AssignmentTDT4258 Assignment 3 - Group 1

Sondre Lefsaker André Philipp Håvard Wormdal Høiby

April 24, 2013

1 Abstract

Contents

2 Introduction

3 Overview

3.1 Led Driver

3.2 Button Driver

3.3 Graphics

3.3.1 Main graphics

The graphics of the game is contained within the /graphics folder. Here the **Screen** object is responsible for communication with the framebuffer through the **/dev/fb0** file. The **Canvas** object holds an array of **Shape** objects. Its has a method called **CanvasPaint** where it renders the screen in an 320x240 **Pixel** array contained within the **Screen**. When the whole array has been updated the **Canvas** object ask the **Screen** object to copy the contents of its internal buffer to the framebuffer. This causes the LED screen on the device to be updated.

All graphical objects are inherits their main structure of the **Shape** object. The most importaint part is the **paint** method which **Canvas** uses to render all the objects on the internal buffer.

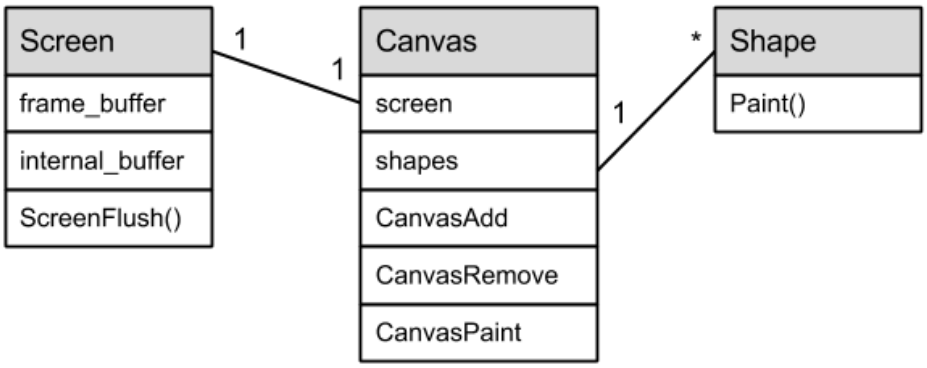

Figure 1: UML for the main part of graphics

3.3.2 Images

For all images the 24 bit version of the **BMP** (Bitmap file format) was implemented. This is done by the object **Bitmap** is hidden behind the **Image** object so that it is easy to add more file formats. The reason that bitmap was chosen is that the 24 bit version of it corresponds almost directly to the layout of the framebuffer, so the implementation was straight forward.

3.3.3 Graphical objects

Both objects for drawing lines and rectangles are where used only in the testing of the graphics module and are not used in the game. They serve as examples on how to develop more graphical objects.

3.4 Sound

3.5 Game

4 Solution

4.1 Led Driver

4.2 Button Driver

4.3 Graphics

4.4 Sound

4.5 Game

5 Test Report

6 Discussion

7 Conclusion