses and sharing of code, but also in the methodologies used when collaborating on projects and sharing software. Due to its distributed nature, variety in work methods and team sizes several methodologies have been utilized and gained popularity in traditional software circles. Public source repositories, mailing lists to discuss development, the use of versioning control systems and other collaborative tools have been massively used in the world of OSS and are increasingly common throughout the world.

4.3. Licenses

As discussed there, are a plethora of OSS compatible licenses. They fit different purposes and have evolved over time. The OSI is the governing instance defining which licenses are to be considered as OSS compatible. A quick overview follows.

4.3.1. The MIT/X, Apache, BSD, Academic Free licenses

The MIT/X and BSD licenses are some of the earliest open source licenses. They are easy and straight-forward and date back to the above mentioned era. They basically give all rights further to the licensees and pose no restriction on relicensing and redistributions other that attribution clauses and that no warranties are given. As the names imply they are designed by the respective universities as needed. They are very permissive for the licensee as they forward most of the rights held by the original copyright holder.

The **BSD Licenses** consist of quite a few variations of the original BSD license. The various BSD based Unix ditributions use the licenses with various clauses. The main BSD license states that:

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the ORGANIZATION nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

The original BSD License, in addition stated that advertisement should include the original copyright holder, this worked fine as long as only the University of California was named, but when third parties used and re-used the license for derivative works, the list of copyright holders accumulated and led to unacceptable terms. The advertisement clause was finally rescinded by the Director of the Office of Technology Licensing of the University of California on July 22 1999. [BSDLICENSE]

Also a warranty disclaimer is added, where the copyright holder seeks to not be applicable for any express or implied warranty claims. This is both in regards of merchantabilty warranties and other not explicitly stated warranties.

The **MIT or X11 License** is almost identical to the main BSD license except it doesn't have a no-endorsement clause and is shorter and somewhat less detailed in its form. [MITLICENSE]

The **Apache License** comes in two flavours, version (1.0) 1.1 and version 2.0. The original 1.0 is very similar to the main BSD license, but it also requires the acknowledgment of the creator's contribution to the work being distributed both in advertisement and documentation. Th1 1.1 version removed the need to also include acknowledgments in advertising material. It also states that products derived from the code shall not use the name "Apache" withouter prior permission from the apache foundation.

The 2.0 version is a re-write of the license to be able to be slightly more compatible with GPL, in addition to being more explicit and detailed. It was released as late as january 2004. The license permits to be referenced rather than included in every file. It differentiates between derivative works that are meant to be part of the original product and independent derivative works. It also mentions explicit the rights of patents granted and the termination of the license to the licensee if she initiates litigation against any contributor to the work carrying this license. Although similar to the GPL in the way derivative works are to be licensed, the license still allows third parties to re-license the work undeer a different license. [APACHELICE]

The **Academidc Free License** operates in a similar fashion to that of the Apache license, but it adds some additional provisions. It makes it clear that rights to pantents owned and utilized by the Licensor are also granted to the licensee. It states explicit that the work may be displayed or performed in addition to the regular uses. All source code must contain previous notices about copyright, pantent claims and trademarks. It also states that if a licensee makes a derivate tive that it should use best efforts to get an assent from the licensor to use the work, but if not possible, the license terms still govern. The license contains several paragraphs regardin potential legal cases, including stating that and cases filed should be

done in the area the licensor operates, that reasonable attorney of the prevailing party shall be covered and a "merger clause" stating that the agreement of the license supersedes prior agreements. The Acadamic free license also does explicitly state that no trademarks are granted. The license has at several revisions, and the current one is version 3.0, the author of the license doesn't recommend usage of versions lower than 2.1. [ACADLICENSE]

4.3.2. The GPL, LGPL and Mozilla licenses

The following licenses differ from the previous ones in that they seek to let the software licensed "survive" modifications and redistribution. As mentioned the GPL was formed by the Free Software Foundation as a reaction to the fact that changes to BSD and similar licensed code could remain in-house, integrated in proprietary code effectively hindering evolvement of the code itself. Licensees benefitted from getting pre-made software without having any obligations themselves. The GNU Public license seeked to cure this problem.

The **GPL License**, or GNU General Public license, is probably the most known open source license. IT was designed with the users of open source in mind, letting the rights follow the software. All derivative works shall be licensed with the same license, giving it a viral characteristic. It ensures that additions and modifications are also made public (if parts of the work are distributed). It is still possible to use GPL-licensed work in.house, possibly only distributing the code to selected few which will not further distribute code, thereby making it closed in practice. The license is greatly more detailed than the previous licenses mentionned. Source code of derived works may be distributed in different ways, one of which includes offering the source code with a included written notice for a fee no more than what the physical work to put forward the code may cost. This offer doesn't need to last more than three years though, so again there is a danger of code being closed after that period is over. GPL license also has the same warranty disclaimers as the others, and also terminates the license if any breach occurs, resulting in the code being governed by original copyrights.

The license itself has to be included with the software and has to be provided unmodified, which measn one can't make derived licenses out of it. Any software distributed together with GPL licensed software forming a package will be tainted by the license, resulting in the licensor to also license the accompanying software as GPL. However if there is significant distinction between the parts, and the parts are distributed separately this does not apply. For further details it is advised to refer to [OPENS2005] and [GPLLICENSE]

The **LGPL License**, or GNU Lesser General Public, is as the name suggest a less strict license, or as the FSF states a license that gives the user less rights. Formerly known as the GNU Library General Public License, the license allows for dynamic linking, and posibly static linking with the LGPL licensed software without tainting the software that is linking to it. It has been mostly used to allow proprietary and non-GPL-licensed code to be able to link with code which is similar to GPL in its policy. Modified LGPL code has the same restrictions in that it has to keep the same license. [LGPLLICENSE]

The **Mozilla Public License** was a result of Netscape Communications releasing their binary code and subsequently their source code to the public in 1998. The license seeks to let the original licensor keep rights to relicense code and use contributed code as their own. It is written in a strong corporate fashion obvious for instance in the way it has a solid list of definitions and tries to clearly define the various parts affected by the license. It is seen as a mixture of BSD and GPL license in that it allows others to use the code within proprietary projects, however modified code will also automatically be licensed under the MPL.

The license also differs from previous licenses in that termination of the license grant does not happen immediately after breach, but allows for a 30 days "cure" period to rectify any violations. Similar to the GPL, termination of grants do not apply to sublicensees. [MOZLICENSE]

4.3.3. The Qt, Artistic and Creative Commons licenses

The Qt and Artistic licenses are similar to the Mozilla Public in that they are originally designed for use with a specific software in mind. The Qt license designed by Trolltech was made for their Qt Toolkit and software utilizing it, while the Artistic license was intended for modules made with and additions and modifications to Larry Wall's Perl interpreter. The Creative Commons licenses are a set of licenses intended for use with non-software works created by artists and the like.

The **Qt license** is similar to the GPL license in that it requires derivatives to also be licensed under the same license, with the exception that it allows distributions of *patches*, independent pieces of software suitable to be applied to existing sotware. Such patches shall not remove copyright notices or in other means remove licenses or notices that the original software needs to carry. This exception has it pros and cons. The original developer is ensured that his work remains intact, but at the same time allowing third parties to add or change behaviour. However it might add a burden to contributors in that they are not ensured that their changes become part of the original software, they need to ship patches separately, and these issues also might make it harder for ed-user to use the software as patching and compiling might be necessary. In addition the Qt license defines a venue for disputes and governing law regarding the license. Although the QPL is pretty permissive, it has some limitations that doesn't make it GPL-compatible, due to pressure from the community, mainly the KDE user and developer base, Trolltech agreed to cross-license their toolkit which KDE is based on. [QTLICENSE]

The **Artistisc license**, or Perl Artistic license, was meant for distribution of perl modifications, changes and modules. [CPAN]. The difference between the Perl version of this license and the Artistic License is that the Perl one allows for an additional option for commercial distribution. The license, perhaps due to its early introduction, is criticized for its vagueness. IT operates similar to previously mentioned licenses, but it also allows re-distribution of modified code if names are changed and the "Standard version" is also made available with its corresponding code. The "Standard version" term is one of the confusing terms used, as it's not clear exactly what that means, wether it is the original non-modified code or any altered version with fixes or modifications. It also states that publicly freely available code may be incorporated and be considered as a Standard version. This raises some questions as publicly available code still have

the author's implicit copyrights attached, unless the author explicitly retracts all rights to it.
[ARTISTLICE]

The **Creative Commons licenses** are a result of the non-for-profit organization Creative Commons trying to set up an open source license platforms for creative works. The licenses are not intended for software, and the organization strongly discourages people from using their clearly written licenses for software and instead recommend using one of the Open Source licenses listed at OSI. They are worth mentioning though as they are gaining momentum in areas where people submit and publicize creative works, including blogs, documentation, images, photos music and other creative work. Their licenses consist of atomic parts that one can put together to describe the rights granted to users of creative works, these include attribution, non-commercial use, forbidding derivative works, share alike (retaining original rights) etc.

5. License choices

As we've seen there are a lot of different open source licenses out ther, the OSI lists 58 of them, which are all approved to be proper open source licenses. [OSILICENSE]. To choose a particular license is not easy. However as the most common licensesin use are the GPL, LGPL and BSD variations, people also tend to choose those mostly because they are familiar with them. In addition there are less troubles when a piece of software is to be used or shipped together with operating systems or packaes licensed with those licenses. The APL and AFL licenses are not too uncommon either, but they are used mostly with software made with or for the same systems that the licenses were originally designed for.

This project is meant to be used by staff and students at IDI, NTNU. Staff and future students should be able to modifiy and make additions to code bot for experimental purposes, to fix bugs and add new features. In addition the project code might be used with other not related projects and given that NTNU has relations with commercial companies that might be interested in engaging NTNU students to use the project and possibly evolve into commercial projects.

We're not able to predict which uses the project may or may not be used for, the project might need to be able to change license, or there might be a desire to utilize bits or pieces of the project in commercial and proprietary systems. Substantial parts of the firmware code used on the various boards are originally BSD, before altered to fit the boards need.

Given those facts, we wish to license all software produced under ths project with a two-clause BSD license which goes like this:

```
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

The two-clause license does not include the no-endorsement restriction that might have rendered the code unsuitable for a future GPL license. As the no-endorsement clause imposes a limitation that is not compatible with the free nature of GPL. This also allows for later licensing as a three-clause BSD license if wanted. The reason for picking BSD and not MIT or APL in this case is mostly since the license is much shorter and more known than the alternatives. One might not that the fairly known operating system FreeBSD uses a two-clause BSD license for all new code and kernel code [BSDTWOCL] and that the only BSD license that the KDE projects accepts for its libraries is a two-clause BSD license. [KDEBSD]

6. Source code retrieval process

The cards which the wall consist of run a firmware developed for previous projects, this code was aquired early on in the project from Norvald Ryeng which is de-facto leader of the Samfundetwall project. He in turn received the code from Åsmund Gamlesæter which was part of the group that made the original wall. [Appendix A]

One challenge was that the firmware made for the motherboards didn't have an explicit license and that there was given an impression that the authors of the source code weren't necessarily interested that it should be available for others. There was no problem in getting access to the code thru parties involved in the Samfundetwall though. In addition it turned out that most of the code was verbatim code of BSD licensed code available on the net.

Luckily after querying Åsmund Gamlesæter, we got permission to license the original code as BSD License, thereby making the entire project open source [Appendix B]. All of it which is licensed under a two-clause BSD-license, with the exception of the PHP extension which is licensed under a PHP license. It should be noted that we were asked to refer to the Ethernut project found at http://www.ethernut.de and that other parts of the source are from the uIP project and some from an individual with the name Paul Hills. More about this can be found in the repository itself in the file ./avr/README

Documentation was poor however after some hands-on experimentation and input from parties like Per Anda and Øystein Handegard we mastered the code and was even able to fix a bug. The bug was basically that when a given light was lit, the light on the above row was also lit slightly, this was crucial to get rid of to properly take advantage of using each individual light as an individual representation. This is yet another factor that should make it likely that a final permission will be given to licence the code with an open source license.

The samfundetwall has an own project based on python with its own repository, however this is mainly controller software to run animations and plot graphics on the wall, that project is run amongst others by Stein-Magnus Jodal. A patch and instructions on how to fix the bug was sent back to the involved parties. However to properly fix the bug one needs to re-flash all the firmware on all cards, so to my knowledge the patch hasn't been rolled out on their system

7. Related projects

7.1. Living wall

The Living wall project was the first project using the same boards as Open Digital Canvas. It is a permanent installation at Studentersamfundet i Trondheim. It reacts on sound and display graphica accordingly.

The project was led by Åsmund Gamlesæter and the firmware code used in Open Digital Canvas is a slightly modified version of the one used in the Livingwall boards.

The following is quoted from their web site: http://www.livingwall.org

> *Living Wall is an ambient installation collecting, recomposing and playing sonic memories. The computational processes that take place are displayed on four architectural scale LED matrices with a total number of 3000 white superflux LEDs.*
>
> *The installation has microphones recording fragments of human interaction. Each new recorded fragment is analysed using an adaptive sound categorization technique, determining its relation to previously stored clips.*
>
> *Based on this analysis, a network of sound clips is constructed by the installation in real time. The network of recorded clips is constantly reorganized as new fragments are recorded and connections between previously stored clips are elaborated. The goal of the self-organization is to maintain a topology where fragments with similar perceptual qualities are grouped in clusters and dissimilar clips are separated from each other. As time goes by the network evolves into a rich mass of information with complex interconnections.*
>
> *Recomposed sequences of sound are played back into the space by walking through the network of sonic memories. The sounds are distributed into the space based on a mapping from the network position to a position in the room. We regard this behaviour as being parallel to the behaviour of how we associate ideas.*
>
> *Living Wall is a permanent installation at Studentersamfundet in Trondheim, Norway, made specifically for UKA-05. The work was partly conceived in the studios of PROJEKT0047 in Berlin. Research, technological development and physical production was done in Trondheim with help from the Department of Computer and Information Science at Norwegian University of Science and Technology, and at the Art & Technology program at the IT University of Göteborg.*
>
> *Living Wall is the first installation in a series of pieces. Please read about the second work in this series RED Screen.*

7.2. Samfundetwall/Lysfontene

The Samfundetwall project continued the Livingwall project by making a system allowing to blit images and animations on the same boards used. The system was made by Øystein Handegard and Terje Wiesener for UKA-05. This system has been lately replaced by a system made in Python, which is administrated by Stein Magnus Jodal. Norvald Ryeng is one of the persons administrating the installation itself and has come with tips on various issues, including how to mount the boards physically in a practical way.

More information about the Lysfontene project can be found at the ITK wiki on Studentersamfundet's web pages: http://itk.samfundet.no/dok/Lysfontene

7.3. Blinkenlights

Project Blinkenlights is related to Open Digital Canvas in that it is a system that uses a similar approach of illustrating images and animations. It was carried out as a project byt the German Chaos Computer Club in 2001 as a celebration of its 20th birthday. They transformed a building on Alexanderplatz in Berlin to a giant monochrome low reoslution display by using each of the windows as a pixel in an image or animation. In addition one could interact with the giant screen using a cell phone, for instance to play pong.

The software used has been released as GPL and is available here: http://www.blinkenlights.de/blinkentools.en.html

In addition a similar installation was made by a group called CCC in the national library of France in Paris. The installation was called Arcade, and had a higher reoslution and several shades of grey.

The Blinkenlight project has given birth to other related projects and to a set of tools and applications that use the blinkenlights format to display animations and images on a set of devices and interfaces. Many of which are available in the above-mentioned link.

7.4. Cooperation

Given the similarity of the related projects listed, some cooperation seemed to be appropriate. Initially the Blinkenlights format used was interesting to study and to possibly implement. Also their software and tools could've been utilized however, when concentrating on getting the ODC system to work that aspect was neglied, however being able to parse the format was kept in mind throughout the project and is still possible without too much additions.

Access to the Lysfontene repository was granted early on in the project, however being locked to one language and not having the same goals as ODC, we didn't get involved that much except observing the occational commits and noting the progress (or lack of).

The firmware changes done for ODC and fixes for the light bug were offered to the Lysfontene-participants, however it is not apparent that they have been utilized yet.

The format of the UDP protocol to communicate with the boards that had been documented by the Lysfontene members were taking into use and become very useful before we started digging into the firmware code itself and explore the inner workings of it.

The hardware programming which we had difficulties with at first was resoled with help from one of the Lysfontene members. In addition we got a quick demonstration and overview of how the system was mounted physically and how its current software worked.

In hindsight one could question the lack of cooperation between the Lysfontene project and the ODC project as they both have similar goals, namely to be able to interact and present regular computer images and present them on an almost identical hardware system. However we woul still think that the ODC provides more possiblities as it is designed with concurrency and abstraction in mind.

8. Project setup

8.1. Wiki

When doing software projects involving several participants (and also one-man projects), a set of tools to distribute information, keep track of revisions, and store information is needed. A set of tools were set up early in the project, all accessible from http://odc.opentheweb.org.

When starting off the project we quickly set up a Wiki, a collaborational web site to be able to store information related to the project. The information includes plans, architecture, ideas, meeting summaries, road maps, links to external resources, pictures and more. Sadly it was only one or two persons that actually contributed with info, but hopefully it is useful as a resource for people interested in the project wanting to keep up with what was going on.

8.2. Blog

A blog was set up initially but was taken down after the introduction of the Wiki which allows one to blog and store information by any partcipant in an easy fashion.

8.3. Mailing lists

A mailing list was set up to be able to quickly push out information and discuss matters related to the project. An invitation to the mailing list was sent out to all interested parties, but only three or four members actually signed up and only two or three persons actually used the list. Mailman is a stte of the art mailing list system with easy to use interfaces to sign up, alter settings an unsubscribe from mailing lists. In addition it provides an online archive to keep track of messages and allows for easy transition in case of future wishes to move the project from the current host.

8.4. Versioning system

To keep track of code revisions, history, commits and having a public area to download code, a Subversion repository was set up. In addiiton to different web frontends are available for it. Subversion was chosen as it is the de-facto standard for versioning systems compatible to CVS, but fixing many of CVS shortcomings.

All code is accessible from the repository available at http://odc.opentheweb.org/svn including firmware code, test suite code, daemon code, plugin and library code and related build instructions, documentaton and license notices. It also contains this thesis. Anyone can inspect or download the code, both in the future and during the project this spring.

Commit logs are sent to odc-commit@lists.opentheweb.org for people interested in keeping track of changes.

9. Hardware architecture and physical mounting

The boards that the wall consist of are 30cm x 30cm with 25 LEDs each evenly distributed in both axis, they are spaced in such a manner so that the outermost lights get the same spacing to an adjacent boards' outer lights than between the lights on the same board. Purposely to let a set of boards assembled together to work as a coordinate system, or as a canvas of pixels.

Each light can have a light intensity with around ten levels, yet the firmware uses a range between 0-99 decimal. The ODC APIs designed though lets you set levels in range between 0 and 255, or 0.0 and 1.0 for convinience.

A physical canvas is supposed to consist of a set of these boards and are to be mounted together using some kind of framing. The front plates that were used on the Red Screen project where the boards originate from are not available for use indoors though, so an alternative approach has to be used.

The process of mounting the boards have stagnated though, luckily early in the process the idea of making a prototype to be able to test and actually see how such a canvas might look and behave arised. Øystein Lysholm from the technical staff at NTNU helpes us make a wooden structure to be able to mount 6 boards on. This 3x2 is available at one of the offices in ground level of the IT building on NTNU.

Having 110 working boards a set of physical layouts are possible. The proposed area to display the canvas is a wall measuring 630cm x 225cm. This leaves at least the possiblity for the following configurations (total amount of boards in parenthesis): 10x11 (110), 9x12 (108), 8x13 (104), 7x15 (105), 6x18 (108), 5x22 (110), 4x27 (108), 3x26 (108), 2x55 (110). Althought at most 20 boards will fit in the width 7 boards in the height of the wall. There seemed to be a consensus that 12x8 boards was an ideal configuration leaving 14 boards available for later exchange or as backup boards.

There are several challenges regarding the light emitted. It's pretty strong and focused. And has it pros and cons both in daylight and in a dark room. In addition as one can see from the simulator on http://odc.opentheweb.org/Simulator it's not very aesthetical to mix a red-pinkish focused light with the green color of the silisium board, so some sort of diffusing filter in front of the lights, and maybe black plates to cover the boards themselves would be an idea.

9.1. Framing

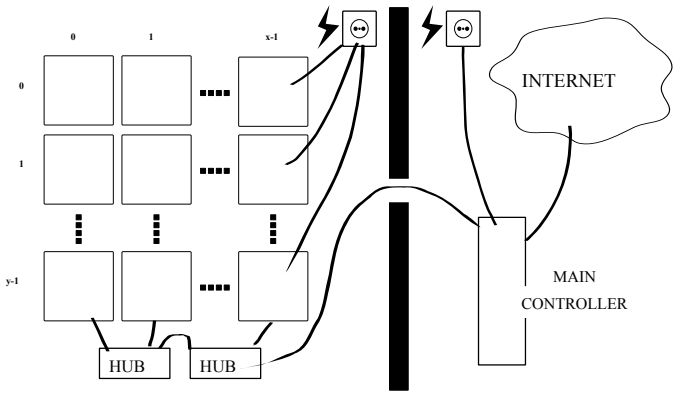
There were several proposals in the air about the frame the board should be mounted on. It shouldn't be too heavy so that it can be mounted on the wall, and the wall could resist the weight. There should space behind the boards for the cabling the Ethernet socket of the boards and one or more powers supply units.

In addition it isn't decided wether the server hardware managing the boards and a switch to connect the boards to the server should reside beneath the wall, behind the wall, or on the other side of the wall, making a big enough hole thru the wall necessary. In addition the room behind the wall is used as a study and work area for students.

The boards should be easily accesible for repair and update and boards mounted high should have enough cabling to pull them down to convinient work height. There were proposals to make a frame that allowed to swing out the canvas or parts of it for maintenance.

9.2. Hardware and cabling

Overview over Hardware architecture



Each board is basically an own computer in the sense that it has an own power input, an RJ45-connection and a CPU to control the lights. In addition they have memory on board. A main controller computer is needed to be able to communicate with all the boards.

At the hardware level this can be carried out by connecting all the boards with TP cables to a given amount of hubs or switches, depending on availabilty, price and space and power requirements.

Some of the hardware requirements are listed below.

1. The computer will typically need two network interfaces, one to communicate with the boards thru the hubs, and one to be reachable online.
2. It should be possible to restart the computer easily, but not by anyone, so some kind of locking mechanism would be nice.
3. Power usage and similar will have to be calculated for the boards, and someone should ensure that there is enough power available in either side of the wall.
4. The actual cabling of the boards to hubs and order physical, shoudn't really matter, except maybe for minimal latency and tidyness. All boards are to be adressed by IP and/or MAC address and their order should be configurable from the main controller using a simple LUT.

This requirements are covered as follows

1. Such a setup is available at the same office mentioned above, where the wooden structure prototype operates.
2. The software designed is made so it runs as a daemon on a Linux system, thereby starting at boot. Also possible to restart remotely via SSH or similar.

3. Although not perfectly calculated the power supply currently in use should be enough for at least 150 boards. However one needs to make branches of cables since the space for connection to the PSU is somewhat limited. A set of big cables that branch out to smaller ones suitable to connect to the boards exist together with the boards.
4. The software addresses the board in an agnostic manner, the direction a board is mounted doesn't matter and neither does the order, as long as the config file of the daemon states the position and direction. Two helper test commands are available for this purpose, one displays an arrow on all boards pointing to the cardinal directon that goes into the config file (NE,NW,SE,SW), and the other displays a number on every board corresponding to its last digit of their decimal IP address (or MAC address for that matter) using parts of the LEDs. They are located on the 3x5 rightmost LEDs, the remaining two columns are used as kind of an abakus showing one lit led per tenth place of the ip address. For instance board with IP address 10.0.10.24 will have a number 4 on the right side and two lit LEDs on the left side. 10.0.10.1 will have a number 1 on the right and no lit LEDs. This works up to 109, when a board with higher ip exists it will have the first LED off, and so forth making a total of 209 adressable boards in this manner.

10. Software architecture

During our initial meeting it seemed clear that the main objective when designing and implementing the system architecture is to let it be as open as possible. This to easily get adopters of the system when in a functional state, to be part of the open source area, to be able to use existing open software and to use existing and proved APIs and interfaces to able to communicate with other systems as easy as possible. In addition the system will be openly documented making it easy for third parties to know more about the architecture or re-implement it themselves.

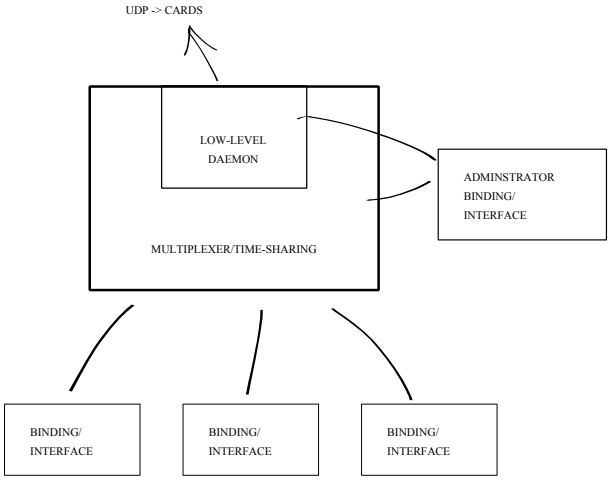
Some of the criterias proposed where:

> *When the system is ready, it should leave at least one interface to control the hardware from a high level language or interface. Wether, how and where multiplexing is done should probably be decided as the project evolves. It was suggested to do an incremenal development moving one layer at the time away from the low-level, starting out with crude putpixel/setpixel stye of operations, possibly convert a bitmap to those, a clear screen. These using a "human" mapping using cartesian coordinates, for instance from top-left. Check out the simulator source code for some examples on functions.*
>
> *Each card runs an own TCP/IP stack which is already implemented and kind of working. Rumour has it, one needs to ping the cards to keep them alive though.*
>
> *To communicate with the cards one can send UDP packets of a certain format to set the various lights to various levels. There is some latency and light levels are somewhat buggy, so one needs to take care of that. For instance it seems lights really only have values between 0x0C and 0x64 (not 0x00 -> 0xFF), there should be a mapping from (0,0 -> 1,0) and/or (0 -> 255) somwhere in the system. Again, check the simulator for inspiration.*
>
> *The main controller needs to run a daemon that translates crude operations to actual UDP packets, as mentioned wether this daemon allows some sort of multiplexing by assembling a set of instructions on a given timeslice, or applying incoming paint commands to a current screen should be decided upon. It could be really simple though and just do exactly what it is told, no matter who does that, and apply some ACL and time-sharing/multiplexing at one layer higher.*
>
> *It would be nice to eventually be able to display any kind of image on the wall, by providing some common interface or framebuffer that is widly supported. This could be anything from an opengl context, to an sdl screen to a java2d canvas or similar. It would even be possible to use the Simulator SVG implementation for instance using XHR to communicate with it, but worst case, a simple c library that anyone can make bindings to from any language should at least be provided.*
>
> *Many of these issues probably need to be solved by experimenting or talking to people that have done similar projects. I have already been in touch with most of the people that participate and are still working with the wall at Samfundet.*

I have got access, although still not looked at the libari and martha code for the "lysfontene"-project. The project is hosted in a bazaar repository. In addition I've had discussions about the wall with Norvald Ryeng, Stein Magnus Jodal and Øystein Handegard.

It was evident from the start that one goal for the system should be that it was open, as the name implies. Open in various senses. The installed system should be accesible physically, but also reachable for communication and manipulation. The software and hardware should be open as in licensed with open source licenses and be properly documented. It should also be easy to hook up to using various programming languages or interfaces. A sketch on how the system could be designed was presented and the participants seemed to agree that it looked like a sensible approach.

Overview over software architecture



The architecture was planned to be a daemon that runs contantly listening to incoming connections. Connecting clients would be able to send commands to the daemon which translated the high level commands to UDP communication with the corresponding boards. The daemon should have an internal representation of the canvas and its state at any given time, although this would slightly clash if the boards where turned of and the server remained on. This, since there is no feedback functioality programmed in the firmware of the cards. A heartbeat solution would only pollute the network, specially on high traffic when updating a canvas rapidly, and it's easy to restore the canvas state on the boards by telling the daemon to update them all.

10.1. Concurrency and scheduling

The system should allow concurrent access by talking to the server's port. There is no limit to actual clients. There was never decided on how to actually multiplex incoming draw commands. However it's possible to both prioritize special users and hosts or combine commands rom various clients overlaying the output graphic.

As of now the server runs commands as is, the socket listener process commands as they are received,

10.2. Priority

An admin user should have more priority than a normal user, an admin user will typically accessing the daemon locally. This can be extended to several levels or using host based priorities.

10.3. Access rights

Basically two levels of access, admin access and user access. Admin connects locally thru a unix socket, normal users use tcp socket or corresponding libraries. One can deny access using the OSes own functionality or using internal filtering.

10.4. Access methods

The system should allow for use of various programming languages using open protocols or means of communication.

11. Software implementation

c++, lib, socket, binary communication, apache (web server, xhr etc. avoiding supporting HTTP/1.1)

11.1. Simulator

Early in the project it became clear that being able to visualize the inal product would be of great benefit. Specially since there were many discussions around how to layout the boards to fit the wall and also what kind of material etc. should be used.

A simulator using SVG and HTML was made being able to configure the layout configuration of the canvas by changing the amount of boards in width and height. In addition one could draw and upload pictures to the simulator to get a feeling on how the wall would work.

The simulator is available from http://odc.opentheweb.org/wiki/Simulator. Note however that although it did work fine in Mozilla Firefox earlier, it doesn't seem to work perfectly at the moment, please use another SVG capable browser like Opera 9.21 to be able to exploit all its features.

11.2. Test suite

When getting to know the boards, rather than starting on a full-fledged system a test suite was created to communicate directly with the boards. These are very simple perl scripts, although, it eventually turned into a perl module for easier devlopment of new tests.

11.2.1. canvas_test.pl

run a couple of random tests

11.2.2. config.txt

config file positioning boards in the canvas the format is:

```
<width> <height>
<ip> <direction>
<ip> <direction>
...
<ip> <direction>
```

width
- amount of boards in width

height
- amount of boards in height

ip
- ip adress of board

direction

- SW,NW,SE,NW, to be able to not force all boards to be placed in a certain way, this soutwest, northwest, southeast, northeast indicates where the inscription "JOVE-JVE-SI" points. down-left equals southwest.

11.2.3. cube_test.pl

An unrealistic rotating cube, unfortunately this never worked properly as it should've been a rotating cube, but there were other matters more important to carry out than to fix the calculations. However it gives one an idea on how animations work.

11.2.4. line_test.pl

This test tries to draw an anti-aliased line, but is not fully working.

11.2.5. single_card_anim.pl

This test takes an ip adress as a parameter and reads from stdin an animation or still picture in the format described below.

```
Example: cat ../all.txt | ./multi_card_anim.pl 10.0.10.1 10.0.10.2
```

11.2.6. multi_card_anim.pl

This test takes a set of ip adresses as input and reads from stdin an animation or still picture in the format described below.

```
Example: cat ../all.txt | ./multi_card_anim.pl 10.0.10.1 10.0.10.2
```

11.2.7. numerize.pl

This test numerizes all boards present. It draws a number on each board based on its IP address. As described earlier, this is used to locate boards in a canvas, and to figure out which IP a board has. The last number is printed in full on the right hand side, and the multiple of tens is denoted by a lit LED on the left hand side.

11.2.8. ODC.pm

This is a perl module used by most of the tests that handle communication to the boards and have a set of primitive draw functions.

11.2.9. show_image.pl

Shows an image feed in as 8bit greyscale values:

```
convert ../ntnulogo_15x10.png gray:- | ./show_image.pl
```

11.2.10. Test animation format

Each line represents an image, the first number is the amount of time to hold the following image, the next x amount of numbers represent the image itself using numbers between 0 and 99 separated with space. If not enough parameters are sent to cover all 25 LEds, the rest will be treated as 0 values.

```
Example: this animation first lights all 25 lights, then lights a dot starting from first
pixel to the 25th pixel with decreasing amount of time.

$ cat ../traverse.txt
0 99 99 99 99 99 00 00 00 00 00 99 99 99 99 99 00 00 00 00 00 00 00 00 00 00
10.5 0 99
1.5 0 0 99
1.5 0 0 0 99
1.4 0 0 0 0 99
1.3 0 0 0 0 0 99
1.2 0 0 0 0 0 0 99
1.1 0 0 0 0 0 0 0 99
1.05 0 0 0 0 0 0 0 0 99
1.04 0 0 0 0 0 0 0 0 0 99
1.03 0 0 0 0 0 0 0 0 0 0 99
1.02 0 0 0 0 0 0 0 0 0 0 0 99
1.01 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
```

11.3. Daemon

Basically the system consists of a daemon running on a computer that communicates with low level UDP commands with the boards. This daemon will run constantly accepting commands from clients and transform them to proper UDP packets sent to the corresponding boards.

Any amount of clients can connect to the server communicating using a low-level protocol. A client connects thru a listening TCP socket on the server, authenticates and starts sending commands and receiving results. The low-level raw protocol was chosen to be able to send commands as fast as possible, compared to say an XML based format that many RPC frameworks rely on. This has the disadvantage that communicating with the daemon is hard doing by humans or standard tools, however the library designed provides a set of functions that one can hook up to using C or any other language supporting dynamically loading and executing ELF symbols. Provided is an example of how to use PHP, argueably the most common web application language in use, to communicate with the daemon.

The daemon is made in C++ and comes with accompanying instructions to be built with make. It installs as a regular /et/init.d/-script and allows to be started automatically at boot.

The daemon listens for incoming connections on port 5001 and converts the raw protocol format command from a library to corresponding UDP commands and sends them to the correct boards. It keeps an internal state of the board LEDs' statuses to allow a client to retrieve the current value of a given coordinate.

The daemon is designed to be able to authenticate users based on both username and password and based on hostname connecting from. The protocol format is badly documented right now, but it's being worked on, the command set of the raw protocol is found in ./include/odc.h

The daemon is able to handle concurrent connections from arbitrary clients, as of now it executes commands to the boards as they are received from clients, but some sort of queue is fully possible.

11.4. Libraries

A library has been made to communicate with the daemon, or any daemon for that matter. The library exposes high-level functions for its users to be able to light LEDs, get LED values, clear entire canvas and update. The latter is used

The library made in C provides the functions listed in ./include/odc.h, the PHP functions available are listed in ./php-extension/odc.def

The PHP extension is made in C and links to the library, thereby providing a set of PHP functions that allows one to connect and access an arbitrary instance of the ODC daemon. This makes it easy to make a web frontend that allows one to communicate with the server. An example of this should be running on http://odc.opentheweb.org/frontend

Build instructions for all of the above are available in the repository or attached to this document.

11.5. Board firmware

The board firmware runs on an AVR system and is made in C. It require a special set of corss compiler to compile for the target.

The source has been altered to be able to compile with the current version of gcc-avr, it has gotten some more convinient build targets, and in addition a bug that made a LED be lit slightly, one row above the LED intended to be lit. A patch is avaiable in the listing benath as ./avr.diff

11.5.1. Programming boards

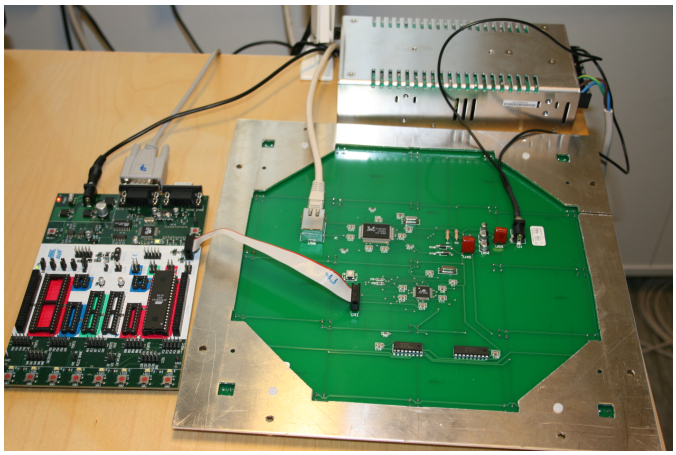
To program the boards, one needs a device to interface from a computer to the board itself.

The board has a 10 pin jtag interface. To connect it use a device like a JTAGICE, JTAGICEMKII or STK500.

Unfortunately we were unable to get communication working with the JTAGICEMKII borrowed from Gunnar Lien at IDI. Luckily after some help and verification with Norvald Ryeng, We got to borrow an STK500 from Gunnar Lien, which worked fine.

11.5.1.1. Physical setup

Communication setup with board



1. Connect power to board (7.5V DC), board should autamatically boot and all LEDs should be turned on, and off again.
2. Connect STK500 with serial cable to a computer
3. Connect power to STK500 (10-15V DC)
4. Turn on STK500, wait for STATUS led to go from RED, YELLO and finally turn GREEN.
5. Connect 10 pin flat cable from ISP10PIN connector to the board's 10 pin connector.

11.5.1.2. Communication with AVRDude

- Install AVRDude on modern deb based systems by doing: `sudo apt-get install avrdude`

```
$ avrdude
Usage: avrdude [options]
Options:
  -p <partno>                 Required. Specify AVR device.
  -b <baudrate>               Override RS-232 baud rate.
  -B <bitclock>               Specify JTAG/STK500v2 bit clock period (us).
  -C <config-file>            Specify location of configuration file.
  -c <programmer>             Specify programmer type.
  -D                          Disable auto erase for flash memory
  -P <port>                   Specify connection port.
  -F                          Override invalid signature check.
  -e                          Perform a chip erase.
  -U <memtype>:r|w|v:<filename>[:format]
                              Memory operation specification.
                              Multiple -U options are allowed, each request
                              is performed in the order specified.
  -n                          Do not write anything to the device.
  -V                          Do not verify.
  -u                          Disable safemode, default when running from a script.
  -s                          Silent safemode operation, will not ask you if
                              fuses should be changed back.
  -t                          Enter terminal mode.
  -E <exitspec>[,<exitspec>] List programmer exit specifications.
  -y                          Count # erase cycles in EEPROM.
  -Y <number>                 Initialize erase cycle # in EEPROM.
  -v                          Verbose output. -v -v for more.
  -q                          Quell progress output. -q -q for less.
  -?                          Display this usage.

avrdude project: <URL:http://savannah.nongnu.org/projects/avrdude>
```

Make sure you have write access to serial device on computer (`id -a` should include `dialout`)

Run for instance with:

```
avrdude -c STK500v2 -p m32
```

This means, use avrdude with an ATMEGA32 chip using a STK500 v2 device.

```
$ avrdude -c STK500v2 -p m32
avrdude: AVR device initialized and ready to accept instructions
Reading | ################################################## | 100% 0.04s
avrdude: Device signature = 0x1e9502
avrdude: safemode: Fuses OK
avrdude done.  Thank you.
```

11.5.1.3. Compiling firmware

11.5.1.3.1. Toolchain setup

To compile the firmware of the boards a set of tools are required. Specifically a cross compiler to compile binaries that are able to run on the designated hardware. A cross compiler is a compiler that makes binaries for a different platform than the one the compiler runs on. In this particular case, we used Ubuntu GNU/Linux to develop on. With its Debian-based APT package manager the only thing needed to get the proper toolchain installed is the following command:

```
sudo apt-get install gcc-avr avr-libc
```

This is the GNU Compiler collection's C compiler with the AVR platform as target. In addition a c library is needed.

11.5.1.3.2. Compilation

The AVR code used in this project comes with a Makefile making everything easy to compile. After a few bugfixes added for GCC and a newer AVR C library than the one used originall, one can simply use "make" in the correct dir to build both flash and eeprom images to use with avrdude. Modified code is available in the repository of this project.

To only write an existing EEPROM and flash file use: `make program`. To just write an existing EEPROM file, use `make eeprom`.

The latter is used like this to easily reprogram the IP address and MAC address of a board:

```
touch uipopt.h && CFLAGS="-DUIP_IPADDR0=10 -DUIP_IPADDR1=0 -DUIP_IPADDR2=10 -DUIP_IPADDR3=6"
make && make program
```

uipopt.h is the file containing definitions of IP address, netmask, default gateway and MAC address. We have added code so that the IP address can be passed from the command line, and the MAC address is generated based on the IP address. Sine the uipopt.h file is not set as a dependency in the Makefile, one needs to touch the uipopt.h file to recompile everything that relies on it, to make sure the new address is propagated to the corresponding object files.

With the above example a board will get the IP address `10.0.10.6` and MAC address: `0A:0A:0A:00:0A:06` (0x0A == 10). With 0A as the first value, we ensure that the second last significant bit of the most significant byte is set to indicate that we have an internal address.

12. Analysis

12.1. Process

It's fair to say that the process of the project hasn't been ideal. Despite the setup of communication channels and a kick-off meeting with much initiative, there was hard to get contribution and feedback on proposals and ideas. There is still no proper installation setup and few decisions made on the actual setup of the installation. However the daemon and client should be working nicely, although missing some features, all of which are easy to append and fits perfectly for an open source project that this has now become.

We would have to admit the lack of communication on our own part has contributed to the lack of progress overall, in addition not being physically available at the designated working area at all times during office hours probably contributed to this.

There were litt if any proper risk assesment documents, detailed design documents and the like. Much because the waters explored were unknown but also due to lack of dicipline and understanding.

For our own matter the lack of a project leader and carved out reponsibility areas and tasks has played in both in lack of motivation and confusion about the tasks having to be carried out.

Nevertheless, we believe the final software product and its continuance and this document will prove useful for both the staff and students at IDI, NTNU.

12.2. Comparison with other projects

In hindsight one might also conclude that possibly a tighter cooperation with the Samfundetwall project would've been beneficial. Although that too includes risks, having more than one person to design a project and program it helps both in planning and programming stages.

However the design with a daemon running, clients being able to connect and people being able to use a library to link with has more in common with a database than more direct approaches like the Livingwall, Samfundetwall or Blinkinglights projects.

12.3. Software design

Some might point out that simply providing a library with high level drawing functions, that still are custom and uncommon might not justify the goal of being open. For instance hooking up directly to common graphic libraries like SDL, OpenGL, DirectX or piggy-backing on the Blinkenlights tools might have been better.

Nonetheless, there is nothing against using the library to provide a framebuffer device or access thru the above-mentiond APIs like demonstrated with the php extension in this system.

The use of the various techniques, languages, APIs, ranging from hardware debugging and programming, to high-level C++ programming, to playing with sockets, web technology, vector graphics, and implementing a server-client solution from scratch, in addition to also get aquianted to an extension API has proven very educational, possibly one of the most educational part of the degree in regard of hands-on experience.

The end result of an open source project with all its regular means with web page, code repository, mailing lists ready for the diestion of the public is quite satisfactory.

In the spirit of open source, the project still has potential for improvements and we will continue its development. Hopefully other people interested will join too.

13. Bibliography

[PERSP2005]

Perspectives on free and open source software / edited by Joseph Feller ... [et al.], Feller, Joseph, c2005, Cambridge, Mass., MIT Press, ISBN:0-262-06246-1, ib, XXXI, 538 s. ill.

[OPENS2005]

Open systems and standards for software product development / P.A. Dargan, Artech House computing library, Dargan, P.A., c2005,, Boston : Artech House, ISBN:1-58053-749-9, ib., XVII, 289 s. ill.

[DOING2005]

Doing your research project : a guide for first-time researchers in education, health and social science / Judith Bell , Bell, Judith, 2005, Maidenhead : Open University Press, ISBN: 978-0-335-21504-1, 4th ed., XV, 267 s. fig.

[UNDER2004]

Understanding open source & free software licensing / Andrew M. St. Laurent, St. Laurent, Andrew M., 2004, Beijing : Sebastopol, Calif. O'Reilly,ISBN: 0-596-00581-4, h., XII, 193 s.

[THESU2004]

The success of open source / Steven Weber, Weber, Steven, c2004, Cambridge, Mass. : Harvard University Press, ISBN: 0-674-01292-5, ib.

[HANDB2001]

Handbook of action research : participative inquiry and practice / edited by Peter Reason and Hilary Bradbury, Reason, Peter, Bradbury, Hilary, 2001, London : Sage, ISBN: 0-7619-6645-5, ib., XLII, 468 s. ill.

[BSDLICENSE]

http://www.opensource.org/licenses/bsd-license.php

[MITLICENSE]

http://www.opensource.org/licenses/mit-license.html

[APACHELICE]

http://www.apache.org/licenses/

[ACADLICENSE]

http://www.opensource.org/licenses/academic-license.php

[GPLLICENSE]

http://www.opensource.org/licenses/gpl-license.php

[GPLLICENSE]

http://www.opensource.org/licenses/lgpl-license.php

[MOZLICENSE]
http://www.mozilla.org/MPL/MPL-1.1-annotated.html

[QTLICENSE]
http://www.opensource.org/licenses/qtpl.php

[ARTISTLICE]
http://www.opensource.org/licenses/artistic-license.php

[OSILICENSE]
http://www.opensource.org/licenses/alphabetical

[BSDTWOCL]
http://www.freebsd.org/copyright/freebsd-license.html

[KDEBSD]
http://techbase.kde.org/Policies/Licensing_Policy

[OSIDEFINITION]
http://www.opensource.org/docs/definition.php

[PERENS1999]
http://perens.com/Articles/OSD.html The Open Source Definition, Voices from the Open Source Revolution by Bruce Perens, ISBN: 1-56592-582-3

14. Online resources

14.1. Software

[SHIFTREG]
http://homepages.which.net/~paul.hills/Embedded/ShiftReg.c (app.c seems to based on this)

[UIP]
http://www.sics.se/~adam/uip/ uIP (BSD)

[UIPAVR]
http://www.laskater.com/projects/uipAVR.htm (uIP-AVR)

[AVRLIBC]
http://savannah.nongnu.org/projects/avr-libc/ AVR Lib C documentation

[MACADR]
http://en.wikipedia.org/wiki/Mac_address Wikipedia - MAC address

[PHPEXT]
http://devzone.zend.com/article/1021-Extension-Writing-Part-I-Introduction-to-PHP-and-Zend

14.2. Hardware

[UDN2981A]
http://futurlec.com/Others/UDN2981A.shtml UDN2981A

[A6275EA]
http://www.ortodoxism.ro/datasheets/allegromicrosystems/6275.pdf A6275EA

14.3. Related

[SAMFARI]
Samfundet wall, Bazaar repository: http://itk.samfundet.no/dok/ari

[SAMFLYS]
Samfundet wall, Lysfontene: http://itk.samfundet.no/dok/lysfontene

[SAMFDIO]
Samfundet wall, DiodeVegg: http://itk.samfundet.no/dok/DiodeVegg

[LEGOPRI]
Lego print: http://www.storedyret.com/cgi/ikonboard.cgi?act=ST;f=17;t=30504

[LIVINGW]
Living wall: http://www.livingwall.org/

[REDSCRE]
Red Screen: http://www.ntnu.no/trestykker/redscreen/

[BLINKEN]
Blinkenlights: http://www.blinkenlights.de/

[LIBCACA]
Libcaca: http://libcaca.zoy.org/

[AALIB]
AAlib: http://aa-project.sourceforge.net/aalib/

14.4. References and tutorials

[BEEJ]
http://beej.us/guide/bgnet/output/html/multipage/index.html

[MYSQLPROT]
http://www.redferni.uklinux.net/mysql/MySQL-Protocol.html

[CPLUSPLUS]
http://www.cplusplus.com

[CPPREF]

http://www.cppreference.com/

[CPPFAQ]

http://www.parashift.com/c++-faq-lite/

14. Appendix A

```
From: "Norvald H. Ryeng"
To: anders@idi.ntnu.no
Subject: Diodevegg-kode

Her er tre .rar-filer vi fikk fra Åsmund Gamlesæter i sommer/høst.
Legger også ved den koden som kjører i Bodegaen på Samfundet nå. (I
tillegg til den vedlagte koden, har vi en cron-jobb som  pinger alle
platene hvert minutt e.l. for at de ikke skal slutte å ta imot data.)

Det vi har laget av skriftlig dokumentasjon finner du på
http://itk.samfundet.no/dok/lysfontene. Hardware-problemene og
-løsningene er dessverre bare dokumentert i hodet mitt.

Kontaktadresse på Samfundet er itk-lysfontene@samfundet.no.

Norvald
```

15. Appendix B

```
From: "Åsmund Gamlesæter"
To: "Nicolas Mendoza"
Subject: Re: Livingwall-firmware-lisens

> Hepps, holder på å skrive ferdig en oppgave basert på kortene som ble
> til
> overs etter Livingwall-prosjetet som IDI har og lurer i den anledning
> nøyaktig på hva slags lisense koden (avr.rar) har. Jeg ser at ting
> bruker
> både uIP og noe annen kode for å håndtere to chip-er som snakker med
> LED-ene. Jeg tror begge er enten BSD eller public domain. Jeg har drevet
> og laget en tjener-implementasjon som vil ende opp som en to-klausul
> BSD-lisens (så åpent som mulig uten å være public domain i grunn) og
> lurer
> på om jeg kan få lov til å lisensiere koden i avr.rar som det samme.

Hei Nicolas,

Det er OK for min del. Fint om du referer til Ethernut prosjektet
http://www.ethernut.de/


Hilsen
Åsmund
```

15. Appendix Z - Source code and build files

15.1. Repository contents

The contents and listings of this repository is accesible in an electronic form at http//odc.opentheweb.org/svn/ or http//odc.opentheweb.org/websvn/. The following is a listing of work done, ranging from Makefiles, perl snippets, a perl module, a php extension, a client library made in C, a Daemon made in C++ and init files for daemon and config files.

```
./avr:
app.c     compiler.h  INSTALL  Makefile  program    rtl8019.h   uip_arp.c
uIPexample_hex.aps  VERSION.h
app.h     delay.c     LICENSE  nic.c     README     uip_arch.c  uip_arp.h
uip.h               VERSION.zip
binary.h  delay.h     main.c   nic.h     rtl8019.c  uip_arch.h  uip.c      uipopt.h

./avr/program:
eeprom.eep  flash.hex

./daemon:
etc  INSTALL  LICENSE  Makefile  README  src  TODO

./daemon/etc:
init.d  odc.xml  odc.xsd

./daemon/etc/init.d:
odcd

./daemon/src:
Board.cc  Canvas.cc  Client.cc  Color.cc  Comm.cc  Daemon.cc  main.cc   modules
pack.h        xmlReader.h
Board.h   Canvas.h   Client.h   Color.h   Comm.h   Daemon.h   Makefile  pack.c
xmlReader.cc

./daemon/src/modules:
auth.h  auth_htpasswd.cc

./include:
odc.h

./legacy:
avr.rar  DiodeVegg.tar.bz2  LystavleController.rar  test3.rar  uip-0.9.tar.gz

./lib:
INSTALL  LICENSE  Makefile  README  src

./lib/src:
Makefile  odc_main.c  test.c

./php-extension:
config.m4  config.w32  INSTALL  LICENSE  odc.c  odc.php  php_odc.h  README  tests

./php-extension/tests:
001.phpt

./test:
all.txt    directions.txt  LICENSE            ntnu_logo.gif  smiley.txt  traverse_fill.txt
clear.txt  INSTALL         ntnulogo_15x10.png  README         src         traverse.txt

./test/src:
canvas_test.pl  cube_test.pl  multi_card_anim.pl  ODC.pm         single_card_anim.pl
config.txt      line_test.pl  numerize.pl         show_image.pl
```

15.2. Contents

Contents generated using:

```
svn export http://odc.opentheweb.org/svn/ odc-export;
cd odc-export
```

```
for a in `find -type f | grep -v "\./avr/" | grep -v "\./thesis" | \
egrep -v "*\.(png|zip|hex|eep|prj|pws|cvsignore|html|gif|bz2|rar|gz|aps)$"`; \
do echo "<h4 class=\"no-num\">$a:</h4><pre>";cat $a | \
perl -e 'use XML::Parser::Expat;  $parser = new XML::Parser::Expat; \
while(<>) { print $parser->xml_escape($_);}'; echo "</pre>"; done
```

./simulator/digitalcanvas.svg.php:

```
<? header("Content-type: image/svg+xml");


   $BOARDS_X = (isset($_GET['boards_x']) && preg_match("/\d+/
",$_GET['boards_x']))?$_GET['boards_x']:18;
   $BOARDS_Y = (isset($_GET['boards_y']) && preg_match("/\d+/
",$_GET['boards_y']))?$_GET['boards_y']:6;
   $LIGHTS_X = (isset($_GET['lights_x']) && preg_match("/\d+/
",$_GET['lights_x']))?$_GET['lights_x']:5;
   $LIGHTS_Y = (isset($_GET['lights_y']) && preg_match("/\d+/
",$_GET['lights_y']))?$_GET['lights_y']:5;

   $WALL_X = 630;
   $WALL_Y = 225;

   $FRAME_SIZE = (isset($_GET['framesize']) && preg_match("/\d+/
",$_GET['framesize']))?$_GET['framesize']:30;
   $BOARD_SIZE = (isset($_GET['boardsize']) && preg_match("/\d+/
",$_GET['boardsize']))?$_GET['boardsize']:26;
   $LIGHT_SIZE = (isset($_GET['lightsize']) && preg_match("/\d+/
",$_GET['lightsize']))?$_GET['lightsize']:1;

   $OFFSET_X = (isset($_GET['offset_x']) && preg_match("/\d+/
",$_GET['offset_x']))?$_GET['offset_y']:(($WALL_X-($BOARDS_X*$FRAME_SIZE))/2);
   $OFFSET_Y = (isset($_GET['offset_y']) && preg_match("/\d+/
",$_GET['offset_y']))?$_GET['offset_y']:(($WALL_Y-($BOARDS_Y*$FRAME_SIZE))/2);
   $STEP = (isset($_GET['step']) && preg_match("/\d+/",$_GET['step']))?$_GET['step']:0.3;
   $STEPTIMEOUT = (isset($_GET['steptimeout']) && preg_match("/\d+/
",$_GET['steptimeout']))?$_GET['steptimeout']:100;
   $ORIGIN_TOPLEFT=1;
   $ORIGIN_BOTTOMLEFT=2;
   $ORIGIN = (isset($_GET['origin']) && preg_match("/\d+/
",$_GET['origin']))?$_GET['origin']:$ORIGIN_TOPLEFT;

   $YRES = ($BOARDS_Y*$LIGHTS_Y);
   $XRES = ($BOARDS_X*$LIGHTS_X);


?>
<svg
   xmlns:svg="http://www.w3.org/2000/svg"
   xmlns:xlink="http://www.w3.org/1999/xlink"
   xmlns="http://www.w3.org/2000/svg"
   version="1.0"
   viewBox="0 0 <? echo "$WALL_X $WALL_Y"; ?>"
   width="<? echo "$WALL_X"; ?>"
   height="<? echo "$WALL_Y"; ?>"
   id="digitalcanvas"
   onload="main()"
   >
  <style type="text/css">
    svg { background-color: #eeeeee; }
    .board { fill: #007700; stroke: grey; stroke-width: 2; }
    .boardoff { fill: grey; stroke: grey; }
    .light { stroke: #000000; fill: #ff2222; fill-opacity: 0; stroke-opacity: 0.02;
stroke-width: 4;}
    #info { font-size: 10px; }
  </style>

  <script type="text/javascript">
  <![CDATA[
   BOARD_SIZE = <? echo $BOARD_SIZE; ?>;
   FRAME_SIZE = <? echo $FRAME_SIZE; ?>;
   LIGHT_SIZE = <? echo $LIGHT_SIZE; ?>;
   BOARDS_X = <? echo $BOARDS_X; ?>;
   BOARDS_Y = <? echo $BOARDS_Y; ?>;
   LIGHTS_X = <? echo $LIGHTS_X; ?>;
   LIGHTS_Y = <? echo $LIGHTS_Y; ?>;
   OFFSET_X = <? echo $OFFSET_X; ?>;
   OFFSET_Y = <? echo $OFFSET_Y; ?>;
```

```
   STEP = <? echo $STEP; ?>;
   STEPTIMEOUT = <? echo $STEPTIMEOUT; ?>;
   ORIGIN_TOPLEFT=<? echo $ORIGIN_TOPLEFT; ?>;
   ORIGIN_BOTTOMLEFT=<? echo $ORIGIN_BOTTOMLEFT; ?>;
   ORIGIN = <? echo $ORIGIN; ?>;

   YRES = <? echo $YRES; ?>;
   XRES = <? echo $XRES; ?>;

   function main() {
     debug("main");
     setupBoards();
     setupLayout();
     if (window.svgwin && svgwin.updateDimensionsForm) {
       svgwin.updateDimensionsForm(BOARDS_X,BOARDS_Y);
     } else if (top.updateDimensionsForm) {
       top.updateDimensionsForm(BOARDS_X,BOARDS_Y);
     } else {
       alert("broken implementation");
     }
     debug("done");
     if (!top.defaultPic) {
       top.defaultPic = {};
       top.defaultPic.x = 31;
       top.defaultPic.y = 40;
       top.defaultPic.stride =
[85,86,87,88,92,94,97,100,98,98,102,107,109,112,112,114,121,121,121,123,126,128,130,132,134,138,139,1
     }
     blitBitmap(top.defaultPic,"center","middle");
   }

   function setupBoards() {
     debug("setting up boards");
     var boardTemplate = document.getElementById("board");
     boardTemplate.setAttributeNS(null,"width",(FRAME_SIZE+BOARD_SIZE)/2);
     boardTemplate.setAttributeNS(null,"height",(FRAME_SIZE+BOARD_SIZE)/2);
   }

   function putPixel(x,y,level) {
     if (ORIGIN==ORIGIN_BOTTOMLEFT) {
       y = (BOARDS_Y*LIGHTS_Y) - 1 - y;
       x = (BOARDS_X*LIGHTS_X) - 1 - x;
     }
     var bx = parseInt(x/LIGHTS_X);
     var by = parseInt(y/LIGHTS_Y);
     var lx = x % LIGHTS_X;
     var ly = y % LIGHTS_Y;
     setLight(bx,by,lx,ly,level);
   }

   function setLight(bx,by,lx,ly,level) {
     var light = document.getElementById("light_" + bx + "_" + by + "_" + lx + "_" +ly);
     if (light) {
     level = parseFloat(level/255);
       if (level > 1) { level = 1; }
       if (level < 0) { level = 0; }
       light.style.setProperty("fill-opacity",level,"");
       light.curOpacity = level;
       light.increasing = 0;
     } else {
       debug("light : light_" + bx + "_" + by + "_" + lx + "_" +ly+ " not found.");
     }
   }

   function setupLayout() {
     var canvas = document.createElementNS("http://www.w3.org/2000/svg","g");
     canvas.setAttributeNS(null,"id","canvas");
     canvas.setAttributeNS("http://www.w3.org/2000/svg", "x", OFFSET_X);
     canvas.setAttributeNS("http://www.w3.org/2000/svg", "y", OFFSET_Y);
     window.onmousedown = function() { window.mousedown=1;};
     window.onmouseup = function() { window.mousedown=0;};

     var LIGHT_SPACING_X = FRAME_SIZE/LIGHTS_X;
     var LIGHT_SPACING_Y = FRAME_SIZE/LIGHTS_Y;
     var LIGHT_OFFSET_X = LIGHT_SPACING_X/3;
     var LIGHT_OFFSET_Y = LIGHT_SPACING_Y/3;

     for (var bx=0;bx<BOARDS_X;bx++) {
       for (var by=0;by<BOARDS_Y;by++) {
         var boardGroup = document.createElementNS("http://www.w3.org/2000/svg","g");
         boardGroup.setAttributeNS(null, "transform",
"translate("+(OFFSET_X+(bx*FRAME_SIZE))+","+(OFFSET_Y+(by*FRAME_SIZE))+")");
```

```
          var board = document.createElementNS("http://www.w3.org/2000/svg","use");
          board.setAttributeNS("http://www.w3.org/1999/xlink", "href","#board");
          boardGroup.appendChild(board);
          for (var lx=0;lx<LIGHTS_X;lx++) {
            for (var ly=0;ly<LIGHTS_Y;ly++) {
              var light = document.createElementNS("http://www.w3.org/2000/svg","circle");
              light.setAttributeNS(null,"id","light_" + bx + "_" + by + "_" + lx + "_" +
ly);
              light.setAttributeNS(null,"class","light");
              light.setAttributeNS(null,"r",LIGHT_SIZE);
              light.setAttributeNS(null, "cx", LIGHT_OFFSET_X+(LIGHT_SPACING_X*lx));
              light.setAttributeNS(null, "cy", LIGHT_OFFSET_Y+(LIGHT_SPACING_Y*ly));
              light.curOpacity = parseInt(0);
              light.onmouseover = function() { if (!this.increasing && window.mousedown)
{this.increasing = 1; increaseLight(this); } };
              boardGroup.appendChild(light);
            }
          }
          canvas.appendChild(boardGroup);
        }
      }
      document.getElementById("digitalcanvas").appendChild(canvas);
    }

    function increaseLight(light) {
      if(light.curOpacity < 1) {
        if ((light.curOpacity+STEP)<1) {
          light.style.setProperty("fill-opacity", light.curOpacity+STEP,"");
          light.curOpacity = (light.curOpacity+STEP);
        } else {
          light.style.setProperty("fill-opacity", 1, "");
          light.curOpacity = 1;
        }
        light.timeout = window.setTimeout(function() { increaseLight(light)},STEPTIMEOUT);
      }
    }

    function clearWall() {
      var lights = document.getElementsByTagName("circle");
      for (var i=0;i<lights.length;i++) { if(lights[i].id != "board") {
window.clearTimeout(lights[i].timeout); lights[i].curOpacity = 0; lights[i].increasing =
0; lights[i].style.setProperty("fill-opacity", 0, ""); } }
    }

    function blitBitmap(bitmap,align,valign) {
      var i=0;
      var xoffset = 0;
      var yoffset = 0;
      if (align=="center") { xoffset = parseInt((XRES-bitmap.x)/2); }
      if (valign=="middle") { yoffset = parseInt((YRES-bitmap.y)/2); }
      for(var y=0;y<bitmap.y;y++) {
        for(var x=0;x<bitmap.x;x++) {
          putPixel(x+xoffset,y+yoffset,bitmap.stride[i]);
          i++;
        }
      }
    }

    function switchBoards() {
      if (top.boardsOff) {
//        document.getElementById('board').style.setProperty("fill", "#004400", "");
        document.getElementById('board').setAttributeNS(null, "class", "board");
      } else {
//        document.getElementById('board').style.setProperty("fill", "#000000", "");
        document.getElementById('board').setAttributeNS(null, "class", "boardoff");
      }
      if (window.opera && window.opera.reDraw) { window.opera.reDraw(); alert("lol"); }
      top.boardsOff = !top.boardsOff;
    }

    function debug(msg) {
      if (window.DEBUG) { if (window.opera && opera.postError) { opera.postError(msg); }
else { alert(msg); } }
    }

  ]]&gt;
  </script>
  <defs>
  <rect id="board" class="board" />
  </defs>
  <text id="info" y="<? echo ($WALL_Y-10); ?>" x="10">
<?
```

```
print "Wall: ($WALL_X"."cm x $WALL_Y"."cm), ";
print "Canvas: ".($BOARDS_X*$FRAME_SIZE)."cm x ".($BOARDS_X*$FRAME_SIZE)."cm), ";
print "Cards: ".($BOARDS_X*$BOARDS_Y)." ($BOARDS_X x $BOARDS_Y), ";
print "Offset: ($OFFSET_X"."cm x $OFFSET_Y"."cm)";
?>
</text>
  <!--text x="90" y="960" onclick="window.DEBUG=!window.DEBUG;" style="fill: white;
font-size: 20px;">Debug</text>
  <text x="190" y="960" onclick="window.blitBitmap(top.defaultPic,'center');" style="fill:
white; font-size: 20px;">Blit testpic</text>
  <text x="340" y="960" onclick="window.clearWall();" style="fill: white; font-size:
20px;">Clear</text>
  <text x="440" y="960" onclick="window.switchBoards();" style="fill: white; font-size:
20px;">No boards</text-->
</svg>
```

./simulator/digitalcanvas.php:

```
<html>
<head>
<title>Digital Canvas Simulator</title>
<style type="text/css">
body { margin: 1%; padding: 0; }
</style>
<script type="application/ecmascript" src="svgaccess.js" />
<script type="text/javascript">
<?



if (sizeof($_GET)) {
  $pairs = array();
  foreach($_GET as $key => $value) {
    $pairs[] = "$key=$value";
  }
  $query = "?".join("&",$pairs);
} else {
  $query = "";
}

$target_path = "uploads/";

$target_path = $target_path . basename( $_FILES['img']['name']);

if(move_uploaded_file($_FILES['img']['tmp_name'], $target_path)) {

    if ($_POST['monochrome']) {
      $monochrome = "-monochrome";
    }

    if ($_POST['edge']) {
      $edge = "-edge 1";
    }

    if ($_POST['normalize']) {
      $normalize = "-normalize";
    }

    if ($_POST['equalize']) {
      $equalize = "-equalize";
    }

/*    print "alert(\"";

    preg_replace("/\n/msi","\\\n",preg_replace("/\"/msi","\\\"",var_dump($_POST)));

    print "\");";*/

    if ($_POST['invert']) {
      $invert = 1;
    }

    if (!isset($_GET['boards_x'])) { $_GET['boards_x'] = 18; }
    if (!isset($_GET['boards_y'])) { $_GET['boards_y'] = 6; }

    $xres = $_GET['boards_x'] * 5;
    $yres = $_GET['boards_y'] * 5;
```

```
    $resolution = $xres . "x" . $yres;

    $newsizedetails = `convert -verbose -resize $resolution $target_path GRAY:/dev/null`;
    preg_match("/=>.*?(\d+)x(\d+)/msi",$newsizedetails,$matches);
    $width = $matches[1];
    $height = $matches[2];
    $newsizedetails = preg_replace("/\n/msi","\\n",$newsizedetails);
    $rawimage = `convert $normalize $edge $equalize $monochrome -resize $resolution
$target_path GRAY:-`;
    $stride_ary = array();
    for($i=0;$i<strlen($rawimage);$i++) {
      if ($invert) {
        $stride_ary[] = 255-ord($rawimage{$i});
      } else {
        $stride_ary[] = ord($rawimage{$i});
      }
    }
?>
  var defaultPic = {};
  defaultPic.x = <? print $width; ?>;
  defaultPic.y = <? print $height; ?>;
  defaultPic.stride = <? print "[".implode(",",$stride_ary)."];"; ?>;
<?

} else {

?>
  var defaultPic = {};
  defaultPic.x = 45;
  defaultPic.y = 40;
  defaultPic.stride = [255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
255, 255, 255, 255, 255, 254, 253, 252, 251, 248, 240,
221, 144, 145, 157, 167, 184, 199, 213, 229, 241, 252, 254, 255, 255, 255, 255, 255, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
255,
255, 255, 255, 255, 255, 255, 253, 250, 250, 245, 227, 193, 155, 125, 103, 93, 149, 218,
228, 197, 164, 139, 122, 144, 207, 213, 244, 254, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
255, 254, 243, 199, 228, 239, 198, 142, 100, 80, 75, 75,
75, 78, 111, 192, 202, 123, 83, 75, 87, 193, 244, 169, 143, 202, 235, 252, 255, 255, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
255, 255, 255, 249, 199, 118, 92, 178, 136, 80, 75, 75, 76, 76, 76, 107, 179, 238, 254,
254, 241, 204, 146, 152, 245, 254, 237, 115, 111, 177, 207,
236, 254, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 254, 229, 143,
84, 120, 203, 244, 200, 119, 78, 75, 75, 86, 154, 232, 254,
254, 255, 254, 254, 254, 209, 103, 125, 171, 215, 172, 78, 98, 175, 197, 212, 250, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 253, 201,
103, 89, 171, 241, 254, 254, 254, 240, 180, 110, 106, 197, 250, 254, 254, 255, 255, 255,
255, 247, 133, 75, 75, 76, 86, 108, 161, 149, 136, 204, 192,
191, 250, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 251, 178, 86, 111, 208, 253, 254,
254, 255, 254, 254, 253, 187, 130, 205, 245, 254, 254, 254,
254, 254, 254, 197, 83, 76, 76, 76, 76, 76, 182, 253, 245, 234, 154, 94, 173, 246, 255,
255, 255, 255, 255, 255, 255, 254, 173, 79, 130, 230, 254,
254, 254, 255, 255, 254, 253, 194, 96, 75, 84, 124, 183, 231, 252, 254, 254, 243, 123, 75,
76, 76, 76, 76, 76, 75, 140, 252, 254, 254, 224, 108, 81, 155,
255, 255, 255, 255, 255, 255, 254, 238, 152, 141, 238, 254, 255, 255, 255, 255, 254, 253,
192, 91, 75, 76, 76, 75, 78, 101, 149, 202, 242, 190, 80,
75, 76, 76, 76, 76, 75, 111, 241, 255, 254, 254, 203, 88, 75, 255, 255, 255, 255, 255,
254, 240, 234, 178, 181, 253, 254, 254, 254, 254, 254, 254,
196, 92, 75, 76, 76, 76, 76, 76, 75, 75, 82, 119, 169, 121, 88, 76, 75, 75, 76, 75, 93,
226, 255, 255, 255, 253, 160, 76, 255, 255, 255, 255, 255,
245, 233, 193, 90, 86, 163, 238, 254, 254, 255, 254, 209, 98, 75, 76, 76, 76, 76, 76, 76,
76, 76, 75, 124, 243, 245, 216, 174, 130, 97, 79, 75, 82,
209, 255, 254, 254, 254, 232, 115, 255, 255, 255, 255, 250, 234, 208, 102, 75, 76, 78,
119, 201, 248, 254, 227, 112, 77, 80, 81, 81, 80, 79, 77, 77,
76, 76, 81, 190, 254, 254, 254, 254, 249, 228, 193, 152, 119, 196, 253, 254, 254, 255,
254, 199, 255, 255, 255, 254, 239, 223, 123, 75, 76, 76, 76,
75, 85, 140, 208, 140, 80, 87, 90, 91, 90, 89, 87, 84, 81, 80, 78, 115, 238, 254, 255,
255, 255, 255, 254, 254, 253, 244, 163, 133, 167, 198, 222,
239, 244, 255, 255, 255, 246, 232, 158, 77, 76, 76, 76, 76, 76, 76, 77, 159, 189, 131,
104, 101, 104, 103, 101, 97, 92, 89, 87, 86, 175, 254, 254,
255, 255, 255, 255, 255, 255, 255, 254, 158, 76, 76, 79, 90, 106, 122, 255, 255, 253, 237,
200, 91, 75, 76, 76, 76, 76, 77, 78, 120, 234, 254, 244,
205, 159, 129, 118, 115, 111, 104, 99, 95, 113, 228, 255, 255, 255, 255, 255, 255, 255,
255, 255, 254, 173, 89, 83, 77, 76, 75, 75, 255, 255, 234,
224, 131, 75, 76, 76, 76, 76, 77, 79, 96, 203, 254, 254, 254, 254, 249, 226, 186, 149,
127, 116, 112, 106, 165, 253, 255, 255, 255, 255, 255,
255, 255, 255, 254, 181, 101, 92, 82, 77, 76, 76, 255, 253, 158, 151, 86, 76, 76, 76, 76,
76, 79, 84, 154, 250, 254, 255, 255, 255, 255, 254, 253,
241, 209, 168, 136, 128, 217, 254, 255, 255, 255, 255, 255, 255, 255, 255, 255, 254, 189,
116, 103, 90, 80, 77, 76, 255, 230, 101, 156, 103, 75, 76,
76, 76, 77, 83, 110, 224, 255, 255, 255, 255, 255, 255, 255, 254, 254, 254, 251, 232, 192,
```

```
214, 245, 254, 254, 254, 254, 255, 255, 255, 255, 255,
254, 195, 130, 115, 98, 86, 78, 76, 255, 185, 110, 239, 212, 111, 75, 76, 76, 80, 88, 169,
253, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
254, 254, 194, 149, 169, 197, 225, 244, 252, 254, 254, 255, 255, 255, 254, 195, 141, 126,
107, 92, 80, 76, 253, 138, 157, 254, 254, 229, 143, 82, 75,
82, 112, 228, 254, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 254, 243, 172, 157,
158, 160, 168, 182, 201, 222, 239, 250, 254, 254, 254, 195,
150, 135, 116, 97, 84, 77, 237, 114, 206, 254, 255, 254, 247, 189, 111, 85, 167, 254, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 254,
220, 164, 163, 164, 166, 169, 172, 172, 174, 180, 192, 209, 226, 238, 192, 156, 140, 121,
102, 87, 79, 209, 120, 235, 254, 255, 255, 254, 254, 233,
170, 209, 254, 254, 254, 255, 255, 255, 255, 255, 255, 255, 255, 255, 252, 194, 164, 165,
166, 168, 171, 173, 174, 174, 173, 173, 173, 175, 182, 219,
217, 190, 160, 128, 101, 84, 180, 144, 247, 254, 255, 255, 255, 254, 254, 199, 134, 195,
240, 254, 254, 255, 255, 255, 255, 255, 255, 255, 255, 238,
174, 164, 163, 164, 166, 170, 173, 174, 174, 174, 174, 174, 173, 179, 239, 254, 254, 250,
239, 221, 202, 157, 170, 249, 254, 255, 255, 255, 255, 249,
146, 108, 132, 165, 207, 240, 253, 254, 255, 255, 255, 255, 254, 254, 214, 163, 161, 160,
160, 163, 166, 170, 174, 174, 174, 174, 174, 173, 184, 245,
255, 255, 255, 255, 254, 254, 149, 191, 248, 254, 255, 255, 255, 255, 224, 108, 110, 131,
148, 161, 179, 206, 234, 251, 254, 254, 254, 254, 250, 185,
156, 152, 150, 152, 156, 162, 166, 171, 174, 174, 174, 174, 173, 191, 250, 255, 255, 255,
255, 255, 255, 188, 210, 245, 253, 255, 255, 255, 254, 182,
91, 108, 128, 145, 159, 166, 171, 178, 196, 222, 243, 253, 254, 234, 162, 148, 143, 140,
142, 147, 154, 161, 166, 170, 173, 173, 173, 170, 198, 253,

255, 255, 255, 255, 255, 255, 198, 173, 241, 251, 254, 255, 254, 250, 137, 87, 104, 122,
139, 153, 162, 168, 171, 171, 173, 182, 202, 228, 202, 145,
136, 130, 128, 130, 136, 144, 152, 160, 165, 167, 169, 168, 165, 207, 254, 255, 255, 255,
255, 255, 255, 187, 89, 172, 246, 254, 255, 255, 233, 103,
85, 98, 114, 130, 144, 155, 162, 165, 167, 166, 164, 160, 163, 214, 200, 159, 128, 114,
115, 122, 132, 142, 150, 156, 160, 162, 161, 160, 218, 254,
255, 255, 255, 255, 255, 255, 183, 73, 78, 157, 240, 254, 254, 207, 83, 82, 93, 106, 120,
134, 144, 152, 157, 159, 159, 156, 150, 167, 246, 254, 247,
223, 185, 148, 125, 121, 127, 136, 144, 148, 151, 150, 154, 231, 254, 255, 255, 255, 255,
255, 255, 191, 76, 68, 74, 127, 218, 254, 175, 75, 79, 87,
97, 109, 121, 132, 140, 144, 147, 146, 144, 137, 186, 254, 254, 254, 254, 254, 249, 230,
201, 172, 150, 138, 134, 134, 134, 154, 244, 254, 255, 255,
255, 255, 255, 255, 202, 77, 66, 71, 72, 95, 171, 139, 75, 76, 81, 89, 98, 108, 118, 125,
130, 132, 132, 129, 128, 214, 254, 255, 255, 255, 255, 254,
254, 254, 253, 244, 226, 200, 174, 148, 168, 247, 254, 254, 254, 254, 255, 255, 255, 219,
84, 65, 69, 72, 73, 83, 170, 137, 84, 76, 82, 90, 97, 104,
110, 114, 116, 116, 112, 132, 238, 254, 255, 255, 255, 255, 255, 255, 255, 255, 255, 254,
254, 253, 244, 169, 139, 164, 186, 207, 226, 239, 247, 251,
236, 99, 63, 67, 69, 71, 86, 220, 249, 199, 128, 86, 79, 87, 92, 96, 100, 101, 101, 97,
157, 253, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255,
255, 255, 255, 246, 129, 85, 84, 81, 84, 93, 106, 119, 131, 251, 127, 60, 65, 67, 68, 91,
228, 255, 254, 247, 203, 137, 92, 81, 84, 89, 90, 89, 89,
192, 254, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 255, 223, 96, 80,
79, 77, 76, 75, 75, 75, 75, 255, 165, 59, 63, 65, 66, 94,
224, 253, 255, 255, 254, 249, 214, 154, 103, 81, 79, 79, 98, 226, 255, 255, 255, 255, 255,
255, 255, 255, 255, 255, 255, 255, 255, 254, 184, 78, 77,
77, 76, 76, 76, 76, 76, 74, 255, 218, 95, 60, 64, 65, 91, 214, 243, 253, 255, 255, 255,
254, 252, 230, 180, 125, 85, 128, 248, 254, 254, 255, 255,
255, 255, 255, 255, 255, 255, 255, 255, 255, 251, 137, 75, 76, 76, 76, 76, 76, 76, 74, 70,
255, 248, 171, 92, 61, 62, 88, 206, 230, 243, 252, 255,
255, 254, 254, 254, 254, 245, 210, 166, 212, 244, 254, 255, 255, 254, 255, 255, 255, 255,
255, 255, 255, 255, 229, 98, 75, 76, 76, 76, 76, 76, 75,
70, 66, 255, 255, 214, 162, 108, 63, 84, 197, 218, 229, 242, 252, 254, 255, 255, 255, 255,
254, 246, 123, 85, 120, 169, 213, 242, 253, 254, 255, 255,
255, 255, 255, 255, 254, 188, 78, 76, 76, 76, 76, 76, 74, 70, 66, 64, 255, 255, 242, 180,
172, 134, 92, 184, 209, 214, 227, 239, 249, 254, 255, 255,
255, 254, 225, 94, 75, 75, 75, 87, 115, 155, 197, 229, 248, 254, 254, 254, 254, 250, 135,
75, 76, 76, 76, 76, 73, 69, 66, 63, 61, 255, 255, 254, 206,
166, 179, 163, 167, 198, 203, 212, 225, 236, 246, 252, 254, 255, 254, 198, 80, 76, 76, 76,
76, 75, 75, 81, 101, 135, 176, 216, 246, 255, 225, 95, 75,
76, 76, 75, 72, 69, 66, 63, 60, 59 ];

<?


}

?>

function launch() {
  init();
}

function updateDimensionsForm(x,y) {
  document.getElementById("boards_x").value = x;
  document.getElementById("boards_y").value = y;
```

```
  calculateTotal();
}

function calculateTotal() {
  document.getElementById("boards").value = (document.getElementById("boards_x").value *
document.getElementById("boards_y").value);
}

window.onload = launch;

</script>
</head>
<body>
<br>
<br>
<center>
<object data="digitalcanvas.svg<? echo $query; ?>" type="image/svg+xml" id="object"
width="1000" height="357">
Please use a browser that supports SVG. <a href="http://opera.com/download">Opera</a> will
do.
</object>
</center>
<form name="imgupload" enctype="multipart/form-data" method="POST" action="">
<br />
Upload an image to be displayed. Any format should work.
<br /><br />
Invert: <input type=checkbox name="invert" value="1" <? print $invert?"checked":""; ?>>
Black/White: <input type=checkbox name="monochrome" value="1" <? print
$monochrome?"checked":""; ?>>
Normalize: <input type=checkbox name="normalize" value="1" <? print
$normalize?"checked":""; ?>>
Equalize: <input type=checkbox name="equalize" value="1" <? print $equalize?"checked":"";
?>>
Edge detect: <input type=checkbox name="edge" value="1" <? print $edge?"checked":""; ?>>
<input type=file name="img">
<input type=submit name="submit" value="Upload">
</form>
<br>
<form name="sizeform" method="GET" action="">
Boards horiz.: <input type="text" name="boards_x" size="3" value="<? echo
$_GET['boards_x']; ?>" onblur="calculateTotal()"><br>
Boards vert.: <input type="text" name="boards_y" size="3" value="<? echo
$_GET['boards_y']; ?>" onblur="calculateTotal()"><br>
Total boards: <input type="text" name="boards" size="3" value="<? echo
($_GET['boards_y']*$_GET['boards_x']); ?>" disabled><br>
<input type="submit" name="submit" value="Change size">
</form>
<input type="button" onclick="svgdoc.parentWindow.clearWall()" name="Clear" value="Clear
Wall" />
</body>
</html>
```

./simulator/LICENSE:

```
Unless otherwise stated, the files in this directory and subdirectories are licensed with
a two-clause BSD License,
and copyright is held by Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>

This source file is licensed under a two-clause BSD license

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
```

```
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

./simulator/README:

```
Open Digital Canvas -- simulator

This is a simulator made with HTML and SVG. It allows you to draw on a simulated canvas
and upload pictures to be
rendered as if they were rendered in a real canvas consisting of boards. This was made
early in the project to help
members visualize how the resulting work would look like.

Please see http://odc.openthweb.org for more information.

Check INSTALL for instrutions to install this software and LICENSE to see its license.
```

./avr.diff:

```
diff -ru avr/app.c ../avr/app.c
--- avr/app.c        2006-02-27 22:37:14.000000000 +0100
+++ ../avr/app.c        2007-05-15 18:39:29.000000000 +0200
@@ -3,11 +3,16 @@
 ****************************************************************************/

 #include "app.h"
-#include <avr\signal.h>
-#include <avr\interrupt.h>
-#include <avr\io.h>
+/* #include <avr/signal.h> */ /* Deprecated */
+#include <avr/interrupt.h>
+#include <avr/io.h>
+#include <string.h>
 #include "binary.h"
 #include "compiler.h"
+#define F_CPU 16000000
+#include <util/delay.h>
+
+

 void setup_card(void);

@@ -21,52 +26,32 @@
 #define PACKET_DATA                                             3
 #define PACKET_UPDATE                                   4

-        unsigned char Outputs[NUM_OUTPUTS];
-
-        unsigned char innverdi[5][5];
-        unsigned char tempverdi[5][5];
-        unsigned char utmatrise[5][RES];
+unsigned char Outputs[NUM_OUTPUTS];

-        unsigned char light_levels[RES];
+unsigned char innverdi[5][5];
+unsigned char tempverdi[5][5];
+unsigned char utmatrise[5][RES];

-        unsigned char busy_updating;
-
-
-        typedef struct
-        {
-                unsigned char bit0 : 1;
-                unsigned char bit1 : 1;
-                unsigned char bit2 : 1;
-                unsigned char bit3 : 1;
-                unsigned char bit4 : 1;
-                unsigned char bit5 : 1;
-                unsigned char bit6 : 1;
-                unsigned char bit7 : 1;
-        } bit_field;
+unsigned char light_levels[RES];
```

```
 //END SR-testing

 void uip_app(){}

-void lights_init(void)
-{
-
+void lights_init(void) {

         LED_DRIVER_DDR = 0x7;


uip_listen(HTONS(5001)); //setter port det skal lyttes på


/* Oppstartssjekk. Naar kortet slås på (strøm settes på) fader alle lysene opp og ned et
gitt antall ganger.  */


         int t = 0;
         int s = 0;

         int count = 0;
         int dir = 1;
-        int fades = 1; //antall fadinger opp og ned
+        int fades = 2; //antall fadinger opp og ned

         LED_SOURCE_1 = 1;
         LED_SOURCE_2 = 1;
@@ -74,16 +59,12 @@
         LED_SOURCE_4 = 1;
         LED_SOURCE_5 = 1;

-
-//while(1) {
-        while(s<RES*2*fades) { //setter antall fadinger
+    while (s<RES*2*fades) { //setter antall fadinger
                 for(int k=0;k<RES;k++){
                         if(count>k){
-                                //if(1) {
-                                utmatrise[0][k]=0b11111000;
-                        }//if
-                        else{
-                                utmatrise[0][k]=0b00000000;
+                                utmatrise[0][k]=b11111000;
+                        } else {
+                                utmatrise[0][k]=b00000000;
                         }
                 }//RES-for

@@ -93,28 +74,29 @@
                         }
                         t++;
                 }
-                        t=0;
-                        s++;
+                t=0;
+                s++;
+
                 switch(dir) {
                         case 0 : count--;
                                 break;
                         case 1 : count++;
                                 break;
                 }
+
                 if(count==RES){
                         dir = 0;
-                }
-                else if(count==0) {
+                } else if(count==0) {
                         dir = 1;
                 }
         }
/* Slutt på lystest */

 /*Setter alle utverdiene til 0 = Alle lys av*/

-        for(int l=0;l<RES;l++){
-                utmatrise[0][l]=0b00000000;
-        }//RES-for
```

```
+/*        for(int l=0;l<RES;l++){
+               utmatrise[0][l]=b00000000;
+        }//RES-for */

         // Setup light levels transformation curve
         for (int i = 0; i < RES; ++i) {
@@ -133,27 +115,34 @@
         switch(uip_appdata[0])
         {
                 case PACKET_DATA_IMMEDIATE: // lysdata
-                        if (busy_updating != 1)
+                        if (!busy_updating_lights)
                         {
-                                busy_updating = 1;
+                                busy_updating_values = 1;
+                                while(busy_updating_lights) { /* wait until lights are
done updating */ }
                                 update_values();
-                                busy_updating = 0;
+                                commit_values();
+                                busy_updating_values = 0;
                         }
-                        commit_values();
                         break;
                 case PACKET_SETUP: // ip/mac setup
                         setup_card();
                         break;
                 case PACKET_DATA:
-                        if (busy_updating != 1)
+                        if (busy_updating_values != 1)
                         {
-                                busy_updating = 1;
+                                busy_updating_values = 1;
                                 update_values();
-                                busy_updating = 0;
+                                busy_updating_values = 0;
                         }
                         break;
                 case PACKET_UPDATE: // oppdater lysene med gjeldende data
-                        commit_values();
+                        if (!busy_updating_lights)
+                        {
+                                busy_updating_values = 1;
+                                while(busy_updating_lights) { /* wait until lights are
done updating */ }
+                                commit_values();
+                                busy_updating_values = 0;
+                        }
                         break;
         }
 }
@@ -201,24 +190,20 @@
 //smooth
 void update_lights(void)
 {
+
         for (int j = 0; j < 5; ++j) {
+                LED_SOURCE_1 = (j == 0 ? 1 : 0);
+                LED_SOURCE_2 = (j == 1 ? 1 : 0);
+                LED_SOURCE_3 = (j == 2 ? 1 : 0);
+                LED_SOURCE_4 = (j == 3 ? 1 : 0);
+                LED_SOURCE_5 = (j == 4 ? 1 : 0);
+                _delay_loop_2(10);
                 for (unsigned char level = 1; level < RES; level++) {
-                        for (int k = 0; k < 5; ++k) {
+                        for (int k = 0; k < 5; k++) {
                                 ChainSetBit(7 - k, (light_levels[innverdi[j][k]] >= level
? 1 : 0));
                         }
-
-                        LED_SOURCE_1 = (j == 0 ? 1 : 0);
-                        LED_SOURCE_2 = (j == 1 ? 1 : 0);
-                        LED_SOURCE_3 = (j == 2 ? 1 : 0);
-                        LED_SOURCE_4 = (j == 3 ? 1 : 0);
-                        LED_SOURCE_5 = (j == 4 ? 1 : 0);

                         UpdateChain();

                 }
-
-
-
```

```
         }
 }

Only in avr/: app.d
diff -ru avr/app.h ../avr/app.h
--- avr/app.h        2006-02-27 21:50:48.000000000 +0100
+++ ../avr/app.h        2007-05-15 18:39:29.000000000 +0200
@@ -23,13 +23,15 @@
 void update_lights(void);


-void            UpdateChain(void);
-void            InitialiseShiftRegisterChain(void);
-void            ChainSetBit(int BitNumber, int State);
-void            ChainSetBitNow(int BitNumber, int State);
-void            ChainSetByte(int ByteNumber, unsigned char Value);
-void            ChainSetByteNow(int ByteNumber, unsigned char Value);
+void UpdateChain(void);
+void InitialiseShiftRegisterChain(void);
+void ChainSetBit(int BitNumber, int State);
+void ChainSetBitNow(int BitNumber, int State);
+void ChainSetByte(int ByteNumber, unsigned char Value);
+void ChainSetByteNow(int ByteNumber, unsigned char Value);

+unsigned int busy_updating_lights;
+unsigned int busy_updating_values;

 #define FS_STATISTICS 0

@@ -38,6 +40,16 @@
 // RTL8019 control port
 //#define LED_DRIVER_PORT                                 PORTE

+typedef struct {
+        unsigned char bit0 : 1;
+        unsigned char bit1 : 1;
+        unsigned char bit2 : 1;
+        unsigned char bit3 : 1;
+        unsigned char bit4 : 1;
+        unsigned char bit5 : 1;
+        unsigned char bit6 : 1;
+        unsigned char bit7 : 1;
+} bit_field;

 #define LED_DRIVER_PORT (*((volatile bit_field*) (&PORTC))) //testlinje
 #define LED_DRIVER_DDR                          DDRC
Only in avr/: app.lst
Only in avr/: app.o
diff -ru avr/compiler.h ../avr/compiler.h
--- avr/compiler.h        2005-09-14 19:57:36.000000000 +0200
+++ ../avr/compiler.h        2007-05-15 18:39:29.000000000 +0200
@@ -26,9 +26,9 @@

 #else /* --- GCC --- */

-#include <avr\signal.h>
-#include <avr\interrupt.h>
-#include <avr\io.h>
+/* #include <avr/signal.h> */ /* Deprecated */
+#include <avr/interrupt.h>
+#include <avr/io.h>

 #define nop() asm volatile("nop\n\t"::);

Only in avr/: delay.d
Only in avr/: delay.lst
Only in avr/: delay.o
Only in avr/: eeprom.hex
Only in avr/: eeprom.read.hex
Only in ../avr/: INSTALL
Only in ../avr/: LICENSE
diff -ru avr/main.c ../avr/main.c
--- avr/main.c        2006-02-27 22:36:01.000000000 +0100
+++ ../avr/main.c        2007-05-15 18:39:29.000000000 +0200
@@ -25,9 +25,12 @@

 #include "compiler.h"

+#include <stdint.h>
+
 #include <avr/eeprom.h>
 #include <avr/wdt.h>
```

```
+#include "app.h"

 #define BUF ((struct uip_eth_hdr *)&uip_buf[0])

@@ -63,10 +66,14 @@
 #endif
 #define F_CPU                                   16000000

+#include <util/delay.h>

 #define TIMERCOUNTER_PERIODIC_TIMEOUT (F_CPU / TIMER_PRESCALE / 2 / 256)
 //#define TIMERCOUNTER_PERIODIC_TIMEOUT 15000

+/*uint16_t log_addr;
+struct uip_udp_conn *log_conn;
+*/
 volatile unsigned int timerCounter;
 unsigned int foo;

@@ -210,8 +217,11 @@
                         }
                     }

-
-                  update_lights();
+                if (!busy_updating_values) {
+                        busy_updating_lights = 1;
+                        update_lights();
+                        busy_updating_lights = 0;
+                }
           }

 }
@@ -247,6 +257,21 @@
         uip_ethaddr.addr[5] = eeprom_read_byte(&eth_mac_5);
 }

+/*void log_init() {
+    uip_addr(&log_addr, 10,0,0,1);
+    log_conn = uip_udp_new(&addr, HTONS(1000));
+}
+
+void log(char *m) {
+  if (log_conn != NULL) {
+        uip_udp_send(5);
+  }
+}
+
+void log_close() {
+    uip_udp_remove(log_conn);
+}*/
+
 /*-----------------------------------------------------------------------------
 *  Main Control Loop
 *
@@ -274,9 +299,14 @@

         sei();

+/*    log_init();
+
+    uip_buf = 12345;
+    log(""); */

         checkNet();

+/*    log_close(); */
+
         return 1;
 }
-
Only in avr/: main.d
Only in avr/: main.lst
Only in avr/: main.o
diff -ru avr/Makefile ../avr/Makefile
--- avr/Makefile        2005-09-18 08:54:41.000000000 +0200
+++ ../avr/Makefile        2007-05-15 18:39:29.000000000 +0200
@@ -87,7 +87,7 @@
 #  -Wall...:  warning level
 #  -Wa,...:   tell GCC to pass this to the assembler.
 #    -ahlms:  create assembler listing
```

```
-CFLAGS = -g -O$(OPT) \
+CFLAGS += -g -O$(OPT) \
 -funsigned-char -funsigned-bitfields -fpack-struct -fshort-enums \
 -Wall -Wstrict-prototypes \
 -Wa,-adhlns=$(<:.c=.lst) \
@@ -145,16 +145,16 @@
 # Type: avrdude -c ?
 # to get a full listing.
 #
-AVRDUDE_PROGRAMMER = avrisp
+AVRDUDE_PROGRAMMER = stk500v2


-AVRDUDE_PORT = com1            # programmer connected to serial device
+AVRDUDE_PORT = /dev/ttyS0            # programmer connected to serial device
 #AVRDUDE_PORT = lpt1        # programmer connected to parallel port

 AVRDUDE_WRITE_FLASH = -e -U flash:w:$(TARGET).hex
 AVRDUDE_WRITE_EEPROM = -U eeprom:w:$(TARGET).eep

-AVRDUDE_FLAGS = -p $(MCU) -P $(AVRDUDE_PORT) -c $(AVRDUDE_PROGRAMMER)
+AVRDUDE_FLAGS = -p $(MCU) -P $(AVRDUDE_PORT) -c $(AVRDUDE_PROGRAMMER) -b 115200

 # Uncomment the following if you want avrdude's erase cycle counter.
 # Note that this counter needs to be initialized first using -Yn,
@@ -168,7 +168,7 @@
 # Increase verbosity level.  Please use this when submitting bug
 # reports about avrdude. See <http://savannah.nongnu.org/projects/avrdude>
 # to submit bug reports.
-AVRDUDE_FLAGS += -v -v
+#AVRDUDE_FLAGS += -v -v



@@ -302,7 +302,11 @@
 program: $(TARGET).hex $(TARGET).eep
         $(AVRDUDE) $(AVRDUDE_FLAGS) $(AVRDUDE_WRITE_FLASH) $(AVRDUDE_WRITE_EEPROM)

+flash: $(TARGET).hex $(TARGET).eep
+        $(AVRDUDE) $(AVRDUDE_FLAGS) $(AVRDUDE_WRITE_FLASH)

+eeprom: $(TARGET).eep
+        $(AVRDUDE) $(AVRDUDE_FLAGS) $(AVRDUDE_WRITE_EEPROM)


 # Create final output files (.hex, .eep) from ELF output file.
Only in avr/: nic.d
Only in avr/: nic.lst
Only in avr/: nic.o
Only in ../avr/program: .svn
Only in ../avr/: README
diff -ru avr/rtl8019.c ../avr/rtl8019.c
--- avr/rtl8019.c        2005-09-28 20:17:23.000000000 +0200
+++ ../avr/rtl8019.c        2007-05-15 18:39:29.000000000 +0200
@@ -262,7 +262,7 @@
 #define TBCR0                0x05
 #define NCR                        0x05
 #define TBCR1                0x06
-#define ISR                        0x07
+#define RTL_ISR                        0x07
 #define CURR                0x07   // Page 1
 #define RSAR0                0x08
 #define CRDA0                0x08
@@ -289,13 +289,13 @@
 /*****************************************************************************
 *
 * RTL ISR Register Bits
-*
+*
 *****************************************************************************/
-#define ISR_RST        7
-#define ISR_OVW 4
-#define ISR_PRX 0
-#define ISR_RDC 6
-#define ISR_PTX 1
+#define RTL_ISR_RST        7
+#define RTL_ISR_OVW 4
+#define RTL_ISR_PRX 0
+#define RTL_ISR_RDC 6
+#define RTL_ISR_PTX 1
```

```
 /*****************************************************************************
@@ -348,7 +348,7 @@
         rtl8019Write(RSAR1,0x40);

         //clear the packet stored interrupt
-        rtl8019Write(ISR,(1<<ISR_PTX));
+        rtl8019Write(RTL_ISR,(1<<RTL_ISR_PTX));

         //load data byte count for remote DMA
         rtl8019Write(RBCR0, (unsigned char)(packetLength));
@@ -381,7 +381,7 @@
         rtl8019Write(CR,0x24);

         // clear the remote DMA interrupt
-        rtl8019Write(ISR, (1<<ISR_RDC));
+        rtl8019Write(RTL_ISR, (1<<RTL_ISR_RDC));
 }


@@ -424,7 +424,7 @@


         // clear the packet received interrupt flag
-        rtl8019Write(ISR, (1<<ISR_PRX));
+        rtl8019Write(RTL_ISR, (1<<RTL_ISR_PRX));

         // the boundary pointer is invalid, reset the contents of the buffer and exit
         if( (bnry >= RXSTOP_INIT) || (bnry < RXSTART_INIT) )
@@ -449,9 +449,9 @@
         // end the DMA operation
     rtl8019Write(CR, 0x22);
     for(i = 0; i <= 20; i++)
-        if(rtl8019Read(ISR) & 1<<6)
+        if(rtl8019Read(RTL_ISR) & 1<<6)
             break;
-    rtl8019Write(ISR, 1<<6);
+    rtl8019Write(RTL_ISR, 1<<6);


         rxlen = (pageheader[enetpacketLenH]<<8) + pageheader[enetpacketLenL];
@@ -483,9 +483,9 @@
         // end the DMA operation
     rtl8019Write(CR, 0x22);
     for(i = 0; i <= 20; i++)
-        if(rtl8019Read(ISR) & 1<<6)
+        if(rtl8019Read(RTL_ISR) & 1<<6)
             break;
-    rtl8019Write(ISR, 1<<6);
+    rtl8019Write(RTL_ISR, 1<<6);

     currentRetreiveAddress += length;
     if( currentRetreiveAddress >= 0x6000 )
@@ -501,9 +501,9 @@
         // end the DMA operation
     rtl8019Write(CR, 0x22);
     for(i = 0; i <= 20; i++)
-        if(rtl8019Read(ISR) & 1<<6)
+        if(rtl8019Read(RTL_ISR) & 1<<6)
             break;
-    rtl8019Write(ISR, 1<<6);
+    rtl8019Write(RTL_ISR, 1<<6);

         // set the boundary register to point to the start of the next packet
     rtl8019Write(BNRY, nextPage);
@@ -523,7 +523,7 @@
                 resend = 0;
         else if(data_L & 0x04)
         {
-                data_L = rtl8019Read(ISR);
+                data_L = rtl8019Read(RTL_ISR);
                 data_L = rtl8019Read(RTL_ISR);
                 if((data_L & 0x02) || (data_L & 0x08))
                     resend = 0;
             else
@@ -536,13 +536,13 @@
         rtl8019Write(CR, 0x62);
         rtl8019Write(CURR, RXSTART_INIT);
         rtl8019Write(CR, 0x22);
-        rtl8019Write(ISR, 0x10);
+        rtl8019Write(RTL_ISR, 0x10);
         rtl8019Write(TCR, TCR_INIT);

     if(resend)
```

```
         rtl8019Write(CR, 0x26);

-    rtl8019Write(ISR, 0xFF);
+    rtl8019Write(RTL_ISR, 0xFF);
 }


@@ -553,7 +553,7 @@
         HARD_RESET_RTL8019();

         // do soft reset
-        rtl8019Write( ISR, rtl8019Read(ISR) ) ;
+        rtl8019Write( RTL_ISR, rtl8019Read(RTL_ISR) ) ;
         delay_ms(50);

     // switch to page 3 to load config registers
@@ -600,7 +600,7 @@
         rtl8019Write(CR,0x21);
         rtl8019Write(DCR, DCR_INIT);
         rtl8019Write(CR,0x22);
-        rtl8019Write(ISR,0xFF);
+        rtl8019Write(RTL_ISR,0xFF);
         rtl8019Write(IMR, IMR_INIT);
         rtl8019Write(TCR, TCR_INIT);

@@ -610,9 +610,9 @@

 void rtl8019ProcessInterrupt(void)
 {
-        unsigned char byte = rtl8019Read(ISR);
+        unsigned char byte = rtl8019Read(RTL_ISR);

-        if( byte & (1<<ISR_OVW) )
+        if( byte & (1<<RTL_ISR_OVW) )
                 rtl8019Overrun();
 }

Only in avr/: rtl8019.d
Only in avr/: rtl8019.lst
Only in avr/: rtl8019.o
Only in ../avr/: .svn
Only in avr/: uip_arch.d
Only in avr/: uip_arch.lst
Only in avr/: uip_arch.o
Only in avr/: uip_arp.d
Only in avr/: uip_arp.lst
Only in avr/: uip_arp.o
Only in avr/: uip.d
Only in avr/: uIPexample.eep
Only in avr/: uIPexample.elf
Only in avr/: uIPexample.hex
Only in avr/: uIPexample.lss
Only in avr/: uIPexample.map
Only in avr/: uIPexample.sym
Only in avr/: uip.lst
Only in avr/: uip.o
diff -ru avr/uipopt.h ../avr/uipopt.h
--- avr/uipopt.h        2005-09-28 20:36:30.000000000 +0200
+++ ../avr/uipopt.h        2007-06-14 15:16:52.000000000 +0200
@@ -130,36 +130,43 @@
   */
 #define UIP_PINGADDRCONF 0

-#define UIP_IPADDR0     192 /**< The first octet of the IP address of
+#ifndef UIP_IPADDR0
+#define UIP_IPADDR0     10 /**< The first octet of the IP address of
                                this uIP node, if UIP_FIXEDADDR is
                                1. \hideinitializer */
-#define UIP_IPADDR1     168 /**< The second octet of the IP address of
+#endif
+#ifndef UIP_IPADDR1
+#define UIP_IPADDR1     0  /**< The second octet of the IP address of
                                this uIP node, if UIP_FIXEDADDR is
                                1. \hideinitializer */
-#define UIP_IPADDR2     0   /**< The third octet of the IP address of
+#endif
+#ifndef UIP_IPADDR2
+#define UIP_IPADDR2     10 /**< The third octet of the IP address of
                                this uIP node, if UIP_FIXEDADDR is
                                1. \hideinitializer */
-#define UIP_IPADDR3     254   /**< The fourth octet of the IP address of
+#endif
```

```
+#ifndef UIP_IPADDR3
+#define UIP_IPADDR3     1   /**< The fourth octet of the IP address of
                                this uIP node, if UIP_FIXEDADDR is
                                1. \hideinitializer */
-
+#endif
 #define UIP_NETMASK0    255 /**< The first octet of the netmask of
                                this uIP node, if UIP_FIXEDADDR is
                                1. \hideinitializer */
 #define UIP_NETMASK1    255 /**< The second octet of the netmask of
                                this uIP node, if UIP_FIXEDADDR is
                                1. \hideinitializer */
-#define UIP_NETMASK2    255 /**< The third octet of the netmask of
+#define UIP_NETMASK2    0 /**< The third octet of the netmask of
                                this uIP node, if UIP_FIXEDADDR is
                                1. \hideinitializer */
 #define UIP_NETMASK3    0   /**< The fourth octet of the netmask of
                                this uIP node, if UIP_FIXEDADDR is
                                1. \hideinitializer */

-#define UIP_DRIPADDR0   192 /**< The first octet of the IP address of
+#define UIP_DRIPADDR0   10 /**< The first octet of the IP address of
                                the default router, if UIP_FIXEDADDR is
                                1. \hideinitializer */
-#define UIP_DRIPADDR1   168 /**< The second octet of the IP address of
+#define UIP_DRIPADDR1   0  /**< The second octet of the IP address of
                                the default router, if UIP_FIXEDADDR is
                                1. \hideinitializer */
 #define UIP_DRIPADDR2   0   /**< The third octet of the IP address of
@@ -180,25 +187,26 @@
   */
 #define UIP_FIXEDETHADDR 0

-#define UIP_ETHADDR0    0x44  /**< The first octet of the Ethernet
+#define UIP_ETHADDR0    10  /**< The first octet of the Ethernet
                                  address if UIP_FIXEDETHADDR is
                                  1. \hideinitializer */
-#define UIP_ETHADDR1    0x45  /**< The second octet of the Ethernet
+#define UIP_ETHADDR1    10  /**< The second octet of the Ethernet
                                  address if UIP_FIXEDETHADDR is
                                  1. \hideinitializer */
-#define UIP_ETHADDR2    0x46  /**< The third octet of the Ethernet
+#define UIP_ETHADDR2    UIP_IPADDR0  /**< The third octet of the Ethernet
                                  address if UIP_FIXEDETHADDR is
                                  1. \hideinitializer */
-#define UIP_ETHADDR3    0x47  /**< The fourth octet of the Ethernet
+#define UIP_ETHADDR3    UIP_IPADDR1  /**< The fourth octet of the Ethernet
                                  address if UIP_FIXEDETHADDR is
                                  1. \hideinitializer */
-#define UIP_ETHADDR4    0x48  /**< The fifth octet of the Ethernet
+#define UIP_ETHADDR4    UIP_IPADDR2  /**< The fifth octet of the Ethernet
                                  address if UIP_FIXEDETHADDR is
                                  1. \hideinitializer */
-#define UIP_ETHADDR5    0x49  /**< The sixth octet of the Ethernet
+#define UIP_ETHADDR5    UIP_IPADDR3  /**< The sixth octet of the Ethernet
                                  address if UIP_FIXEDETHADDR is
                                  1. \hideinitializer */

+
 /** @} */
 /*---------------------------------------------------------------------------*/
 /**
```

./test/src/line_test.pl:

```
#!/usr/bin/perl -w

use ODC;

ODC::clear();
ODC::drawWuLine(2, 2, 12.75, 9.5, 99);
ODC::redraw();
```

./test/src/ODC.pm:

```
#!/usr/bin/perl -w

package ODC;

use strict;
use IO::Socket::INET;
use Time::HiRes qw ( time alarm sleep );

my $MAX_LIGHTS = 25;
my $UPDATE_DELAY = 0.0001;

our $lights_x = sqrt($MAX_LIGHTS);
our $lights_y = $lights_x;

my $port = 5001;
#new Net::IP($ARGV[0]) or die (Net::IP::Error());

my $OP_SETLIGHT = 1;
my $OP_SETIP = 2;

our $width;
our $height;
our @cards;

if (open(my $conf,"config.txt")) {
my $res_set = 0;
while(<$conf>) {
  my $line = $_;
  next if ($line =~ m/\s*#/);
  if($res_set) {
    if ($line =~ /(\d+\.\d+\.\d+.\d+)\s+(NE|NW|SE|SW|)/) {
      my %card;
      $card{'ip'} = $1;
      $card{'dir'} = $2;
      my @light_values = split(/,/,("0,"x$MAX_LIGHTS));
      $card{'light_values'} = \@light_values;
      push(@cards,\%card);
    }
  } elsif ($line =~ /(\d+)\s+(\d+)/) {
    $width = $1;
    $height = $2;
    $res_set = 1;
  }
}
close($conf);
}

my @socks;

foreach my $card (@cards) {
 print "Setting up UDP socket to: ".$card->{'ip'}.":$port\n";
 $card->{'sock'} = new
IO::Socket::INET->new(PeerPort=>$port,Proto=>'udp',PeerAddr=>$card->{'ip'}) or die ($!);
}

sub setPixel {
  setRawPixel(@_);
}

sub setRawPixel {
  my $level = int((pop()/255)*99);
  #print "$level\n";
  my $y = pop;
  my $x = pop;
  if ($x>-1 && $x < ($width*$lights_x) && $y >-1 && $y < ($height*$lights_y)) {
    my $bx = int($x/$lights_x);
    my $by = int($y/$lights_y);
    my $cindex = $bx+($by*$width);
    my $lindex = calcLightIndex($cindex,$x,$y,$lights_x,$lights_y);
    $cards[$cindex]->{'light_values'}->[$lindex] = $level;
  }
}

sub setBoardPixel {
  my $level = int((pop()/255)*99);
  #print "$level\n";
  my $y = pop;
  my $x = pop;
```

```
  my $cindex = pop;

  if ($x>-1 && $x < ($lights_x) && $y >-1 && $y < ($lights_y)) {
    my $lindex = calcLightIndex($cindex,$x,$y,$lights_x,$lights_y);
    $cards[$cindex]->{'light_values'}->[$lindex] = $level;
  }

}

sub getPixel {
  my $y = pop;
  my $x = pop;

  my $bx = int($x/$lights_x);
  my $by = int($y/$lights_y);
  my $cindex = $bx+($by*$width);
  my $lindex = calcLightIndex($cindex,$x,$y,$lights_x,$lights_y);

  return $cards[$cindex]->{'light_values'}->[$lindex];
}

sub calcLightIndex {
  my $lights_y = pop;
  my $lights_x = pop;
  my $y = pop;
  my $x = pop;
  my $cindex = pop;

  my $lx = $x % $lights_x;
  my $ly = $y % $lights_y;

  my $lindex;

  if ($cards[$cindex]->{'dir'} eq "SW") {
    $lindex = (($lights_x-1-$ly)*$lights_x)+($lx);
  } elsif ($cards[$cindex]->{'dir'} eq "SE") {
#    $lindex = ($lx+($ly*$lights_x));
    $lindex = (($lights_y-$ly)+(($MAX_LIGHTS-1)-($lights_x*($lx+1))));
  } elsif ($cards[$cindex]->{'dir'} eq "NE") {
    $lindex = (($lights_x-1)-$lx)+($ly*$lights_x);
  } else {
    #if ($cards[$cindex]->{'dir'} eq "NW") {
    $lindex = (($ly+($lx*$lights_x)));
  }

  return $lindex;
}

sub drawLine {
  my $level = pop;
  my $y1 = pop;
  my $x1 = pop;
  my $y0 = pop;
  my $x0 = pop;

  my $level2 = $level;

  #print "drawing line ($x0,$y0) -> ($x1,$y1) : $level2\n";

  my $steep = abs($y1 - $y0) > abs($x1 - $x0);

  my $tmp;

  if ($steep) {
    $tmp = $x0;
    $x0 = $y0;
    $y0 = $tmp;

    $tmp = $x1;
    $x1 = $y1;
    $y1 = $tmp;
  }
  if ($x0 > $x1) {
    $tmp = $x0;
    $x0 = $x1;
    $x1 = $tmp;

    $tmp = $y0;
    $y0 = $y1;
    $y1 = $tmp;

  }
```

```
  my $deltax = $x1 - $x0;
  my $deltay = abs($y1 - $y0);
  my $error = 0;
  my $ystep;
  my $y = $y0;
  if ($y0 < $y1)  { $ystep = 1; } else { $ystep = -1; }
  for my $x ($x0..$x1) {
    if ($steep) { setRawPixel($y,$x,$level2); } else { setRawPixel($x,$y,$level2); }
    $error = $error + $deltay;
    if ((2*$error) >= $deltax) {
      $y = $y + $ystep;
      $error = $error - $deltax;
      $level2 = $level;
    }
  }
}

sub drawWuLine {
  my $level = pop;
  my $y1 = pop;
  my $x1 = pop;
  my $y0 = pop;
  my $x0 = pop;

  my ($startx, $starty, $endx, $endy) = $x0<$x1?($x0,$y0,$x1,$y1):($x1,$y1,$x0,$y0);

  print "$startx, $starty, $endx, $endy\n";

  my $deltay = ($endy-$starty)/($endx-$startx);
  my $fracy;

  for(my $x = int($startx);$x<$endx;$x++) {
    $fracy=0;
    my $y = $x*$deltay;
    if ($y =~ m/^([^\.]+)\.([^\.]*)$/) {
       $y = $1;
       $fracy = $2+0.0;
    }
#    ($starty, $fracy) = $starty =~ m/^([^\.]+)\.?([^\.]*)$/;
    my $levely;
    if ($fracy == 0) { $levely = 1; } else { $levely = "0.".$fracy; }
    print "$y -- $fracy -- $x -- $deltay -- ".$x*$deltay." -- ".$level*$levely." --
".$level*(1-$levely)."\n";
    setRawPixel($x,$y,$level*$levely);
    if ($fracy != 0) { setRawPixel($x,$y+1,$level*(1-$levely)); }
  }
}


sub redraw {
  foreach my $card (@cards) {
    my $msg = pack("C"x($MAX_LIGHTS+1), $OP_SETLIGHT, @{$card->{'light_values'}});
    #print $msg;
#    if (!$card->{'sock'}) {
#      $card->{'sock'} = new
IO::Socket::INET->new(PeerPort=>$port,Proto=>'udp',PeerAddr=>$card->{'ip'}) or die ($!);
#    }
    $card->{'sock'}->send($msg);
    #print $card->{'ip'}.": ".join(",",@{$card->{'light_values'}})."\n";
    sleep $UPDATE_DELAY;
  }
}

sub clear {
  foreach my $card (@cards) {
    @{$card->{'light_values'}} = split(/,/,"0,"x$MAX_LIGHTS);
  }
  redraw();
}

sub runtests {

print "Starting tests\n";

print "Clearing\n";
clear();

my $level;

print "first test\n";

$level = 0;
```

```
for (1..10) {
for my $a (0..149) {
  my $x=0;
  my $y=0;
  for(my $i=0;$i<150;$i++) {
    if ($x==($width*$lights_x)) {
      $x=0; $y++;
    } else {
      $x++;
    }
#clear();
    setPixel($x-1,$y-1,($i+(149-$a))%149);
#redraw();
  }
  #sleep 1;
  redraw();
}
}

$level = 0;
foreach(1..100) {
  ($level < 100)?($level+=10):($level=0);
  foreach my $y (1..$height*$lights_y) {
#    foreach my $x (1..$width*$lights_x) {
      clear();
      drawLine(0,0,($width*$lights_x-1),$y-1,$level);
      redraw();
#    }
  }
}

print "second test\n";

### traverse whole canvas

$level = 0;
while($level<100) {
  $level+=20;
  foreach my $y (1..$height*$lights_y) {
    foreach my $x (1..$width*$lights_x) {
      setPixel($x-1,$y-1,$level);
      redraw();
    }
  }
}

### blink some times

$level = 0;
foreach(1..5) {
  ($level < 100)?($level+=10):($level=0);
  foreach my $y (1..$height*$lights_y) {
    foreach my $x (1..$width*$lights_x) {
      setPixel($x-1,$y-1,$level);
    }
  }
  redraw();
}


}

1;
```

./test/src/multi_card_anim.pl:

```
#!/usr/bin/perl -w

use IO::Socket::INET;
use Time::HiRes qw ( time alarm sleep );

my $MAX_LIGHTS = 25;
my $UPDATE_DELAY = 0.001;

my $port = 5001;
my @ips = @ARGV;
#new Net::IP($ARGV[0]) or die (Net::IP::Error());
```

```
my $OP_SETLIGHT = 1;
my $OP_SETIP = 2;


my @socks;
foreach(@ips) {
 print "Setting up UDP socket to: $_:$port\n";
 push @socks, new IO::Socket::INET->new(PeerPort=>$port,Proto=>'udp',PeerAddr=>$_) or die
($!);
}

#IO::Socket::INET::Error());

my $in;
{
local $/;
$in = <STDIN>;
}

while(1) {
foreach $line (split(/\n/,$in)) {
  next if ($line =~ m/\s*#/);
  my($delay,@light_values) = split(/\s+/,$line);
  sleep($delay);
  $msg = pack("C"x($MAX_LIGHTS+1), $OP_SETLIGHT, @light_values);
  foreach(@socks) {
    $_->send($msg);
    #print (inet_ntoa($_->peeraddr())."\n");
    sleep $UPDATE_DELAY;
  }
#  sleep $UPDATE_DELAY;
}
}
```

./test/src/config.txt:

```
# layout
10 11
10.0.10.1 SW
10.0.10.2 NW
10.0.10.3 NW
10.0.10.4 NE
10.0.10.5 SE
10.0.10.6 NW
10.0.10.7 SW
10.0.10.8 SE
# new cards
10.0.10.9 SE
10.0.10.10 SE
10.0.10.11 SE
10.0.10.12 SE
10.0.10.13 SE
10.0.10.14 SE
10.0.10.15 SE
10.0.10.16 SE
10.0.10.17 SE
10.0.10.18 SE
10.0.10.19 SE
10.0.10.20 SE
10.0.10.21 SE
10.0.10.22 SE
10.0.10.23 SE
10.0.10.24 SE
10.0.10.25 SE
10.0.10.26 SE
10.0.10.27 SE
10.0.10.28 SE
10.0.10.29 SE
10.0.10.30 SE
10.0.10.31 SE
10.0.10.32 SE
10.0.10.33 SE
10.0.10.34 SE
10.0.10.35 SE
10.0.10.36 SE
10.0.10.37 SE
10.0.10.38 SE
10.0.10.39 SE
```

```
10.0.10.40 SE
10.0.10.41 SE
10.0.10.42 SE
10.0.10.43 SE
10.0.10.44 SE
10.0.10.45 SE
10.0.10.46 SE
10.0.10.47 SE
10.0.10.48 SE
10.0.10.49 SE
10.0.10.50 SE
10.0.10.51 SE
10.0.10.52 SE
10.0.10.53 SE
10.0.10.54 SE
10.0.10.55 SE
10.0.10.56 SE
10.0.10.57 SE
10.0.10.58 SE
10.0.10.59 SE
10.0.10.60 SE
10.0.10.61 SE
10.0.10.62 SE
10.0.10.63 SE
10.0.10.64 SE
10.0.10.65 SE
10.0.10.66 SE
10.0.10.67 SE
10.0.10.68 SE
10.0.10.69 SE
10.0.10.70 SE
10.0.10.71 SE
10.0.10.72 SE
10.0.10.73 SE
10.0.10.74 SE
10.0.10.75 SE
10.0.10.76 SE
10.0.10.77 SE
10.0.10.78 SE
10.0.10.79 SE
10.0.10.80 SE
10.0.10.81 SE
10.0.10.82 SE
10.0.10.83 SE
10.0.10.84 SE
10.0.10.85 SE
10.0.10.86 SE
10.0.10.87 SE
10.0.10.88 SE
10.0.10.89 SE
10.0.10.90 SE
10.0.10.91 SE
10.0.10.92 SE
10.0.10.93 SE
10.0.10.94 SE
10.0.10.95 SE
10.0.10.96 SE
10.0.10.97 SE
10.0.10.98 SE
10.0.10.99 SE
10.0.10.100 SE
10.0.10.101 SE
10.0.10.102 SE
10.0.10.103 SE
10.0.10.104 SE
10.0.10.105 SE
10.0.10.106 SE
10.0.10.107 SE
10.0.10.108 SE
10.0.10.109 SE
10.0.10.110 SE
10.0.10.111 SE
10.0.10.112 SE
10.0.10.113 SE
10.0.10.114 SE
10.0.10.115 SE
10.0.10.116 SE
10.0.10.117 SE
10.0.10.118 SE
```

./test/src/canvas_test.pl:

```
#!/usr/bin/perl -w

use strict;
use ODC;

ODC::runtests();
```

./test/src/cube_test.pl:

```
#!/usr/bin/perl -w

use strict;
use ODC;
use Time::HiRes qw ( time alarm sleep );


my $d;
my $x;
my $x2;
my $x3;
my $x4;

my $y;
my $y2;
my $y3;
my $y4;

my $PI = 3.1415;
my $PI2 = $PI*2;
my $PI_2 = $PI/2;

my $width_2=$ODC::width*$ODC::lights_x/2;
my $height_2=$ODC::height*$ODC::lights_y/2;


while(++$d) {
  $x=cos($d)*$height_2+($width_2);
  $y=sin($d)*$height_2+($height_2);
  $x2=cos($d+$PI_2)*$height_2+($width_2);
  $y2=sin($d+$PI_2)*$height_2+($height_2);
  $x3=cos($d+$PI)*$height_2+($width_2);
  $y3=sin($d+$PI)*$height_2+($height_2);
  $x4=cos($d-$PI_2)*$height_2+($width_2);
  $y4=sin($d-$PI_2)*$height_2+($height_2);
#  print "$x,$y $x2,$y2\n";
  ODC::clear();
  ODC::drawWuLine($x,$y,$x2,$y2,99);
  ODC::drawLine($x2,$y2,$x3,$y3,99);
  ODC::drawLine($x3,$y3,$x4,$y4,99);
  ODC::drawLine($x4,$y4,$x,$y,99);
  ODC::redraw();
  sleep(0.05);
}
```

./test/src/numerize.pl:

```
#!/usr/bin/perl -w

use ODC;

my @numbers =
(
"00 00 00 99 00
 00 00 99 00 99
 00 00 99 00 99
 00 00 99 00 99
 00 00 00 99 00",

"00 00 00 99 00
```

```
 00 00 99 99 00
 00 00 00 99 00
 00 00 00 99 00
 00 00 99 99 99",

"00 00 99 99 00
 00 00 00 00 99
 00 00 00 99 00
 00 00 99 00 00
 00 00 99 99 99",

"00 00 99 99 99
 00 00 00 00 99
 00 00 99 99 00
 00 00 00 00 99
 00 00 99 99 00",

"00 00 99 00 99
 00 00 99 00 99
 00 00 99 99 99
 00 00 00 00 99
 00 00 00 00 99",

"00 00 99 99 99
 00 00 99 00 00
 00 00 99 99 00
 00 00 00 00 99
 00 00 99 99 00",

"00 00 00 99 99
 00 00 99 00 00
 00 00 99 99 00
 00 00 99 00 99
 00 00 00 99 00",

"00 00 99 99 99
 00 00 00 00 99
 00 00 00 99 00
 00 00 00 99 00
 00 00 00 99 00",

"00 00 00 99 00
 00 00 99 00 99
 00 00 00 99 00
 00 00 99 00 99
 00 00 00 99 00",

"00 00 00 99 00
 00 00 99 00 99
 00 00 00 99 99
 00 00 00 00 99
 00 00 00 99 00",

);

ODC::clear();

my $number = 0;
foreach my $card (@ODC::cards) {
   my $y=0;
   my $x=0;
   foreach my $line (split(/\n/,$numbers[($number+1)%10])) {
     $line=~s/^\s//msgi;
     foreach my $char (split(/\s+/,$line)) {
       for $lx (0..1) {
         for $ly (0..9) {
           if (POSIX::floor(($number+1)/10) > (($lx*$ODC::lights_x)+$ly)) {
             ODC::setBoardPixel($number,$lx,$ly,99);
           }
         }
       }
       ODC::setBoardPixel($number,$x,$y,getnum($char));
       $x++;
     }
     $y++;
     $x=0;
   }
   $number++;
}
ODC::redraw();

sub getnum {
```

```
  use POSIX qw(strtod);
  my $str = shift;
  $str =~ s/^\s+//;
  $str =~ s/\s+$//;
  $! = 0;
  my($num, $unparsed) = strtod($str);
  if (($str eq '') || ($unparsed != 0) || $!) {
    return undef;
  } else {
    return $num;
  }
}
```

./test/src/single_card_anim.pl:

```
#!/usr/bin/perl -w

use IO::Socket::INET;
use Time::HiRes qw ( time alarm sleep );

my $MAX_LIGHTS = 25;
my $UPDATE_DELAY = 0.05;

my $port = 5001;
my $ip = $ARGV[0];
#new Net::IP($ARGV[0]) or die (Net::IP::Error());

my $OP_SETLIGHT = 1;
my $OP_SETIP = 2;

print "Setting up UDP socket to: $ip:$port\n";

my $sock = new IO::Socket::INET->new(PeerPort=>$port,Proto=>'udp',PeerAddr=>$ip) or die
($!);
#IO::Socket::INET::Error());

my $in;
{
local $/;
$in = <STDIN>;
}

while(1) {
foreach $line (split(/\n/,$in)) {
  next if ($line =~ m/\s*#/);
  my($delay,@light_values) = split(/\s+/,$line);
  sleep($delay);
  $msg = pack("C"x($MAX_LIGHTS+1), $OP_SETLIGHT, @light_values);
  $sock->send($msg);
  sleep $UPDATE_DELAY;
}
}
```

./test/src/show_image.pl:

```
#!/usr/bin/perl -w

use ODC;

use strict;

my $in;
{
local $/;
$in = <STDIN>;
}

my @stride_ary;

foreach(split//,$in) {
#  if ($invert) {
    push(@stride_ary,255-ord($_));
```

```
#  } else {
#    $stride_ary[] = ord($rawimage{$i});
#  }
}

my $x=0;

my $y=0;

foreach(@stride_ary) {
  if($x==($ODC::width*$ODC::lights_x)) { $x = 0; $y++; }
#  print "$x,$y,$_\n";
  ODC::setPixel($x,$y,$_);
  $x++;
}
ODC::redraw();
exit();
```

./test/traverse.txt:

```
0 99 99 99 99 99 00 00 00 00 00 99 99 99 99 99 00 00 00 00 00 00 00 00 00 00
10.5 0 99
1.5 0 0 99
1.5 0 0 0 99
1.4 0 0 0 0 99
1.3 0 0 0 0 0 99
1.2 0 0 0 0 0 0 99
1.1 0 0 0 0 0 0 0 99
1.05 0 0 0 0 0 0 0 0 99
1.04 0 0 0 0 0 0 0 0 0 99
1.03 0 0 0 0 0 0 0 0 0 0 99
1.02 0 0 0 0 0 0 0 0 0 0 0 99
1.01 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 99
```

./test/traverse_fill.txt:

```
0 99
0 99 99
0 99 99 99
0 99 99 99 99
0 99 99 99 99 99
0 99 99 99 99 99 99
0 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
0 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99
```

./test/smiley.txt:

```
# simple input to control lights
# <sec> <light 1 value> <light 2 value> ..
2  0 99  0 99  0     0 99  0 99  0     0  0  0  0  0   99  0  0  0 99   00 99 99 99  0
2
```

./test/directions.txt:

```
0 00 00 00 00 00   00 99 99 99 00    00 99 99 00 00    00 99 00 99 00    00 00 00 00 99
10
```

./test/LICENSE:

```
Unless otherwise stated, the files in this directory and subdirectories are licensed with
a two-clause BSD License,
and copyright is held by Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>

This source file is licensed under a two-clause BSD license

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

./test/clear.txt:

```
0 0
```

./test/INSTALL:

```
The test suite does not need any installation, except you might want to copy them
elsewhere, as long as the hierarchy
is remained the test commands should be usable.

This is an early test suite made to experiment with the cards, communication and used for
debugging purposes and also
designing the real daemon-client system.

FOr more information on how to run the tests, please refer to: http://odc.opentheweb.org/
wiki/Initial_test_setup
```

./test/all.txt:

```
0        99 99 99 99 99 99    99 99 99 99 99    99 99 99 99 99   99 99 99 99 99    99 99
99 99 99
60
```

./test/README:

```
Open Digital Canvas -- test suite

Please see http://odc.openthweb.org for more information.

This is a set of tests both stand alone and using a perl module to communicate directly
with cards. One needs to run
them on a computer with direct network access to the boards that are to be tested. Check
http://odc.opentheweb.org/wiki/Initial_test_setup for more information on how to use the
tests.

Check INSTALL for instructions to install this software and LICENSE to see its license.
```

./php-extension/tests/001.phpt:

```
--TEST--
Check for odc presence
--SKIPIF--
<?php if (!extension_loaded("odc")) print "skip"; ?>
--FILE--
<?php
echo "odc extension is available";
/*
        you can add regression tests for your extension here

  the output of your test code has to be equal to the
  text in the --EXPECT-- section below for the tests
  to pass, differences between the output and the
  expected text are interpreted as failure

        see php5/README.TESTING for further information on
  writing regression tests
*/
?>
--EXPECT--
odc extension is available
```

./php-extension/config.w32:

```
// $Id$
// vim:ft=javascript

// If your extension references something external, use ARG_WITH
// ARG_WITH("odc", "for odc support", "no");

// Otherwise, use ARG_ENABLE
// ARG_ENABLE("odc", "enable odc support", "no");

if (PHP_ODC != "no") {
        EXTENSION("odc", "odc.c");
}
```

./php-extension/LICENSE:

```
Unless otherwise stated, the files in this directory and subdirectories are licensed with
a two-clause BSD License,
and copyright is held by Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>

This source file is licensed under a two-clause BSD license

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

./php-extension/odc.def:

```
resource odc_connect(string host, int port, string username, string password,int canvas);
void odc_disconnect(resource odc_handle);
int odc_get_pixel_level(resource odc_handle,int x,int y);
int odc_set_pixel_level(resource odc_handle,int x,int y,int level);
array odc_get_pixel_rgba(resource odc_handle,int x,int y);
int odc_set_pixel_rgba(resource odc_handle,int x,int y,float R,float G,float B,float A);
int odc_blit_array(resource odc_handle,int x,int y,array ary,int length, int stride);
int odc_blit_array_rgba(resource odc_handle,int x,int y,resource colorary,int length, int
stride);
int odc_is_valid_pos(resource odc_handle,int x, int y);
int odc_clear(resource odc_handle);
int odc_update(resource odc_handle);
```

./php-extension/php_odc.h:

```
/*
  +----------------------------------------------------------------------+
  | PHP Version 5                                                        |
  +----------------------------------------------------------------------+
  | Copyright (c) 1997-2007 The PHP Group                                |
  +----------------------------------------------------------------------+
  | This source file is subject to version 3.01 of the PHP license,      |
  | that is bundled with this package in the file LICENSE, and is        |
  | available through the world-wide-web at the following url:           |
  | http://www.php.net/license/3_01.txt                                  |
  | If you did not receive a copy of the PHP license and are unable to   |
  | obtain it through the world-wide-web, please send a note to          |
  | license@php.net so we can mail you a copy immediately.               |
  +----------------------------------------------------------------------+
  | Author:                                                              |
  +----------------------------------------------------------------------+
*/

/* $Id: php_odc.h 63 2007-06-22 00:50:01Z mendoza $ */

#ifndef PHP_ODC_H
#define PHP_ODC_H

extern zend_module_entry odc_module_entry;
#define phpext_odc_ptr &odc_module_entry
```

```
#ifdef PHP_WIN32
#define PHP_ODC_API __declspec(dllexport)
#else
#define PHP_ODC_API
#endif

#ifdef ZTS
#include "TSRM.h"
#endif

PHP_MINIT_FUNCTION(odc);
PHP_MSHUTDOWN_FUNCTION(odc);
PHP_RINIT_FUNCTION(odc);
PHP_RSHUTDOWN_FUNCTION(odc);
PHP_MINFO_FUNCTION(odc);

PHP_FUNCTION(confirm_odc_compiled);        /* For testing, remove later. */
PHP_FUNCTION(odc_connect);
PHP_FUNCTION(odc_disconnect);
PHP_FUNCTION(odc_get_pixel_level);
PHP_FUNCTION(odc_set_pixel_level);
PHP_FUNCTION(odc_get_pixel_rgba);
PHP_FUNCTION(odc_set_pixel_rgba);
PHP_FUNCTION(odc_blit_array);
PHP_FUNCTION(odc_blit_array_rgba);
PHP_FUNCTION(odc_is_valid_pos);
PHP_FUNCTION(odc_clear);
PHP_FUNCTION(odc_update);

/*
          Declare any global variables you may need between the BEGIN
        and END macros here:

ZEND_BEGIN_MODULE_GLOBALS(odc)
        long  global_value;
        char *global_string;
ZEND_END_MODULE_GLOBALS(odc)
*/

#define PHP_ODC_VERSION "1.0"
#define PHP_ODC_EXTNAME "odc"

#define PHP_ODC_RES_NAME "ODC Handle"

/* In every utility function you add that needs to use variables
   in php_odc_globals, call TSRMLS_FETCH(); after declaring other
   variables used by that function, or better yet, pass in TSRMLS_CC
   after the last function argument and declare your utility function
   with TSRMLS_DC after the last declared argument.  Always refer to
   the globals in your function as ODC_G(variable).  You are
   encouraged to rename these macros something shorter, see
   examples in any other php module directory.
*/

#ifdef ZTS
#define ODC_G(v) TSRMG(odc_globals_id, zend_odc_globals *, v)
#else
#define ODC_G(v) (odc_globals.v)
#endif

#include "odc.h"

#endif        /* PHP_ODC_H */


/*
 * Local variables:
 * tab-width: 4
 * c-basic-offset: 4
 * End:
 * vim600: noet sw=4 ts=4 fdm=marker
 * vim<600: noet sw=4 ts=4
 */
```

./php-extension/config.m4:

```
dnl $Id$
dnl config.m4 for extension odc
```

```
dnl Comments in this file start with the string 'dnl'.
dnl Remove where necessary. This file will not work
dnl without editing.

dnl If your extension references something external, use with:

PHP_ARG_WITH(odc, for odc support,
Make sure that the comment is aligned:
[  --with-odc             Include odc support])

dnl Otherwise use enable:

dnl PHP_ARG_ENABLE(odc, whether to enable odc support,
dnl Make sure that the comment is aligned:
dnl [  --enable-odc           Enable odc support])

if test "$PHP_ODC" != "no"; then
  dnl Write more examples of tests here...

  # --with-odc -> check with-path
  SEARCH_PATH="/usr/local /usr"     # you might want to change this
  SEARCH_FOR="/include/odc.h"  # you most likely want to change this
  if test -r $PHP_ODC/$SEARCH_FOR; then # path given as parameter
    ODC_DIR=$PHP_ODC
  else # search default path list
    AC_MSG_CHECKING([for odc files in default path])
    for i in $SEARCH_PATH ; do
      if test -r $i/$SEARCH_FOR; then
        ODC_DIR=$i
        AC_MSG_RESULT(found in $i)
      fi
    done
  fi
  dnl
  if test -z "$ODC_DIR"; then
    AC_MSG_RESULT([not found])
    AC_MSG_ERROR([Please reinstall the odc distribution])
  fi

  # --with-odc -> add include path
  PHP_ADD_INCLUDE($ODC_DIR/include)

  # --with-odc -> check for lib and symbol presence
  LIBNAME=odc # you may want to change this
  LIBSYMBOL=odc_connect # you most likely want to change this

  PHP_CHECK_LIBRARY($LIBNAME,$LIBSYMBOL,
  [
    PHP_ADD_LIBRARY_WITH_PATH($LIBNAME, $ODC_DIR/lib, ODC_SHARED_LIBADD)
    AC_DEFINE(HAVE_ODCLIB,1,[ ])
  ],[
    AC_MSG_ERROR([wrong odc lib version or lib not found])
  ],[
    -L$ODC_DIR/lib -lm -ldl
  ])
  dnl
  PHP_SUBST(ODC_SHARED_LIBADD)

  PHP_NEW_EXTENSION(odc, odc.c, $ext_shared)
fi
```

./php-extension/INSTALL:

```
1. run phpize (requires installation of the php5-dev package)
2. ./configure
3. make
4. make install
5. Use module either with dl("odc.so") in a php script or by adding "extension=odc.so" in
your php.ini
   You may run a test after installing the module, by executing "php odc.php".
```

./php-extension/odc.php:

```
<?php
dl("odc.so");
$br = (php_sapi_name() == "cli")? "":"<br>";

if(!extension_loaded('odc')) {
        dl('odc.' . PHP_SHLIB_SUFFIX);
}
$module = 'odc';
$functions = get_extension_funcs($module);
echo "Functions available in the test extension:$br\n";
foreach($functions as $func) {
    echo $func."$br\n";
}
echo "$br\n";
$function = 'confirm_' . $module . '_compiled';
if (extension_loaded($module)) {
        $str = $function($module);
} else {
        $str = "Module $module is not compiled into PHP";
}
echo "$str\n";

$handler = odc_connect("129.241.110.149",5001,"","",0);

odc_clear($handler);
odc_update($handler);

odc_set_pixel_level($handler,5,5,255);
odc_set_pixel_rgba($handler,6,6,0.5,0.5,0.5,0.5);

odc_update($handler);

odc_disconnect($handler);
?>
```

./php-extension/odc.c:

```
/*
  +----------------------------------------------------------------------+
  | PHP Version 5                                                        |
  +----------------------------------------------------------------------+
  | Copyright (c) 1997-2007 The PHP Group                                |
  +----------------------------------------------------------------------+
  | This source file is subject to version 3.01 of the PHP license,      |
  | that is bundled with this package in the file LICENSE, and is        |
  | available through the world-wide-web at the following url:           |
  | http://www.php.net/license/3_01.txt                                  |
  | If you did not receive a copy of the PHP license and are unable to   |
  | obtain it through the world-wide-web, please send a note to          |
  | license@php.net so we can mail you a copy immediately.               |
  +----------------------------------------------------------------------+
  | Author:                                                              |
  +----------------------------------------------------------------------+
*/

/* $Id: odc.c 75 2007-06-22 03:06:19Z mendoza $ */

#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include "php.h"
#include "php_ini.h"
#include "ext/standard/info.h"
#include "php_odc.h"

/* If you declare any globals in php_odc.h uncomment this:
ZEND_DECLARE_MODULE_GLOBALS(odc)
*/

/* True global resources - no need for thread safety here */
static int le_odc;

/* {{{ odc_functions[]
```

```
 *
 * Every user visible function must have an entry in odc_functions[].
 */
zend_function_entry odc_functions[] = {
        PHP_FE(confirm_odc_compiled,        NULL)                /* For testing, remove
later. */
        PHP_FE(odc_connect,        NULL)
        PHP_FE(odc_disconnect,        NULL)
        PHP_FE(odc_get_pixel_level,        NULL)
        PHP_FE(odc_set_pixel_level,        NULL)
        PHP_FE(odc_get_pixel_rgba,        NULL)
        PHP_FE(odc_set_pixel_rgba,        NULL)
        PHP_FE(odc_blit_array,        NULL)
        PHP_FE(odc_blit_array_rgba,        NULL)
        PHP_FE(odc_is_valid_pos,        NULL)
        PHP_FE(odc_clear,        NULL)
        PHP_FE(odc_update,        NULL)
        {NULL, NULL, NULL}        /* Must be the last line in odc_functions[] */
};
/* }}} */

/* {{{ odc_module_entry
 */
zend_module_entry odc_module_entry = {
#if ZEND_MODULE_API_NO >= 20010901
        STANDARD_MODULE_HEADER,
#endif
        "odc",
        odc_functions,
        PHP_MINIT(odc),
        PHP_MSHUTDOWN(odc),
        PHP_RINIT(odc),                /* Replace with NULL if there's nothing to do at
request start */
        PHP_RSHUTDOWN(odc),        /* Replace with NULL if there's nothing to do at
request end */
        PHP_MINFO(odc),
#if ZEND_MODULE_API_NO >= 20010901
        "0.1", /* Replace with version number for your extension */
#endif
        STANDARD_MODULE_PROPERTIES
};
/* }}} */

#ifdef COMPILE_DL_ODC
ZEND_GET_MODULE(odc)
#endif

/* {{{ PHP_INI
 */
/* Remove comments and fill if you need to have entries in php.ini
PHP_INI_BEGIN()
    STD_PHP_INI_ENTRY("odc.global_value",      "42", PHP_INI_ALL, OnUpdateLong,
global_value, zend_odc_globals, odc_globals)
    STD_PHP_INI_ENTRY("odc.global_string", "foobar", PHP_INI_ALL, OnUpdateString,
global_string, zend_odc_globals, odc_globals)
PHP_INI_END()
*/
/* }}} */

/* {{{ php_odc_init_globals
 */
/* Uncomment this function if you have INI entries
static void php_odc_init_globals(zend_odc_globals *odc_globals)
{
        odc_globals->global_value = 0;
        odc_globals->global_string = NULL;
}
*/
/* }}} */

static void php_odc_c_dtor(zend_rsrc_list_entry *rsrc TSRMLS_DC)
{
    struct odc_connection *odc_c = (struct odc_connection *)rsrc->ptr;

    if (odc_c) {
      efree(odc_c);
    }
}


/* {{{ PHP_MINIT_FUNCTION
 */
```

```
PHP_MINIT_FUNCTION(odc)
{
        /* If you have INI entries, uncomment these lines
        REGISTER_INI_ENTRIES();
        */
    le_odc = zend_register_list_destructors_ex(php_odc_c_dtor, NULL, PHP_ODC_RES_NAME,
module_number);

        return SUCCESS;
}
/* }}} */



/* {{{ PHP_MSHUTDOWN_FUNCTION
 */
PHP_MSHUTDOWN_FUNCTION(odc)
{
        /* uncomment this line if you have INI entries
        UNREGISTER_INI_ENTRIES();
        */
        return SUCCESS;
}
/* }}} */

/* Remove if there's nothing to do at request start */
/* {{{ PHP_RINIT_FUNCTION
 */
PHP_RINIT_FUNCTION(odc)
{
        return SUCCESS;
}
/* }}} */

/* Remove if there's nothing to do at request end */
/* {{{ PHP_RSHUTDOWN_FUNCTION
 */
PHP_RSHUTDOWN_FUNCTION(odc)
{
        return SUCCESS;
}
/* }}} */

/* {{{ PHP_MINFO_FUNCTION
 */
PHP_MINFO_FUNCTION(odc)
{
        php_info_print_table_start();
        php_info_print_table_header(2, "odc support", "enabled");
        php_info_print_table_end();

        /* Remove comments if you have entries in php.ini
        DISPLAY_INI_ENTRIES();
        */
}
/* }}} */


/* Remove the following function when you have succesfully modified config.m4
   so that your module can be compiled into PHP, it exists only for testing
   purposes. */

/* Every user-visible function in PHP should document itself in the source */
/* {{{ proto string confirm_odc_handleompiled(string arg)
   Return a string to confirm that the module is compiled in */
PHP_FUNCTION(confirm_odc_compiled)
{
        char *arg = NULL;
        int arg_len, len;
        char string[256];

        if (zend_parse_parameters(ZEND_NUM_ARGS() TSRMLS_CC, "s", &arg, &arg_len) ==
FAILURE) {
                return;
        }

        len = sprintf(string, "Congratulations! You have successfully modified ext/%.78s/
config.m4. Module %.78s is now compiled into PHP.", "odc", arg);
        RETURN_STRINGL(string, len, 1);
}
/* }}} */
/* The previous line is meant for vim and emacs, so it can correctly fold and
```

```
   unfold functions in source code. See the corresponding marks just before
   function definition, where the functions purpose is also documented. Please
   follow this convention for the convenience of others editing your code.
*/

/* {{{ proto resource odc_connect(string host, int port, string username, string password,
int canvas)
   ; */
PHP_FUNCTION(odc_connect)
{
        char *host = NULL;
        char *username = NULL;
        char *password = NULL;
        int argc = ZEND_NUM_ARGS();
        int host_len;
        int username_len;
        int password_len;
        long port;
        long canvas;

        struct odc_connection *odc_c = emalloc(sizeof(struct odc_connection));

        if (zend_parse_parameters(argc TSRMLS_CC, "slssl", &host, &host_len, &port,
&username, &username_len, &password, &password_len, &canvas) == FAILURE) {
          RETURN_FALSE;
        }

        if ((odc_c = odc_connect(odc_c,host,port,username,password,canvas)) == NULL) {
          php_error(E_ERROR, "odc_connect: unable to connect server");
        }

        ZEND_REGISTER_RESOURCE(return_value, odc_c, le_odc);
}
/* }}} */

/* {{{ proto void odc_disconnect(resource odc_handle)
   ; */
PHP_FUNCTION(odc_disconnect)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        zval *odc_handle = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "r", &odc_handle) == FAILURE)
                RETURN_FALSE;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }

        odc_disconnect(odc_c);
        zend_list_delete(Z_LVAL_P(odc_handle));
        RETURN_TRUE;
}
/* }}} */

/* {{{ proto int odc_get_pixel_level(resource odc_handle, int x, int y)
   ; */
PHP_FUNCTION(odc_get_pixel_level)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        long x;
        long y;
        zval *odc_handle = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "rll", &odc_handle, &x, &y) == FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }

        php_error(E_WARNING, "odc_get_pixel_level: not yet implemented");
}
/* }}} */
```

```
/* {{{ proto int odc_set_pixel_level(resource odc_handle, int x, int y, int level)
   ; */
PHP_FUNCTION(odc_set_pixel_level)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        long x;
        long y;
        long level;
        zval *odc_handle = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "rlll", &odc_handle, &x, &y, &level) ==
FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }

        odc_set_pixel_level(odc_c,x,y,level);
}
/* }}} */

/* {{{ proto array odc_get_pixel_rgba(resource odc_handle, int x, int y)
   ; */
PHP_FUNCTION(odc_get_pixel_rgba)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        long x;
        long y;
        zval *odc_handle = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "rll", &odc_handle, &x, &y) == FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }

        php_error(E_WARNING, "odc_get_pixel_rgba: not yet implemented");
}
/* }}} */

/* {{{ proto int odc_set_pixel_rgba(resource odc_handle, int x, int y, float R, float G,
float B, float A)
   ; */
PHP_FUNCTION(odc_set_pixel_rgba)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        long x;
        long y;
        double R;
        double G;
        double B;
        double A;
        zval *odc_handle = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "rllddddd", &odc_handle, &x, &y, &R, &G,
&B, &A) == FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }

        odc_set_pixel_rgba(odc_c,x,y,R,G,B,A);
}
/* }}} */

/* {{{ proto int odc_blit_array(resource odc_handle, int x, int y, array ary, int length,
```

```
int stride)
   ; */
PHP_FUNCTION(odc_blit_array)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        long x;
        long y;
        long length;
        long stride;
        zval *odc_handle = NULL;
        zval *ary = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "rllall", &odc_handle, &x, &y, &ary,
&length, &stride) == FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }

        php_error(E_WARNING, "odc_blit_array: not yet implemented");
}
/* }}} */

/* {{{ proto int odc_blit_array_rgba(resource odc_handle, int x, int y, resource colorary,
int length, int stride)
   ;     */
PHP_FUNCTION(odc_blit_array_rgba)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        int colorary_id = -1;
        long x;
        long y;
        long length;
        long stride;
        zval *odc_handle = NULL;
        zval *colorary = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "rllrll", &odc_handle, &x, &y,
&colorary, &length, &stride) ==
FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }
        if (colorary) {
/*                ZEND_FETCH_RESOURCE(???, ???, colorary, colorary_id, "???",
???_rsrc_id); */
        }

        php_error(E_WARNING, "odc_blit_array_rgba: not yet implemented");
}
/* }}} */

/* {{{ proto int odc_is_valid_pos(resource odc_handle, int x, int y)
   ; */
PHP_FUNCTION(odc_is_valid_pos)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        long x;
        long y;
        zval *odc_handle = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "rll", &odc_handle, &x, &y) == FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }
```

```
        php_error(E_WARNING, "odc_is_valid_pos: not yet implemented");
}
/* }}} */

/* {{{ proto int odc_clear(resource odc_handle)
   ; */
PHP_FUNCTION(odc_clear)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        zval *odc_handle = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "r", &odc_handle) == FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }

        odc_clear(odc_c);
}
/* }}} */

/* {{{ proto int odc_update(resource odc_handle)
   ; */
PHP_FUNCTION(odc_update)
{
        int argc = ZEND_NUM_ARGS();
        int odc_handle_id = -1;
        zval *odc_handle = NULL;

        struct odc_connection *odc_c;

        if (zend_parse_parameters(argc TSRMLS_CC, "r", &odc_handle) == FAILURE)
                return;

        if (odc_handle) {
                ZEND_FETCH_RESOURCE(odc_c, struct odc_connection *, &odc_handle,
odc_handle_id, PHP_ODC_RES_NAME, le_odc);
        }

        odc_update(odc_c);
}
/* }}} */


/*
 * Local variables:
 * tab-width: 4
 * c-basic-offset: 4
 * End:
 * vim600: noet sw=4 ts=4 fdm=marker
 * vim<600: noet sw=4 ts=4
 */
```

./php-extension/README:

```
Open Digital Canvas -- php-extension

This is an example of usage of the libodc library which in turn connects to an ODC server
implemented as a php
module. One should be able to make python modules, hook up with Java or the like.

Please see http://odc.openthweb.org for more information.

Check INSTALL for instrutions to install this software and LICENSE to see its license.
```

./daemon/src/Daemon.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2007 <mendoza@pvv.ntnu.no>
 *
 */

#include <string>

#include "Daemon.h"
#include "pack.h"
#include "odc.h"

/**
 * processNode:
 * @reader: the xmlReader
 *
 * Dump information about the current node
 */

void Daemon::processNode(xmlTextReaderPtr reader)
{
  const xmlChar *name, *value;

  name = xmlTextReaderConstName(reader);
  if (name == NULL)
    name = BAD_CAST "--";
  value = xmlTextReaderConstValue(reader);

  if ((xmlTextReaderNodeType(reader) == 1)) {
    if (0 == std::string((char *) name).compare("canvas")) {
      xmlChar *width;

      if ((width = xmlTextReaderGetAttribute(reader, xmlCharStrdup("width")))) {
        Canvas myCanvas;

        myCanvas.width = atoi((char *) width);
        this->canvases.push_back(myCanvas);
      } else {
        ODC_LOG("canvas element is missing width attribute\n");
      }
    } else if (0 == std::string((char *) name).compare("board")) {
      xmlChar *direction, *ip;

      if ((direction = xmlTextReaderGetAttribute(reader, xmlCharStrdup("direction")))
          && (ip = xmlTextReaderGetAttribute(reader, xmlCharStrdup("ip")))) {
        Board myBoard;

        myBoard.direction = (char *) direction;
        myBoard.ip = (char *) ip;
        ODC_LOG("%s %s\n", ip, direction);
        if (this->canvases.size() > 0) {
          ODC_LOG("lol\n");
          (this->canvases.end() - 1)->boards.push_back(myBoard);
        }
      }
    }
    // printf("1 -- %d %p\n",this->canvases.size(),this->canvases.end());
```

```
    if ((this->canvases.size() > 0)
        && ((this->canvases.end() - 1)->boards.size() > 0)) {
      ODC_LOG("2 -- %d\n", ((this->canvases.end() - 1)->boards.end() - 1)->lightsX);
    }
  }
}

bool Daemon::addClient(Client *client) {
  clients.insert(std::pair < int, Client * >(client->fd, client));
  return true;
}

bool Daemon::removeClient(int key) {
  if (!clients.erase(key)) { return false; }
  return true;
}
```

./daemon/src/Comm.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <iostream>
#include <stdio.h>

#include "Comm.h"

bool Comm::updateBoard(char *ip, char *packet) {
  int sock;
  struct sockaddr_in server_addr;
  server_addr.sin_family = AF_INET;
  server_addr.sin_port = htons(BOARD_UDP_PORT);
  printf("sending '");
  for(int i=0;i<=MAX_LIGHTS;i++) {
          printf("%d ",packet[i]);
  }
  printf("' to %s\n",ip);
  printf("'%s' to %s\n",packet,ip);
  if (inet_aton(ip,&server_addr.sin_addr)) {
    sock = socket(PF_INET, SOCK_DGRAM, 0);
    sendto(sock,packet,MAX_LIGHTS+1,0, (struct sockaddr *) &server_addr,sizeof(struct
sockaddr_in));
    close(sock);
  } else {
    std::cerr << "Illegal ip address: " << ip << "\n";
  }
  return true;
}
```

./daemon/src/Comm.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef __ODCCOMM_H
#define __ODCCOMM_H
#include "Daemon.h"
//#include "Board.h"

#define BOARD_UDP_PORT 5001


class Comm {
  //connections
  public:
  static bool updateBoard(char* ip, char* packet);
  static bool authorize(char *user, char* passwd);
  static bool isAuthorized(int fd);
  };
#endif
```

./daemon/src/Canvas.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#include "Canvas.h"
```

```
#include <iostream>

int Canvas::getPixelLevel(int x,int y) {
  return getPixelRGBA(x,y).getLevel();
}

bool Canvas::setPixelLevel(int x,int y,int level) {
  return setPixelRGBA(x,y,(float)level/MAX_LEVEL,(float)level/MAX_LEVEL,(float)level/
MAX_LEVEL,1.0);
}

Color Canvas::getPixelRGBA(int x,int y) {
  if (isValidPos(x,y)) {
    int bx = (int)(x/LIGHTS_X);
    int by = (int)(y/LIGHTS_Y);
    int cindex = bx+(by*width);
    return boards[cindex].getPixelRGBA(x,y);
  }
  return 0;
}

bool Canvas::setPixelRGBA(int x,int y,float R,float G,float B,float A) {
  if (isValidPos(x,y)) {
    int bx = (int)(x/LIGHTS_X);
    int by = (int)(y/LIGHTS_Y);
    int cindex = bx+(by*width);
        printf("%d %d %d\n",bx,by,cindex);
    return boards[cindex].setPixelRGBA(x,y,R,G,B,A);
  }
  printf("%d %d invalid\n",x,y);
  return false;
}

bool Canvas::blitArray(int x,int y,int* array,int length, int stride) {
  for(int i=0;i<length;i++) {
    setPixelLevel(i%stride,i/stride,array[i]);
  }

  return true;

  }

bool Canvas::blitArrayRGBA(int x,int y,int* array,int length, int stride) {
  for(int i=0;i<length;i++) {
    setPixelRGBA(i%stride, i/stride,
    (( array[i] >> 24 ) & 0xFF),
    (( array[i] >> 16 ) & 0xFF),
    (( array[i] >> 8 ) & 0xFF),
    (( array[i] >> 0 ) & 0xFF));
  }
  return false;
}

bool Canvas::clear() {
  for(std::vector<int>::size_type i=0;i<boards.size();i++) {
    boards[i].clear();
  }
  return true;
}

bool Canvas::update() {
  for(std::vector<int>::size_type i=0;i<boards.size();i++) {
    boards[i].update();
  }
  return true;
}

bool Canvas::isValidPos(int x, int y) {
  return ((x>-1) && (x < (width*LIGHTS_X)) && (y >-1) && (y < (int)((boards.size()/
width)*LIGHTS_Y)));
}
```

./daemon/src/Daemon.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
```

```
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2007 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef __ODCDAEMON_H
#define __ODCDAEMON_H

#include "Canvas.h"
#include "Client.h"
#include "libxml/xmlreader.h"
#include <vector>
#include <map>

class Daemon {
public:
  std::map < int, Client * >clients;
      std::map < int, Client * >::iterator it;
      std::vector < Canvas > canvases;
  void processNode(xmlTextReaderPtr);
  bool addClient(Client *client);
bool removeClient(int key);
};

#endif
```

./daemon/src/Client.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2007 <mendoza@pvv.ntnu.no>
 *
 */

#include "Client.h"
#include "Daemon.h"
#include <sys/types.h>
#include <sys/socket.h>
```

```
extern Daemon odcd;

bool Client::authenticate(unsigned char *input, int inputSize)
{
  return true;
}

bool Client::greet()
{
  unsigned char buf[1024];
  size_t packetsize;

  std::cout << "Greeting " << fd << "\n";
  packetsize = pack(buf, "hhs", PROTOCOL_VERSION, SRV_AUTH_REQUIRED, WELCOME_MSG);
  //packi32(buf, packetsize);     // we don't bother with packet sizes for now
  if (send(fd, buf, packetsize, 0) == -1) {
    perror("send greeting");
    return false;
  }
  return true;
}

bool Client::sendCommand(int cmd)
{
  unsigned char buf[1024];
  size_t packetsize;

  packetsize = pack(buf, "h", cmd);
  //packi32(buf, packetsize);     // we don't bother with packet sizes for now
  if (send(fd, buf, packetsize, 0) == -1) {
    perror("send packet");
    return false;
  }
  return true;
}

bool Client::handleInput(unsigned char *input, int inputSize)
{
  short x, y, level;
  int arraysize, arraystride;
  float r, g, b, a;
  Color color(0);
  unsigned char buf[1024];
  size_t packetsize;
  short cmd;

  /*
   * char *cmd = (char *) malloc(sizeof(char) * (inputSize + 1));
   *
   * if (*cmd) { strncpy(cmd, input, inputSize); cmd[inputSize] = 0; sscanf(cmd, "%d %d %d
%d\n", &c, &x, &y, &l); printf("handling: %d %d %d %d\n", c, x, y, l);
this->canvases[c].setPixelLevel(x, y,
   * l); this->canvases[c].update(); printf("%s", cmd); free(cmd); return true; } else {
return false; }
   */
  unpack(input, "h", &cmd);

  if (authenticated) {

    ODC_LOG("Got a %hd command\n", cmd);

    switch (cmd) {

    case CLT_QUIT:
      // not implemented (need to refactor to have access to client)
      break;
    case CLT_SETPIXELLEVEL:
      unpack(input, "hhhh", &cmd, &x, &y, &level);
      // we must add a fix here so that when a wrong packet arrives and all values are 0,
it wouldn't mean that (0,0) gets value 0.
      // trust blindly for now...
      ODC_LOG("setpixellevel: %hd %hd %hd %hd\n", cmd, x, y, level);
      odcd.canvases[0].setPixelLevel(x, y, level);
      break;
    case CLT_SETPIXELRGBA:
      unpack(input, "hhhffff", &cmd, &x, &y, &r, &g, &b, &a);
      // we must add a fix here so that when a wrong packet arrives and all values are 0,
it wouldn't mean that (0,0) gets value 0.
      // trust blindly for now...
      ODC_LOG("setpixellevel: %hd %hd %hd %f %f %f %f\n", cmd, x, y, r, g, b, a);
      odcd.canvases[0].setPixelRGBA(x, y, r, g, b, a);
      break;
```

```
    case CLT_CLEAR:
      if (!odcd.canvases[0].clear()) {
        return false;
      }
      break;
    case CLT_UPDATE:
      if (!odcd.canvases[0].update()) {
        return false;
      }
      break;
    case CLT_AUTOUPDATEMODE:
      // always update after a command, for lazy people, and perhaps useful for blitting
arrays
      break;
    case CLT_GETPIXELLEVEL:
      // should send a proper error code when out of range
      unpack(input, "hhh", &cmd, &x, &y);
      level = odcd.canvases[0].getPixelLevel(x, y);
      packetsize = pack(buf, "hh", SRV_LEVEL, level);
      //packi32(buf, packetsize);     // we don't bother with packet sizes for now
      if (send(fd, buf, packetsize, 0) == -1) {
        perror("send packet");
        return false;
      }
      break;
    case CLT_GETPIXELRGBA:
      // should send a proper error code when out of range
      unpack(input, "hhh", &cmd, &x, &y);
      color = odcd.canvases[0].getPixelRGBA(x, y);
      packetsize = pack(buf, "hhfffff", SRV_RGBA, color.r, color.g, color.b, color.a);
      //packi32(buf, packetsize);     // we don't bother with packet sizes for now
      if (send(fd, buf, packetsize, 0) == -1) {
        perror("send packet");
        return false;
      }
      break;
    case CLT_BLITARRAYLEVEL:
      unpack(input, "hii", &cmd, &arraysize, &arraystride);
//      bool blitArray(int x, int y, int *array, int length, int stride);
      // FIXME: read in array by size
    // we need to make a proper recv loop to fetch big packages (larger than 8k for
loopback or 1500-ish for ethernet)
      break;

    case CLT_BLITARRAYRGBA:
      unpack(input, "hii", &cmd, &arraysize, &arraystride);
//      bool blitArrayRGBA(int x, int y, int *array, int length, int stride);
    // we need to make a proper recv loop to fetch big packages (larger than 8k for
loopback or 1500-ish for ethernet)
      // FIXME: read in array by size
      break;
    default:
      std::cerr << "Unknown command:" << cmd << std::endl;
      return false;
      break;
    }
    if (sendCommand(SRV_OK) == -1) {
      perror("send ok");
    }
    ODC_LOG("%hd %p\n", cmd, input);
  } else {
    if (remoteAddress == "127.0.0.1") {
      authType = AUTH_TYPE_ADMIN;       // admin type
      authenticated = true;
      std::cout << "Connection fd: " << fd << " with ip: " << remoteAddress << "
authenticated as admin\n";
      if (sendCommand(SRV_AUTH_ACCEPTED) == -1) {
        perror("send");
      }
    } else if (authenticate((unsigned char *) input, inputSize)) {
      authType = AUTH_TYPE_USER;
      authenticated = true;
      std::cout << "Connection fd: " << fd << " with ip: " << remoteAddress << "
authenticated as user\n";
      if (sendCommand(SRV_AUTH_ACCEPTED) == -1) {
        perror("send");
      }
    } else {
      // invalid authentication
      if (sendCommand(SRV_AUTH_ERR) == -1) {
        perror("send");
      }
```

```
    }
  }
  return true;
}
```

./daemon/src/pack.c:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#ifdef __cplusplus
extern "C" {
#endif

#include <stdio.h>
#include <ctype.h>
#include <stdarg.h>
#include <string.h>
#include <stdint.h>

#include "pack.h"

/*
** pack754() -- pack a floating point number into IEEE-754 format
*/
long long pack754(long double f, unsigned bits, unsigned expbits)
{
        long double fnorm;
        int shift;
        long long sign, exp, significand;
        unsigned significandbits = bits - expbits - 1; // -1 for sign bit

        if (f == 0.0) return 0; // get this special case out of the way

        // check sign and begin normalization
        if (f < 0) { sign = 1; fnorm = -f; }
        else { sign = 0; fnorm = f; }

        // get the normalized form of f and track the exponent
        shift = 0;
        while(fnorm >= 2.0) { fnorm /= 2.0; shift++; }
        while(fnorm < 1.0) { fnorm *= 2.0; shift--; }
        fnorm = fnorm - 1.0;

        // calculate the binary form (non-float) of the significand data
        significand = (long long) (fnorm * ((1LL<<significandbits) + 0.5f));

        // get the biased exponent
        exp = shift + ((1<<(expbits-1)) - 1); // shift + bias

        // return the final answer
        return (sign<<(bits-1)) | (exp<<(bits-expbits-1)) | significand;
}
```

```
/*
** unpack754() -- unpack a floating point number from IEEE-754 format
*/
long double unpack754(long long i, unsigned bits, unsigned expbits)
{
        long double result;
        long long shift;
        unsigned bias;
        unsigned significandbits = bits - expbits - 1; // -1 for sign bit

        if (i == 0) return 0.0;

        // pull the significand
        result = (i&((1LL<<significandbits)-1)); // mask
        result /= (1LL<<significandbits); // convert back to float
        result += 1.0f; // add the one back on

        // deal with the exponent
        bias = (1<<(expbits-1)) - 1;
        shift = ((i>>significandbits)&((1LL<<expbits)-1)) - bias;
        while(shift > 0) { result *= 2.0; shift--; }
        while(shift < 0) { result /= 2.0; shift++; }

        // sign it
        result *= (i>>(bits-1))&1? -1.0: 1.0;

        return result;
}

/*
** packi16() -- store a 16-bit int into a char buffer (like htons())
*/
void packi16(unsigned char *buf, unsigned int i)
{
        *buf++ = i>>8; *buf++ = i;
}

/*
** packi32() -- store a 32-bit int into a char buffer (like htonl())
*/
void packi32(unsigned char *buf, unsigned long i)
{
        *buf++ = i>>24; *buf++ = i>>16;
        *buf++ = i>>8;  *buf++ = i;
}

/*
** unpacki16() -- unpack a 16-bit int from a char buffer (like ntohs())
*/
unsigned int unpacki16(unsigned char *buf)
{
        return (buf[0]<<8) | buf[1];
}

/*
** unpacki32() -- unpack a 32-bit int from a char buffer (like ntohl())
*/
unsigned long unpacki32(unsigned char *buf)
{
        return (buf[0]<<24) | (buf[1]<<16) | (buf[2]<<8) | buf[3];
}

/*
** pack() -- store data dictated by the format string in the buffer
**
**  h - 16-bit              l - 32-bit
**  c - 8-bit char          f - float, 32-bit
**  s - string (16-bit length is automatically prepended)
*/
size_t pack(unsigned char *buf, char *format, ...)
{
        va_list ap;
        int h;
        int l;
        char c;
        float f;
        char *s;
        size_t size = 0, len;

        va_start(ap, format);

        for(; *format != '\0'; format++) {
```

```
                switch(*format) {
                case 'h': // 16-bit
                        size += 2;
                        h = va_arg(ap, int); // promoted
                        packi16(buf, h);
                        buf += 2;
                        break;

                case 'l': // 32-bit
                        size += 4;
                        l = va_arg(ap, int);
                        packi32(buf, l);
                        buf += 4;
                        break;

                case 'c': // 8-bit
                        size += 1;
                        c = va_arg(ap, int); // promoted
                        *buf++ = (c>>0)&0xff;
                        break;

                case 'f': // float
                        size += 4;
                        f = va_arg(ap, double); // promoted
                        l = pack754_32(f); // convert to IEEE 754
                        packi32(buf, l);
                        buf += 4;
                        break;

                case 's': // string
                        s = va_arg(ap, char*);
                        len = strlen(s);
                        size += len + 2;
                        packi16(buf, len);
                        buf += 2;
                        memcpy(buf, s, len);
                        buf += len;
                        break;
                }
        }

        va_end(ap);

        return size;
}

/*
** unpack() -- unpack data dictated by the format string into the buffer
*/
void unpack(unsigned char *buf, char *format, ...)
{
        va_list ap;
        short *h;
        int *l;
        int pf;
        char *c;
        float *f;
        char *s;
        size_t len, count, maxstrlen=0;

        va_start(ap, format);

        for(; *format != '\0'; format++) {
                switch(*format) {
                case 'h': // 16-bit
                        h = va_arg(ap, short*);
                        *h = unpacki16(buf);
                        buf += 2;
                        break;

                case 'l': // 32-bit
                        l = va_arg(ap, int*);
                        *l = unpacki32(buf);
                        buf += 4;
                        break;

                case 'c': // 8-bit
                        c = va_arg(ap, char*);
                        *c = *buf++;
                        break;

                case 'f': // float
```

```
                        f = va_arg(ap, float*);
                        pf = unpacki32(buf);
                        buf += 4;
                        *f = unpack754_32(pf);
                        break;

                case 's': // string
                        s = va_arg(ap, char*);
                        len = unpacki16(buf);
                        buf += 2;
                        if (maxstrlen > 0 && len > maxstrlen) count = maxstrlen - 1;
                        else count = len;
                        memcpy(s, buf, count);
                        s[count] = '\0';
                        buf += len;
                        break;

                default:
                        if (isdigit(*format)) { // track max str len
                                maxstrlen = maxstrlen * 10 + (*format-'0');
                        }
                }

                if (!isdigit(*format)) maxstrlen = 0;
        }

        va_end(ap);
}
/*
int main(void)
{
        unsigned char buf[1024];
        char magic;
        short monkeycount;
        long altitude;
        float absurdityfactor;
        char *s = "Great unmitigated Zot!  You've found the Runestaff!";
        char s2[96];
        size_t packetsize, ps2;

        packetsize = pack(buf, "chhlsf", 'B', 0, 37, -5, s, -3490.6677);
        packi16(buf+1, packetsize); // store packet size in packet for kicks

        printf("packet is %d bytes\n", packetsize);

        unpack(buf, "chhl96sf", &magic, &ps2, &monkeycount, &altitude, s2,
                &absurdityfactor);

        printf("'%c' %d %d %ld \"%s\" %f\n", magic, ps2, monkeycount, altitude,
                s2, absurdityfactor);

        return 0;
}

*/

#ifdef __cplusplus
}
#endif
```

./daemon/src/Canvas.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
```

```
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef __ODCCANVAS_H
#define __ODCCANVAS_H

#include <vector>
#include "Board.h"


class Canvas {
  private:
    int RGBA2level(float R,float G,float B,float A);
  public:
    int width;
    std::vector<Board> boards;

    int getPixelLevel(int x,int y);
    bool setPixelLevel(int x,int y,int level);
    Color getPixelRGBA(int x,int y);
    bool setPixelRGBA(int x,int y,float R,float G,float B,float A);
    bool blitArray(int x,int y,int* array,int length, int stride);
    bool blitArrayRGBA(int x,int y,int* array,int length, int stride);
    bool isValidPos(int x, int y);
    bool clear();
    bool update();
};

#endif
```

./daemon/src/Color.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#include "Color.h"
#include "Board.h"

#include <iostream>
#include <math.h>

Color::Color(float r,float g,float b,float a) {
   this->r=r;
   this->g=g;
   this->b=b;
   this->a=a;
```

```
};
Color::Color(int level) {
//       std::cout << "setting level " <<  level/255.0F << "\n";
   this->r = this->g = this->b = level/MAX_LEVEL;
   this->a = 1;
};

// the following two functions weights the different RGB channel differently instead of
doing a simple average. Using the common 0.3,0.59,0.11 weight relation.

int Color::getLevel() {
//  std::cout << r << " " << g << " " << b << " " << (int) round((r + g + b)*255.0/3.0) <<
"\n";
  return (int) round((0.3*r + 0.59*g + 0.11*b)*MAX_LEVEL);
}

int Color::getBoardLevel() {
//  std::cout << r << " " << g << " " << b << " " << (int) round((r + g + b)*255.0/3.0) <<
"\n";
  return (int) round((0.3*r + 0.59*g + 0.11*b)*MAX_REAL_LEVEL);
}
```

./daemon/src/xmlReader.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#include <stdio.h>
#include <stdbool.h>
#include <libxml/xmlreader.h>
#include "xmlReader.h"

#ifndef LIBXML_READER_ENABLED

#error LIBXML_READER_ENABLED not defined

#else

/**
 * streamFile:
 * @filename: the file name to parse
 *
 * Parse, validate and print information about an XML file.
 */
bool xmlReader::streamFile(const char *filename, Daemon *daemon) {
    xmlTextReaderPtr reader;
    int ret;


    /*
     * Pass some special parsing options to activate DTD attribute defaulting,
     * entities substitution and DTD validation
     */
    reader = xmlReaderForFile(filename, NULL,
```

```
                XML_PARSE_DTDATTR |  /* default DTD attributes */
                XML_PARSE_NOENT); /* |     substitute entities */
/*                XML_PARSE_DTDVALID);  validate with the DTD */
    if (reader != NULL) {
        ret = xmlTextReaderRead(reader);
        while (ret == 1) {
            daemon->processNode(reader);
            ret = xmlTextReaderRead(reader);
        }
        /*
         * Once the document has been fully parsed check the validation results
         */
/*        if (xmlTextReaderIsValid(reader) != 1) {
            fprintf(stderr, "Document %s does not validate\n", filename);
        } */
        xmlFreeTextReader(reader);
        if (ret != 0) {
            fprintf(stderr, "%s : failed to parse\n", filename);
            return false;
        }
    } else {
        fprintf(stderr, "Unable to open %s\n", filename);
        return false;
    }
    return true;
}

#endif
```

./daemon/src/Color.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef __ODCCOLOR_H
#define __ODCCOLOR_H
class Color {
  public:
    float r,g,b,a;

    Color(float r,float g,float b,float a);
    Color(int level);
    int getLevel();
    int getBoardLevel();
};
#endif
```

./daemon/src/modules/auth.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */
```

./daemon/src/modules/auth_htpasswd.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */
```

./daemon/src/Client.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
```

```
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2007 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef __ODC_CLIENT_H
#define __ODC_CLIENT_H

#include <string>
#include "pack.h"
#include "odc.h"


class Client {
public:
  bool authenticated;
  int authType;
  bool autoUpdate;
  std::string remoteAddress;
int fd;
  bool greet();
  bool sendCommand(int cmd);
         bool handleInput(unsigned char *input, int inputSize);
    bool authenticate(unsigned char *input, int inputSize);
};

#endif
```

./daemon/src/pack.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef ODC_PACK_H
#define ODC_PACK_H

#ifdef __cplusplus
extern "C" {
#endif

// macros for packing floats and doubles:
```

```
#define pack754_32(f) (pack754((f), 32, 8))
#define pack754_64(f) (pack754((f), 64, 11))
#define unpack754_32(i) (unpack754((i), 32, 8))
#define unpack754_64(i) (unpack754((i), 64, 11))

/*
** pack754() -- pack a floating point number into IEEE-754 format
*/
long long pack754(long double f, unsigned bits, unsigned expbits);

/*
** unpack754() -- unpack a floating point number from IEEE-754 format
*/
long double unpack754(long long i, unsigned bits, unsigned expbits);

/*
** packi16() -- store a 16-bit int into a char buffer (like htons())
*/
void packi16(unsigned char *buf, unsigned int i);

/*
** packi32() -- store a 32-bit int into a char buffer (like htonl())
*/
void packi32(unsigned char *buf, unsigned long i);

/*
** unpacki16() -- unpack a 16-bit int from a char buffer (like ntohs())
*/
unsigned int unpacki16(unsigned char *buf);

/*
** unpacki32() -- unpack a 32-bit int from a char buffer (like ntohl())
*/
unsigned long unpacki32(unsigned char *buf);

/*
** pack() -- store data dictated by the format string in the buffer
**
**  h - 16-bit              l - 32-bit
**  c - 8-bit char          f - float, 32-bit
**  s - string (16-bit length is automatically prepended)
*/
size_t pack(unsigned char *buf, char *format, ...);

/*
** unpack() -- unpack data dictated by the format string into the buffer
*/
void unpack(unsigned char *buf, char *format, ...);

/*
int main(void)
{
        unsigned char buf[1024];
        char magic;
        short monkeycount;
        long altitude;
        float absurdityfactor;
        char *s = "Great unmitigated Zot!  You've found the Runestaff!";
        char s2[96];
        size_t packetsize, ps2;

        packetsize = pack(buf, "chhlsf", 'B', 0, 37, -5, s, -3490.6677);
        packi16(buf+1, packetsize); // store packet size in packet for kicks

        printf("packet is %d bytes\n", packetsize);

        unpack(buf, "chhl96sf", &magic, &ps2, &monkeycount, &altitude, s2,
                &absurdityfactor);

        printf("'%c' %d %d %ld \"%s\" %f\n", magic, ps2, monkeycount, altitude,
                s2, absurdityfactor);

        return 0;
}

*/

#ifdef __cplusplus
}
#endif

#endif
```

./daemon/src/xmlReader.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef __ODCXMLREADER_H
#define __ODCXMLREADER_H

#include <libxml/xmlreader.h>
#include "Daemon.h"

class xmlReader {
  public:
    bool streamFile(const char *filename, Daemon *daemon);
};

#endif
```

./daemon/src/main.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2007 <mendoza@pvv.ntnu.no>
 *
 */

#include <iostream>

#include "xmlReader.h"
#include "Daemon.h"
#include "Client.h"
```

```
#include "pack.h"
#include "odc.h"

#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <sys/wait.h>
#include <signal.h>
#include <sys/un.h>

#include <map>
#include <string>

#define CONFIG_FILE "/etc/odc.xml"
#define SOCKET_FILE "/tmp/odc.sock"
#define SERVER_PORT 5001
#define BUFSIZE 2048

#define BACKLOG 10              // how many pending connections queue will
                                // hold


void sigchld_handler(int s)
{
  while (waitpid(-1, NULL, WNOHANG) > 0);
}

Daemon odcd;

int main(int argc, char **argv)
{
  bool parserRet;

  /*
   * this initialize the library and check potential ABI mismatches
   * between the version it was compiled for and the actual shared
   * library used.
   */
  LIBXML_TEST_VERSION xmlReader myXmlReader;

  parserRet = myXmlReader.streamFile(CONFIG_FILE, &odcd);

  /*
   * Cleanup function for the XML library.
   */
  xmlCleanupParser();

  /*
   * set up listening server
   */

  fd_set master;                // master file descriptor list
  fd_set read_fds;              // temp file descriptor list for select()
  struct sockaddr_in myaddr;    // server address
  struct sockaddr_in remoteaddr;        // client address


  int fdmax;                    // maximum file descriptor number
  int listener;                 // listening socket descriptor
  int newfd;                    // newly accept()ed socket descriptor
  char *buf;                    // buffer for client data
  int nbytes;
  int yes = 1;                  // for setsockopt() SO_REUSEADDR, below
  socklen_t addrlen;
  int i;

  FD_ZERO(&master);             // clear the master and temp sets
  FD_ZERO(&read_fds);

  buf = (char *) malloc(sizeof(char) * BUFSIZE);

  // get the listener
  if ((listener = socket(PF_INET, SOCK_STREAM, 0)) == -1) {
    perror("socket");
    exit(1);
  }
  // lose the pesky "address already in use" error message
  if (setsockopt(listener, SOL_SOCKET, SO_REUSEADDR, &yes, sizeof(int)) == -1) {

    perror("setsockopt");
    exit(1);
```

```
  }
  // bind
  myaddr.sin_family = AF_INET;
  myaddr.sin_addr.s_addr = INADDR_ANY;
  myaddr.sin_port = htons(SERVER_PORT);
  memset(myaddr.sin_zero, '\0', sizeof myaddr.sin_zero);
  if (bind(listener, (struct sockaddr *) &myaddr, sizeof(myaddr)) == -1) {
    perror("bind");
    exit(1);
  }
  // listen
  if (listen(listener, 10) == -1) {
    perror("listen");
    exit(1);
  }
  // add the listener to the master set
  FD_SET(listener, &master);

  // keep track of the biggest file descriptor
  fdmax = listener;             // so far, it's this one

  // main loop
  for (;;) {
    read_fds = master;          // copy it

    if (select(fdmax + 1, &read_fds, NULL, NULL, NULL) == -1) {
      perror("select");
      exit(1);
    }
    // run through the existing connections looking for data to read
    for (i = 0; i <= fdmax; i++) {
      if (FD_ISSET(i, &read_fds)) {
        if (i == listener) {
          // handle new connections
          addrlen = sizeof(remoteaddr);
          if ((newfd = accept(listener, (struct sockaddr *)
                              &remoteaddr, &addrlen)) == -1) {
            perror("accept");
          } else {
            FD_SET(newfd, &master);     // add to master set
            if (newfd > fdmax) {        // keep track of the
              // maximum
              fdmax = newfd;
            }
            ODC_LOG("selectserver: new connection from %s on " "socket %d\n",
inet_ntoa(remoteaddr.sin_addr), newfd);

            Client *tmpClient = new Client();

            tmpClient->remoteAddress = std::string(inet_ntoa(remoteaddr.sin_addr));
            tmpClient->authenticated = false;
            tmpClient->fd = newfd;
            ODC_LOG("bar\n");
            odcd.addClient(tmpClient);
            ODC_LOG("foo\n");
            if (tmpClient->greet() == false) {
              perror("send");
            }
          }
          std::cout << "mymap contains:\n";
          for (odcd.it = odcd.clients.begin(); odcd.it != odcd.clients.end(); odcd.it++) {
            std::cout << (*odcd.it).first << " => " << (*odcd.it).second << std::endl;
          }
        } else {
          // handle data from a client
          if ((nbytes = recv(i, buf, BUFSIZE, 0)) <= 0) {
            // got error or connection closed by client
            if (nbytes == 0) {
              // connection closed
              ODC_LOG("selectserver: socket %d hung up\n", i);
            } else {
              perror("recv");
            }
            odcd.removeClient(i);
            close(i);           // bye!
            FD_CLR(i, &master); // remove from master set
          } else {
            // we got some data from a client
            if (!odcd.clients[i]->handleInput((unsigned char *) buf, nbytes)) {
              odcd.removeClient(i);
              close(i);         // bye!
              FD_CLR(i, &master);       // remove from master set
```

```
            }
          }
        }
      }
    }
  }
  for (odcd.it = odcd.clients.begin(); odcd.it != odcd.clients.end(); odcd.it++) {
    free(odcd.it->second);
  }
  free((void *) buf);
  return 0;
}
```

./daemon/src/Board.cc:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#include <math.h>
#include <iostream>
#include "Board.h"
#include "Comm.h"

Board::Board() {
  init();
}

Board::Board(char *direction, char *ip) {
  this->direction = direction;
  this->ip = ip;
  init();
}

void Board::init() {
  for(int i=0;i<MAX_LIGHTS;i++) {
    lightValues.push_back(Color(0));
  }
  lightsX = (int) round(sqrt(MAX_LIGHTS));
  lightsY = lightsX;
}

bool Board::setPixelLevel(int x, int y, int level) {
  int lindex = calcLightIndex(x,y);
  lightValues[lindex] = Color(level);
  return true;
}

bool Board::clear() {
  for(unsigned int i=0;i<lightValues.size();i++) {
    lightValues[i] = 0;
  }
  return update();
}
```

```
bool Board::update() {
  char packet[MAX_LIGHTS+2];
  for(unsigned int j=0;j<lightValues.size();j++) {
    packet[0] = OP_SETLIGHT;
    packet[j+1] = (char) lightValues[j].getBoardLevel();
    packet[MAX_LIGHTS+1] = 0;
  }

  if (!Comm::updateBoard(ip,packet)) {
    return false;
  }

  return true;
}

int Board::getPixelLevel(int x, int y) {
  int lindex = calcLightIndex(x,y);
  return lightValues[lindex].getLevel();
}

Color Board::getPixelRGBA(int x, int y) {
  int lindex = calcLightIndex(x,y);
  return lightValues[lindex];
}

bool Board::setPixelRGBA(int x,int y,float R,float G,float B,float A) {
  int lindex = calcLightIndex(x,y);
  lightValues[lindex] = Color(R,G,B,A);
  return true;
}

int Board::calcLightIndex(int x, int y) {
  int lx = x % LIGHTS_X;
  int ly = y % LIGHTS_Y;

  int lindex;

  if (!strcmp(direction,"SW")) {
        printf("SW\n");
    lindex = ((LIGHTS_X-1-ly)*LIGHTS_X)+(lx);
  } else if (!strcmp(direction,"SE")) {
        printf("%s %s SE\n",ip,direction);
        lindex = ((LIGHTS_Y-ly)+((MAX_LIGHTS-1)-(LIGHTS_X*(lx+1))));
  } else if (!strcmp(direction,"NE")) {
        printf("NE\n");
    lindex = ((LIGHTS_X-1)-lx)+(ly*LIGHTS_X);
  } else {
        printf("NW\n");
    lindex = ((ly+(lx*LIGHTS_X)));
  }

  return lindex;
}
```

./daemon/src/Board.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
 *
 * Copyright Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef __ODCBOARD_H
#define __ODCBOARD_H

#include <vector>
#include <iostream>

#include "Color.h"

#define MAX_LIGHTS 25
#define MAX_LEVEL 255
#define MAX_REAL_LEVEL 99
#define LIGHTS_X 5
#define LIGHTS_Y 5
enum {OP_SETLIGHT = 1,OP_SETIP = 2};


class Board {
  private:
    int calcLightIndex(int boardx, int boardy);
  public:
    Board();

    Board(char *direction,char *ip);
    bool setPixelLevel(int x, int y,int level);
    int getPixelLevel(int x,int y);
    void init();
    Color getPixelRGBA(int x, int y);
    bool setPixelRGBA(int x,int y,float R,float G,float B,float A);
    char *direction;
    char *ip;
    int lightsX;
    int lightsY;
    std::vector<Color> lightValues;
    bool clear();
    bool update();
};

#endif
```

./daemon/src/Makefile:

```
#
# $Id: Makefile 61 2007-06-22 00:48:53Z mendoza $
#

DEBUG=-ggdb
INCLUDES=`xml2-config --cflags` -I../include

SVNDEF := -D'SVN_REV="$(shell svnversion -n .)"'

CXXFLAGS=-Wall -Wuninitialized -O2 $(INCLUDES) $(DEBUG) $(SVNDEF)
CFLAGS=-Wall -Wuninitialized -O2 $(INCLUDES) $(DEBUG) $(SVNDEF)
LIBS=`xml2-config --libs`

OBJS = xmlReader.o Daemon.o Comm.o Canvas.o Board.o main.o Color.o pack.o Client.o

DAEMON=odcd

srcdir = src
top_srcdir = src

prefix = /usr
exec_prefix = ${prefix}

bindir = ${exec_prefix}/bin
sbindir = ${exec_prefix}/sbin
libexecdir = ${exec_prefix}/libexec
datadir = ${prefix}/share
sysconfdir = ${prefix}/etc
sharedstatedir = ${prefix}/com
localstatedir = ${prefix}/var
libdir = ${exec_prefix}/lib
infodir = ${prefix}/info
```

```
mandir = ${prefix}/man
includedir = ${prefix}/include
oldincludedir = /usr/include
pkgdatadir = $(datadir)/odcd
pkglibdir = $(libdir)/odcd
pkgincludedir = $(includedir)/odcd
top_builddir = .

OBJS:=${OBJS:%=${srcdir}/%}


$(DAEMON): $(OBJS)
        $(CXX) $(CXXFLAGS) $(CPPFLAGS) $(LDFLAGS) -o $@ $(OBJS) $(LIBS)

clean:
        rm $(OBJS) $(DAEMON)

install:
        cp $(DAEMON) $(sbindir)
        cp etc/init.d/odcd /etc/init.d/
        cp etc/odc.x* /etc/
```

./daemon/LICENSE:

```
Unless otherwise stated, the files in this directory and subdirectories are licensed with
a two-clause BSD License,
and copyright is held by Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>

This source file is licensed under a two-clause BSD license

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

./daemon/TODO:

```
* proper installation
* run daemon in background
* At least one proof-of-concept wrapper (php, java, python, perl)
```

./daemon/INSTALL:

```
To install the Open Digital Canvas daemon, simply type "make" in this directory, and "sudo
make install" to copy the
files to your system.

The daemon will be located at /usr/bin/odcd, the init script in /etc/init.d and the config
file and the schema
belonging to it in /etc named odc.xml and odc.xsd correspondingly.

Start and stop the daemon with "/etc/init.d/odcd start" and "/etc/init.d/odcd stop"
```

./daemon/etc/init.d/odcd:

```
#! /bin/sh
### BEGIN INIT INFO
# Provides:          odcd
# Required-Start:    $local_fs $remote_fs
# Required-Stop:     $local_fs $remote_fs
# Default-Start:     2 3 4 5
# Default-Stop:      S 0 1 6
# Short-Description: Example initscript
# Description:       This file should be used to construct scripts to be
#                    placed in /etc/init.d.
### END INIT INFO

# Author: Nicolas Mendoza <mendoza@pvv.ntnu.no>
#
# Please remove the "Author" lines above and replace them
# with your own name if you copy and modify this script.

# Do NOT "set -e"

# PATH should only include /usr/* if it runs after the mountnfs.sh script
PATH=/usr/sbin:/usr/bin:/sbin:/bin
DESC="Open Digital Canvas Daemon"
NAME=odcd
DAEMON=/usr/sbin/$NAME
DAEMON_ARGS=""
PIDFILE=/var/run/$NAME.pid
SCRIPTNAME=/etc/init.d/$NAME

# Exit if the package is not installed
[ -x "$DAEMON" ] || exit 0

# Read configuration variable file if it is present
[ -r /etc/default/$NAME ] && . /etc/default/$NAME

# Load the VERBOSE setting and other rcS variables
[ -f /etc/default/rcS ] && . /etc/default/rcS

# Define LSB log_* functions.
# Depend on lsb-base (>= 3.0-6) to ensure that this file is present.
. /lib/lsb/init-functions

#
# Function that starts the daemon/service
#
do_start()
{
        # Return
        #   0 if daemon has been started
        #   1 if daemon was already running
        #   2 if daemon could not be started
        start-stop-daemon --start --quiet --pidfile $PIDFILE --exec $DAEMON --test > /dev/
null \
                || return 1
        start-stop-daemon --start --quiet --pidfile $PIDFILE --exec $DAEMON -- \
                $DAEMON_ARGS \
                || return 2
        # Add code here, if necessary, that waits for the process to be ready
        # to handle requests from services started subsequently which depend
        # on this one.  As a last resort, sleep for some time.
}

#
# Function that stops the daemon/service
#
do_stop()
{
        # Return
        #   0 if daemon has been stopped
        #   1 if daemon was already stopped
        #   2 if daemon could not be stopped
        #   other if a failure occurred
        start-stop-daemon --stop --quiet --retry=TERM/30/KILL/5 --pidfile $PIDFILE --name
$NAME
        RETVAL="$?"
        [ "$RETVAL" = 2 ] && return 2
        # Wait for children to finish too if this is a daemon that forks
        # and if the daemon is only ever run from this initscript.
        # If the above conditions are not satisfied then add some other code
```

```
        # that waits for the process to drop all resources that could be
        # needed by services started subsequently.  A last resort is to
        # sleep for some time.
        start-stop-daemon --stop --quiet --oknodo --retry=0/30/KILL/5 --exec $DAEMON
        [ "$?" = 2 ] && return 2
        # Many daemons don't delete their pidfiles when they exit.
        rm -f $PIDFILE
        return "$RETVAL"
}

#
# Function that sends a SIGHUP to the daemon/service
#
do_reload() {
        #
        # If the daemon can reload its configuration without
        # restarting (for example, when it is sent a SIGHUP),
        # then implement that here.
        #
        start-stop-daemon --stop --signal 1 --quiet --pidfile $PIDFILE --name $NAME
        return 0
}

case "$1" in
  start)
        [ "$VERBOSE" != no ] && log_daemon_msg "Starting $DESC" "$NAME"
        do_start
        case "$?" in
                0|1) [ "$VERBOSE" != no ] && log_end_msg 0 ;;
                2) [ "$VERBOSE" != no ] && log_end_msg 1 ;;
        esac
        ;;
  stop)
        [ "$VERBOSE" != no ] && log_daemon_msg "Stopping $DESC" "$NAME"
        do_stop
        case "$?" in
                0|1) [ "$VERBOSE" != no ] && log_end_msg 0 ;;
                2) [ "$VERBOSE" != no ] && log_end_msg 1 ;;
        esac
        ;;
  #reload|force-reload)
        #
        # If do_reload() is not implemented then leave this commented out
        # and leave 'force-reload' as an alias for 'restart'.
        #
        #log_daemon_msg "Reloading $DESC" "$NAME"
        #do_reload
        #log_end_msg $?
        #;;
  restart|force-reload)
        #
        # If the "reload" option is implemented then remove the
        # 'force-reload' alias
        #
        log_daemon_msg "Restarting $DESC" "$NAME"
        do_stop
        case "$?" in
          0|1)
                do_start
                case "$?" in
                        0) log_end_msg 0 ;;
                        1) log_end_msg 1 ;; # Old process is still running
                        *) log_end_msg 1 ;; # Failed to start
                esac
                ;;
          *)
                  # Failed to stop
                log_end_msg 1
                ;;
        esac
        ;;
  *)
        #echo "Usage: $SCRIPTNAME {start|stop|restart|reload|force-reload}" >&2
        echo "Usage: $SCRIPTNAME {start|stop|restart|force-reload}" >&2
        exit 3
        ;;
esac

:
```

./daemon/etc/odc.xsd:

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://odc.opentheweb.org"
xmlns="http://odc.opentheweb.org"
elementFormDefault="qualified">

<xs:annotation>
  <xs:appinfo>ODC config file schema</xs:appinfo>
  <xs:documentation xml:lang="en">
  This Schema defines an ODC config file.
  </xs:documentation>
</xs:annotation>

<xs:element name="odc">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="canvas" minOccurs="1" maxOccurs="unbounded" >
        <xs:complexType>
          <xs:sequence>
            <xs:element name="board" type="boardtype" minOccurs="1" maxOccurs="unbounded"
/>
          </xs:sequence>
          <xs:attribute name="width" type="inttype" use="required" />
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:complexType name="boardtype">
    <xs:attribute name="ip" type="xs:string" use="required"/>
    <xs:attribute name="direction" type="directiontype" use="required"/>
</xs:complexType>

<xs:simpleType name="inttype">
  <xs:restriction base="xs:positiveInteger"/>
</xs:simpleType>

<xs:simpleType name="iptype">
  <xs:restriction base="xs:string">
    <xs:pattern
value="((0|1[0-9]{1,2}|2([0-4][0-9]?|5[0-5]?|[6-9]))\.){3}(0|1[0-9]{1,2}|2([0-4][0-9]?|5[0-5]?|[6-9])
/>
   </xs:restriction>
</xs:simpleType>

<xs:simpleType name="directiontype">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NW"/>
    <xs:enumeration value="SW"/>
    <xs:enumeration value="NE"/>
    <xs:enumeration value="SE"/>
  </xs:restriction>
</xs:simpleType>

</xs:schema>
```

./daemon/etc/odc.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<odc xmlns="http://odc.opentheweb.org">
<!-- you can have many canvas elements, containing many boards. each canvas needs to know
its width to properly lay
out graphics (and calculate the height, based on amount of board elements inside a canvas)
-->
<canvas width="3">
<board direction="SW" ip="10.0.10.1" />
<board direction="NW" ip="10.0.10.2" />
<board direction="NW" ip="10.0.10.3" />
<board direction="NE" ip="10.0.10.4" />
<board direction="SE" ip="10.0.10.5" />
<board direction="NW" ip="10.0.10.6" />
</canvas>
</odc>
```

./daemon/Makefile:

```
include src/Makefile
```

./daemon/README:

```
Open Digital Canvas -- daemon

Please see http://odc.openthweb.org for more information.

The format of the odc.xml config file is pretty straight forward, and the accompanying
odc.xml and its xml schema
should be self-explanatory.

The default config file defines one canvas consisting of 6 boards. The width is 3 baords
making it two rows.

Each board can have a direction which means which quadrant of a card the following chip
lays in, this is to simplify
mounting of a canvas, making the system agnostic of which direction a board is rotated
when mounted.

    ..
    ||
   /  \
  :|  |
   ....
    ....

  (fig.1 layout of trace of chip)

The quadrants are named after cardinal directions, where north means upwards, the
following valid values are
NW,NE,SW,SE.

Each board will have an ip address assigned. The order they appear in in the config file
states their order in the
system. this can also be re-arranged to match the actual physical configuration when
mounted.

The boards in the system at IDI, NTNU have ip addresses ranging from 10.0.10.1 to
10.0.10.110, of course one can use
any address one wants. The port of a card can be configured in the daemon source code for
now.

Check INSTALL for instrutions to install this software and LICENSE to see its license.
```

./include/odc.h:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
 *
 * Copyright Nicolas Mendoza 2007 <mendoza@pvv.ntnu.no>
 *
 */

#ifndef ODC_COMMANDS_H
#define ODC_COMMANDS_H

#define PROTOCOL_VERSION    1

#define SRV_AUTH_REQUIRED   1
#define SRV_AUTH_ACCEPTED   2
#define SRV_AUTH_ERR        3
#define SRV_UNKNOWN_CMD     4
#define SRV_OK              5
#define SRV_LEVEL           6
#define SRV_RGBA            7
#define SRV_LEVEL_ARRAY     8
#define SRV_RGBA_ARRAY      9
#define SRV_CANVAS_INFO     10

#define CLT_AUTH            1

#define CLT_QUIT            2
#define CLT_SETPIXELLEVEL   3
#define CLT_SETPIXELRGBA    4
#define CLT_CLEAR           5
#define CLT_UPDATE          6
#define CLT_AUTOUPDATEMODE  7
#define CLT_GETPIXELLEVEL   8
#define CLT_GETPIXELRGBA    9
#define CLT_BLITARRAYLEVEL  10
#define CLT_BLITARRAYRGBA   11
#define CLT_SELECTCANVAS    12

#define AUTH_TYPE_ADMIN 1
#define AUTH_TYPE_USER 2

#define WELCOME_MSG "ODC " SVN_REV " Copyright Nicolas Mendoza <mendoza@pvv.ntnu.no>\n"

#define ODC_DEBUG 0

#ifdef ODC_DEBUG
#define STRINGIFY(x) #x
#define TOSTRING(x) STRINGIFY(x)
#define ODC_LOG(...) printf(__FILE__ ":" TOSTRING(__LINE__) " " __VA_ARGS__)
#else
#define ODC_LOG(...)
#endif

#include <netinet/in.h>
#include <sys/socket.h>

struct odc_color {
  float r;
  float g;
  float b;
  float a;
};

struct odc_connection {
  int sockfd;
  struct hostent *he;
  struct sockaddr_in their_addr;
  };

/* connect to an odcd server, canvas is the index of the canvas, only supports one canvas
atm, so defaults to 0.
   returns an id to current connection */
struct odc_connection *odc_connect(struct odc_connection *odc_c,char *host, int port, char
*username, char *password,int canvas);

/* disconnect from given odc server */
void odc_disconnect(struct odc_connection *odc_c);

/* return light level 0-255 */
int odc_get_pixel_level(struct odc_connection *odc_c,int x,int y);

/* set a light to level 0-255 */
int odc_set_pixel_level(struct odc_connection *odc_c,int x,int y,int level);

/* get a color struct with 0.0-1.0 values for r,g,b,a */
```

```
struct odc_color odc_get_pixel_rgba(struct odc_connection *odc_c,int x,int y);

/* set a color struct with 0.0-1.0 values for r,g,b,a */
int odc_set_pixel_rgba(struct odc_connection *odc_c,int x,int y,float R,float G,float
B,float A);

/* blit an array with defined stride and length of level (0-255) values */
int odc_blit_array(struct odc_connection *odc_c,int x,int y,int* array,int length, int
stride);

/* blit an array with defined stride and length of odc_color values */
int odc_blit_array_rgba(struct odc_connection *odc_c,int x,int y,struct odc_color*
array,int length, int stride);

/* check if a position is valid (within current canvas) */
int odc_is_valid_pos(struct odc_connection *odc_c,int x, int y);

/* clear board */
int odc_clear(struct odc_connection *odc_c);

/* update board, ie. execute commands in queue to the physical boards
   auto-update mode currently not implemented */
int odc_update(struct odc_connection *odc_c);

#endif
```

./lib/src/Makefile:

```
#
# $Id: Makefile 59 2007-06-22 00:43:40Z mendoza $
#

DEBUG=-ggdb
INCLUDEDIR=../include
INCLUDES=-I$(INCLUDEDIR)

SVNDEF := -D'SVN_REV="$(shell svnversion -n .)"'

CFLAGS=-Wall -Wuninitialized -O2 $(INCLUDES) $(DEBUG) $(SVNDEF)
LIBS=
LDFLAGS=-L.

OBJS = odc_main.o ../../daemon/src/pack.o
SOLIB = libodc.so
LIB = libodc.a
TESTOBJS = test.o

DAEMON=odcd

srcdir = src
top_srcdir = src

prefix = /usr
exec_prefix = ${prefix}

bindir = ${exec_prefix}/bin
sbindir = ${exec_prefix}/sbin
libexecdir = ${exec_prefix}/libexec
datadir = ${prefix}/share
sysconfdir = ${prefix}/etc
sharedstatedir = ${prefix}/com
localstatedir = ${prefix}/var
libdir = ${exec_prefix}/lib
infodir = ${prefix}/info
mandir = ${prefix}/man
includedir = ${prefix}/include
oldincludedir = /usr/include
pkgdatadir = $(datadir)/odcd
pkglibdir = $(libdir)/odcd
pkgincludedir = $(includedir)/odcd
top_builddir = .

OBJS:=${OBJS:%=${srcdir}/%}
TESTOBJS:=${TESTOBJS:%=${srcdir}/%}

all: $(SOLIB) $(LIB)

$(LIB): $(OBJS)
```

```
	$(AR) rcs $@ $(OBJS)

$(SOLIB): $(OBJS)
	$(CC) -shared $(CFLAGS) $(CPPFLAGS) $(LDFLAGS) -o $@ $(OBJS) $(LIBS)

clean:
	rm $(OBJS) $(SOLIB) $(LIB) $(TESTOBJS) test

test:	all $(TESTOBJS)
	$(CC) $(CFLAGS) $(CPPFLAGS) $(LDFLAGS) -lodc -o $@ $(TESTOBJS) $(LIBS)

install: all
	cp $(SOLIB) $(libdir)
	cp $(LIB) $(libdir)
	cp $(INCLUDEDIR)/*.h $(includedir)
```

./lib/src/test.c:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2007 <mendoza@pvv.ntnu.no>
 *
 */

#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include "odc.h"

#define ODC_HOST "localhost"
#define ODC_PORT 5001


int main(void)
{

  struct odc_connection *odc_c = (struct odc_connection *) malloc(sizeof(struct
odc_connection));

  printf("%p\n",odc_c);

  if ((odc_c = odc_connect(odc_c, ODC_HOST, ODC_PORT, (char *) 0, (char *) 0, 0)) == 0) {
    printf("Unable to connect to %s:%d\n", ODC_HOST, ODC_PORT);
    return EXIT_FAILURE;
  }

  printf("%p\n",odc_c);

  printf("Setting pixel levels\n");

  odc_set_pixel_level(odc_c, 0, 0, 255);
  odc_set_pixel_level(odc_c, 1, 1, 128);
  odc_set_pixel_level(odc_c, 2, 2, 64);
  odc_set_pixel_level(odc_c, 3, 3, 32);

  struct odc_color mycolor;
```

```
mycolor.r = 1.0F;
mycolor.g = 1.0F;
mycolor.b = 1.0F;
mycolor.a = 1.0F;

printf("Setting pixel RGBA\n");

odc_set_pixel_rgba(odc_c, 1, 0, mycolor.r, mycolor.g, mycolor.b, mycolor.a);

mycolor.r /= 2;
mycolor.g /= 2;
mycolor.b /= 2;
mycolor.a /= 2;

odc_set_pixel_rgba(odc_c, 2, 1, mycolor.r, mycolor.g, mycolor.b, mycolor.a);

mycolor.r /= 2;
mycolor.g /= 2;
mycolor.b /= 2;
mycolor.a /= 2;

odc_set_pixel_rgba(odc_c, 3, 2, mycolor.r, mycolor.g, mycolor.b, mycolor.a);

mycolor.r /= 2;
mycolor.g /= 2;
mycolor.b /= 2;
mycolor.a /= 2;

odc_set_pixel_rgba(odc_c, 4, 3, mycolor.r, mycolor.g, mycolor.b, mycolor.a);

printf("Updating canvas\n");

odc_update(odc_c);

sleep(2);

printf("Clear canvas\n");

odc_clear(odc_c);

printf("Setting pixel levels\n");

odc_set_pixel_level(odc_c, 4, 4, 255);
odc_set_pixel_level(odc_c, 5, 5, 128);
odc_set_pixel_level(odc_c, 6, 6, 64);
odc_set_pixel_level(odc_c, 7, 7, 32);

mycolor.r = 1.0F;
mycolor.g = 1.0F;
mycolor.b = 1.0F;
mycolor.a = 1.0F;

printf("Setting pixel RGBA\n");

odc_set_pixel_rgba(odc_c, 5, 4, mycolor.r, mycolor.g, mycolor.b, mycolor.a);

mycolor.r /= 2;
mycolor.g /= 2;
mycolor.b /= 2;
mycolor.a /= 2;

odc_set_pixel_rgba(odc_c, 6, 5, mycolor.r, mycolor.g, mycolor.b, mycolor.a);

mycolor.r /= 2;
mycolor.g /= 2;
mycolor.b /= 2;
mycolor.a /= 2;

odc_set_pixel_rgba(odc_c, 7, 6, mycolor.r, mycolor.g, mycolor.b, mycolor.a);

mycolor.r /= 2;
mycolor.g /= 2;
mycolor.b /= 2;
mycolor.a /= 2;

odc_set_pixel_rgba(odc_c, 8, 7, mycolor.r, mycolor.g, mycolor.b, mycolor.a);

printf("Updating canvas\n");

odc_update(odc_c);

printf("Disconnecting\n");
```

```
  odc_disconnect(odc_c);

  printf("%p\n",odc_c);
  free((void *)odc_c);
  return EXIT_SUCCESS;
}
```

./lib/src/odc_main.c:

```
/*
 * This source file is licensed under a two-clause BSD license
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 *  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 *
 * Copyright Nicolas Mendoza 2007 <mendoza@pvv.ntnu.no>
 *
 */

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <errno.h>
#include <string.h>
#include <netdb.h>
#include <sys/types.h>
#include <netinet/in.h>
#include <sys/socket.h>

#include "odc.h"
#include "../../daemon/src/pack.h"

#define ODC_PORT 5001 // the port client will be connecting to
#define MAXDATASIZE 100000 // max number of bytes we can get at once
#define LIBODC_PROTOCOL 1

int numbytes;
char buf[MAXDATASIZE];

/* connect to an odcd server, canvas is the index of the canvas, only supports one canvas
atm, so defaults to 0.
   returns an id to current connection */
struct odc_connection * odc_connect(struct odc_connection *odc_c, char *host, int port,
char *username, char *password,int canvas) {

    if ((odc_c->he=gethostbyname(host)) == NULL) {  // get the host info
        herror("gethostbyname");
        return 0;
    }

    if ((odc_c->sockfd = socket(PF_INET, SOCK_STREAM, 0)) == -1) {
        perror("socket");
        return 0;
    }

    odc_c->their_addr.sin_family = AF_INET;    // host byte order
    odc_c->their_addr.sin_port = htons((port?port:ODC_PORT));  // short, network byte
order
    odc_c->their_addr.sin_addr = *((struct in_addr *)odc_c->he->h_addr);
    memset(odc_c->their_addr.sin_zero, '\0', sizeof odc_c->their_addr.sin_zero);
```

```
    if (connect(odc_c->sockfd, (struct sockaddr *)&odc_c->their_addr,
                                          sizeof(struct sockaddr)) == -1) {
        perror("connect");
        return 0;
    }

    if ((numbytes=recv(odc_c->sockfd, buf, MAXDATASIZE-1, 0)) == -1) {
        perror("recv");
        odc_disconnect(odc_c);
        return 0;
    }

    short protocol;
    short cmd;
    size_t bufsize;
    char greet[1024],outbuf[1024];

    unpack((unsigned char*)buf,"hhs",&protocol, &cmd, &greet);

    ODC_LOG("Received: %hd %hd %s\n",protocol, cmd, greet);

    if (protocol != LIBODC_PROTOCOL) {
      ODC_LOG("Incompatible protocol %d\n",protocol);
      odc_disconnect(odc_c);
      return NULL;
    }

    bufsize = pack((unsigned char*)outbuf,"h",CLT_AUTH);
    // FIXME: send username and hashed password
    if (send(odc_c->sockfd,outbuf,bufsize,0) == -1) {
      perror("auth");
    }

    if ((numbytes=recv(odc_c->sockfd, buf, MAXDATASIZE-1, 0)) == -1) {
        perror("recv");
        odc_disconnect(odc_c);
        return NULL;
    }

    unpack((unsigned char*)buf,"h", &cmd);
    if (cmd != SRV_AUTH_ACCEPTED) {
      ODC_LOG("cmd: %d\n",cmd);
      perror("authentication failed");
      odc_disconnect(odc_c);
      return NULL;
    }
    ODC_LOG("Connection successful\n");
    return odc_c;
}

/* disconnect from server */
void odc_disconnect(struct odc_connection *odc_c) {
  unsigned char buf[100];
  size_t packetsize;

  packetsize = pack(buf,"h",CLT_QUIT);
  if (send(odc_c->sockfd,buf,packetsize,0) == -1) {
    perror("send quit");
  }
  ODC_LOG("Disconnecting\n");
  close(odc_c->sockfd);
}

/* return light level 0-255 */
int odc_get_pixel_level(struct odc_connection *odc_c, int x,int y) { return 0;
  int level;
  unsigned char buf[MAXDATASIZE];
  size_t packetsize,numbytes;
  short result_cmd;

  ODC_LOG("Trying to get pixel (%d,%d)\n",x,y);
  packetsize = pack(buf,"hhhffff",CLT_GETPIXELLEVEL, x, y);
  if (send(odc_c->sockfd,buf,packetsize,0) == -1) {
    perror("send get_pixel_rgba");
  }

  if ((numbytes=recv(odc_c->sockfd, buf, MAXDATASIZE-1, 0)) == -1) {
    perror("recv");
    return 0;
  }
```

```
  unpack(buf,"hfffff",&result_cmd,&level);
  if (result_cmd != SRV_RGBA) {
    return 0;
  }
  return level;
//FIXME: be able to tell if it failed properly (now returns 0)
}

/* set a light to level 0-255 */
int odc_set_pixel_level(struct odc_connection *odc_c, int x,int y,int level) {
  unsigned char buf[MAXDATASIZE];
  size_t packetsize,numbytes;
  short result_cmd;

  ODC_LOG("Trying to set pixel (%d,%d) to level %d\n",x,y,level);
  packetsize = pack(buf,"hhhh",CLT_SETPIXELLEVEL, x, y, level);
  if (send(odc_c->sockfd,buf,packetsize,0) == -1) {
    perror("send set_pixel_level");
  }

  if ((numbytes=recv(odc_c->sockfd, buf, MAXDATASIZE-1, 0)) == -1) {
    perror("recv");
    return 0;
  }

  unpack(buf,"h",&result_cmd);
  if (result_cmd != SRV_OK) {
    return 0;
  }
  return 1;
}

/* get a color struct with 0.0-1.0 values for r,g,b,a */
struct odc_color odc_get_pixel_rgba(struct odc_connection *odc_c, int x,int y) {
  struct odc_color color;
  unsigned char buf[MAXDATASIZE];
  size_t packetsize,numbytes;
  short result_cmd;

  ODC_LOG("Trying to get pixel (%d,%d)\n",x,y);
  packetsize = pack(buf,"hhhffff",CLT_GETPIXELRGBA, x, y);
  if (send(odc_c->sockfd,buf,packetsize,0) == -1) {
    perror("send get_pixel_rgba");
  }

  if ((numbytes=recv(odc_c->sockfd, buf, MAXDATASIZE-1, 0)) == -1) {
    perror("recv");
// FIXME: proper error return
  }

  unpack(buf,"hfffff",&result_cmd,&(color.r),&(color.g),&(color.b),&(color.a));
  if (result_cmd != SRV_RGBA) {
// FIXME: proper error return
  }
  return color;
}

/* set a color struct with 0.0-1.0 values for r,g,b,a */
int odc_set_pixel_rgba(struct odc_connection *odc_c, int x,int y,float R,float G,float
B,float A) {
  unsigned char buf[MAXDATASIZE];
  size_t packetsize,numbytes;
  short result_cmd;

  ODC_LOG("Trying to set pixel (%d,%d) to rgba %f %f %f %f\n",x,y,R, G, B, A);
  packetsize = pack(buf,"hhhffff",CLT_SETPIXELRGBA, x, y, R, G, B, A);
  if (send(odc_c->sockfd,buf,packetsize,0) == -1) {
    perror("send set_pixel_rgba");
  }

  if ((numbytes=recv(odc_c->sockfd, buf, MAXDATASIZE-1, 0)) == -1) {
    perror("recv");
    return 0;
  }

  unpack(buf,"h",&result_cmd);
  if (result_cmd != SRV_OK) {
    return 0;
  }

  return 1;
}
```

```
/* blit an array with defined stride and length of level (0-255) values */
int odc_blit_array(struct odc_connection *odc_c, int x,int y,int* array,int length, int
stride) {
// FIXME: needs proper sendall functionality
  return 0;
}

/* blit an array with defined stride and length of odc_color values */
int odc_blit_array_rgba(struct odc_connection *odc_c, int x,int y,struct odc_color*
array,int length, int stride) {
// FIXME: needs proper sendall functionality
  return 0;
}

/* check if a position is valid (within current canvas) */
int odc_is_valid_pos(struct odc_connection *odc_c, int x, int y) {
  return 0;
}

/* clear board */
int odc_clear(struct odc_connection *odc_c) {
  unsigned char buf[MAXDATASIZE];
  size_t packetsize,numbytes;
  short result_cmd;

  ODC_LOG("Trying clear canvas\n");
  packetsize = pack(buf,"h",CLT_CLEAR);
  if (send(odc_c->sockfd,buf,packetsize,0) == -1) {
    perror("send clear");
  }

  if ((numbytes=recv(odc_c->sockfd, buf, MAXDATASIZE-1, 0)) == -1) {
    perror("recv");
    return 0;
  }

  unpack(buf,"h",&result_cmd);
  if (result_cmd != SRV_OK) {
    return 0;
  }
  return 1;
}

/* update board, ie. execute commands in queue to the physical boards
   auto-update mode currently not implemented */
int odc_update(struct odc_connection *odc_c) {
  unsigned char buf[MAXDATASIZE];
  size_t packetsize,numbytes;
  short result_cmd;

  ODC_LOG("Trying update canvas\n");
  packetsize = pack(buf,"h",CLT_UPDATE);
  if (send(odc_c->sockfd,buf,packetsize,0) == -1) {
    perror("send update");
  }

  if ((numbytes=recv(odc_c->sockfd, buf, MAXDATASIZE-1, 0)) == -1) {
    perror("recv");
    return 0;
  }

  unpack(buf,"h",&result_cmd);
  if (result_cmd != SRV_OK) {
    return 0;
  }
  return 1;
  return 0;
}
```

./lib/LICENSE:

```
Unless otherwise stated, the files in this directory and subdirectories are licensed with
a two-clause BSD License,
and copyright is held by Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>

This source file is licensed under a two-clause BSD license
```

```
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

./lib/INSTALL:

```
To install the Open Digital Canvas library, simply type "make" in this directory, and
"sudo make install" to
copy the files to your system. To compile a small test, use "make test", and run it by
typing ./test after running
make install for the libraries. Also make sure that you alter the ODC_HOST and ODC_PORT
defines in the test to test
with your own server, the default is localhost:5001

The libraries will be located in /usr/lib/libodc.a and /usr/lib/libodc.so, for static and
dynamic linking
respectively. The include file "odc.h" will be located in /usr/include.
```

./lib/README:

```
Open Digital Canvas -- library

Please see http://odc.openthweb.org for more information.

Check INSTALL for instrutions to install this software and LICENSE to see its license.
```

./lib/Makefile:

```
include src/Makefile
```

./INSTALL:

```
To install the various parts. Read the INSTALL file on the corresponding directories.
```

./README:

```
This is the full repository of Open Digital Canvas. It's a server-client framework to
access canvases set up by
specific motherboards from regular software with comprehensible functions.

The repository contains the following:
```

```
avr                Updated firmware code, donated by Norvald Ryeng, origins from Livingwall
                bugfixed and streamlined by Nicolas Mendoza
daemon                ODC Daemon. Runs as a daemon and controls cards
lib                libodc - library to connect to daemon and send primitive commands
include                 includes for projects using libodc
legacy                older versions of code and related projects, mostly for reference
php-extension        Example of how to use libodc, this is source for a php extension
test                Perl test suite to test cards and communicate directly
thesis                Thesis written based on the whole project by Nicolas Mendoza
```

./avr/README:

```
Open Digital Canvas -- firmware

This is the source for the firmware on the boards making up the original system made for
IDI, NTNU. The code
originates from the Livingwall project (http://livingwall.org) and has been donated by
Ã…smund GamlesÃ¦ter, thoush
transferred thru Norvald Ryeng. Permission to license the code as BSD two-clause license
was granted by Ã…smund
GamlesÃ¦ter in spring 2007. This version differs from the original one provided in that a
bug that caused lights to
have less intensity and lit the light on the previous row slightly is fixed, in addition
the Makefile and some header
files have been altered to easily change both MAC and IP from the command line when making
a firmware image.

The code is based on uIP (http://www.sics.se/~adam/uip/), uIP-AVR (http://www.laskater.com/
projects/uipAVR.htm), Paul
Hill's exmaple code (http://homepages.which.net/~paul.hills/Embedded/ShiftReg.c) and the
Ethernut project
(http://ethernut.de/)

Check INSTALL for instrutions to install this software and LICENSE to see its license.
```

./avr/LICENSE:

```
Unless otherwise stated, the files in this directory and subdirectories are licensed with
a two-clause BSD License,
and copyright is held by Nicolas Mendoza 2006 <mendoza@pvv.ntnu.no>

This source file is licensed under a two-clause BSD license

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

18. About this document

This document was made by Nicolas Mendoza as part of a thesis delivere spring 2007 at IDI, NTNU, Trondheim, Norway.

The text has been authored using Bluefish 1.0.7 on Ubuntu GNU/Linux. The writing has been done using XHTML and CSS.

Table of contents, cross-referencing and assembling the various XHTML parts was doing using Tailor, a custom assembling tool by Nicolas Mendoza, available at http://odc.opentheweb.org/svn/thesis/tailor

The conversion from XHTML to PDF was done with PrinceXML available from http://www.princexml.com.

The source file and this generated file is available at http://odc.opentheweb.org/svn/thesis/