Project Report

by
Carl Oechsner

Table of Contents

1. Overview

Polly ...what? is inspired by the classic children's tabletop game Don't Break The Ice! first published in 1968. In the original version the players have to hit “ice”-cubes with a tiny hammer that are clamped within a frame. Placed on the middle cube there is a tiny polar bear which must not fall down while the cubes are being hit out of the frame. Whoever makes the polar bear fall off the platform has lost the game.

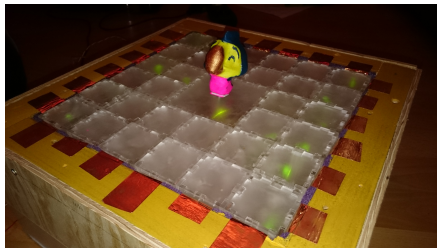

Fig. 1: Top view of the game

Our idea was to build a “pimped” version of this game including cubes that form a 2D LED matrix, an active game token to place in the middle and game logics that allow to add some difficulty to the game using an Arduino Mega.

The next sections will describe in detail the particular parts of our final system whose complete circuit diagram is shown in Fig. 2 and can be downloaded in Fritzing file format here.

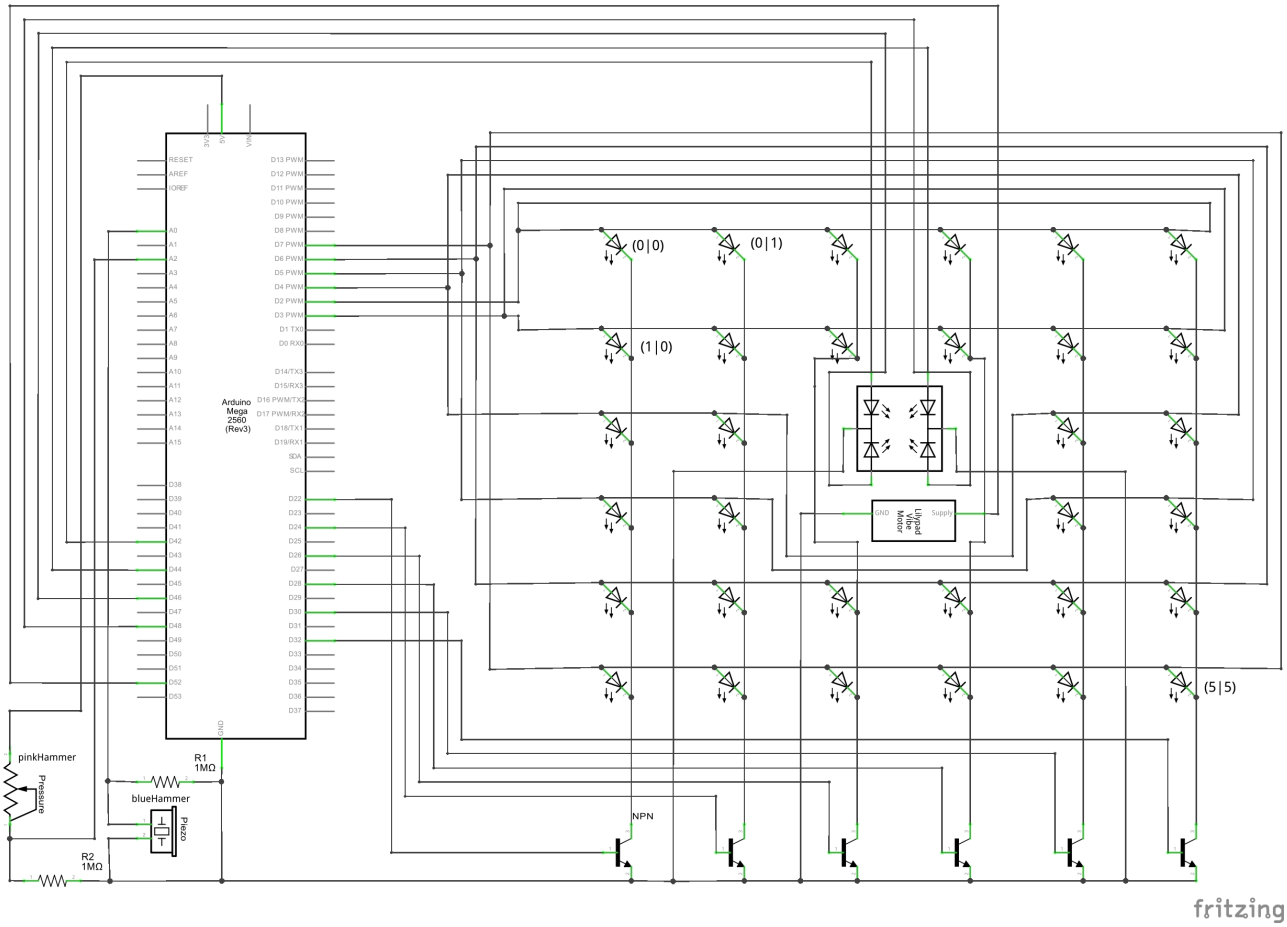

Fig. 2: Complete circuit diagram of "Polly ...what?". (PDF)

2. LED-Cubes

The game consists of 33 custom-cut acrylic glass (PMMA[1], Plexiglas®, perspex® or Lucite®) cubes of which 32 were 4cm in edge length and one 8x8x4cm in size to be the centre cube. The particular side walls were cut on a 40W laser cutter and glued together with superglue (UHU).

To use the cubes as a 2D LED matrix, we wanted the system to be as flexible as possible and the cubes to be easily exchangeable. We thought of how to simply connect the cubes and came up with a connector consisting of two push pins (9mm in diameter) soldered to each side of a copper wire that leads from one side of a cube to the other. The pins were pre-processed (hit with a hammer) to make their heads flat.

Before cutting there was an immersion engraved into the outside centre of each sidewall (diameter: 11mm) so that the pin heads would fit evenly with the rest of the surface. Additionally an aperture 2,5mm in diameter was cut out in the centre of the immersion. To get the best results, we used the values in Table 1. The original Illustrator file can be downloaded here.

The pins were soldered onto the wire already put through both opposing apertures. The wire was then swirled in such way that it pulls both evenly towards the surface. Then a green LED was soldered to the appropriate wires and hot glue was applied for stability and isolation purposes (see Fig. 4).

For the centre cube we used an RGBB LED to give the players feedback by not only different blinking patterns but also by displaying different colours. The plus and minus contacts coming in from the adjacent "normal" cubes are simply looped through to the opposing sides.

The blocks were then processed with sandpaper which results in the LED light being dispersed more evenly over the surface.

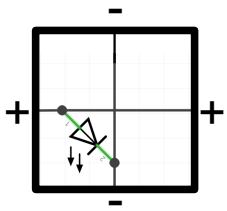

Fig. 3: Schematic depiction of one LED cube

Settings for processing 3mm PMMA	
Vector settings	
Intensity	40%
Speed	40%
Frequency	5000Hz
Raster settings	
Intensity	100%
Speed	12%

Table 1: Used settings to process 3mm thick RMMA on a 40W laser cutter

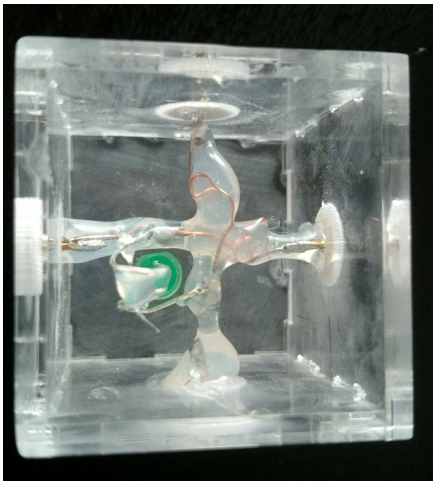

Fig. 4: Inside view of a cube

3. The frame

The frame was constructed using four scantlings cut out from an old shelf which were cut to a length of 28,7 cm and screwed perpendicularly onto each other. This length took into account the width of one row consisting of six cubes, the thickness of the foam and the width of one scantling. On the inside of the frame we glued foam cut out from a swimming aid to put pressure onto the rows and columns of cubes. As

1 *Polymethyl methacrylate:* Only use this material to cut, you get it the cheapest at the DIY store. When you go there, be sure to know the exact description and alternative descriptions. Don't trust anyone at the store to point you in the right direction or to sell you the right product. We named the game *Polly …what?* quoting a staff member whom we told that last time he sold us Polysterol (which is useless in a laser cutter, it just melts and stinks).

this is a crucial part of the whole game sticking together this flexible element was needed to compensate any inaccuracies.

Afterwards the positions where the cube pins will be located were marked and copper tape was placed vertically over the whole height of the foam onto all sides to ensure a maximum contact area. Also the two ends of each column and row were connected by wire to make sure signals can reach a cube from both ends.

Finally apertures cut out from plywood were attached with screws to each side of the frame to place the electronics on the inside and to have the cubes at a higher level, which is necessary to allow them to fall down.

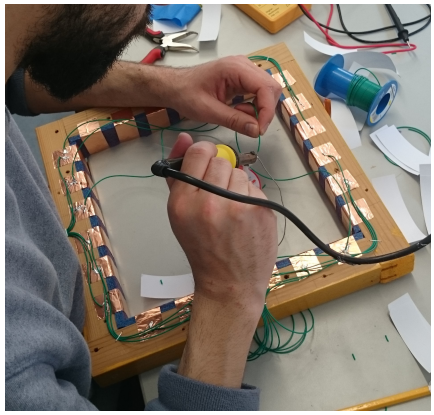

Fig. 5: Frame with copper contact tape

4. Polly

Polly is the main character in *Polly... what?* and sits on the central cube. Polly was designed in Blender and printed by a Ultimaker desktop 3D printer (see Fig. 6, Polly's Blender files can be downloaded here) The Model consists of two parts: the body and a closing cap. Inside the bird we put a little vibration motor to increase the game's difficulty by vibrating when certain blocks are being hit. Finally, Polly got a finish with colourful acrylic paint (see Fig. 1).

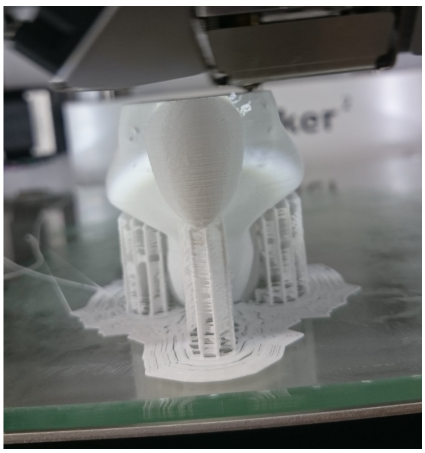

Fig. 6: Polly being born

5. Hammertime

As for the hammers we cut sticks out of a slat which we sanded and fixed perpendicularly to each other with a simple screw (see Fig. 7). At the tip of each hammer, we placed a sensor to detect whether a hit was performed, a pressure sensitive resistor for the pink and a piezo element for the blue hammer. To prevent the contacts being ripped out due to excessive gameplay a cord was weaved together with the wires and fixed with nails to the hammer and the frame respectively.

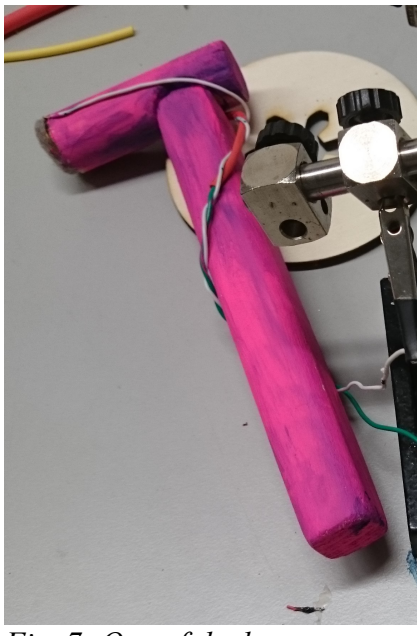

Fig. 7: One of the hammers used as input devices

6. Arduino

An Arduino Mega is implementing game logics and driving the 2D LED-Matrix. As for its low output current we decided to drive all LEDs in multiplexing mode, i.e. only light one LED at a time repeatedly for a very short time. By connecting each row to a pin (plus pole) and connecting each column to a transistor before it is grounded, we are able to address every cube individually. The different patterns to be displayed are stored in two-dimensional matrices just like the following:

```
//displays a "W" on the matrix
bool wMatrix[squareWidth][squareWidth] = {
  {w,f,f,f,f,w},
  {w,f,f,f,f,w},
  {w,f,f,f,f,w},
  {w,f,f,f,f,w},
  {w,f,w,w,f,w},
  {f,w,f,f,w,f}
};
```

Every time the `loop()`-Function is called a function is called, that loops over a certain array (in this case `drawW()`). As there is no easy possibility to hand over a 2D-array to a function on the Arduino, we wrote a function for each different pattern. When looping over the array and detecting a `true`-value, the current indices are sent to a function that shortly lights the corresponding LED by applying voltage to the dedicated row and the transistor connected to the dedicated column at the same time (`blinkOnce(int row, int col, int del)`).

While the game is not started yet, the letters “S”, “W” and “H” are shown successively, as the name of the course framing this project is “Sketching with Hardware”. As soon as both hammers are hit at the same time, the game switches from “demo mode”-status to “game started”-status. A certain tolerance (`sameTimeThreshold`) was introduced as it is almost impossibly difficult to exactly hit at the same time.

If the game has been started, four random cubes are picked, which will then be Polly's “danger cubes”, that must not be hit otherwise Polly gets angry and starts jumping around which can be devastating depending on the progress of the game.

Each player has three hits per round. If an input is detected on one of the sensors described in 5. that is higher than a pre-set threshold, it counts as a hit and a counter is being decremented. If all hits have been used up, the LED in the centre block turns red indicating that the round is over. Additionally, after each hit all lights turn off shortly to give feedback to the player.

7. Summary and Outlook

When we started this project, none of us had greater knowledge of electronics, laser cutting or programming an Arduino nor could we imagine the effort realizing this game afforded. Considering the very tight time frame of the project and the complexity of the system the outcome is quite satisfactory.

We now have the base to extend *Polly.. what?* with more features to improve the gaming experience. A function that

allows for finding out if there is actually a current flowing out of a digital Arduino output would allow for finding out which cubes are still in the game and which are not and enable a better game state detection.

Using foam as an element to put pressure onto the cubes works well but its stability decreases over time. During construction we thought of a system using bent acrylic class acting as a feather pressing against each row individually.

These improvements will be the milestones until the next exhibition of our prototype in November 2016. In summary, this project was a great, fun and instructive experience, not least thanks to the expertise of our supervisors and of course the creative ideas and hard work of my team partner.