



User Friendly Tangible Construction Kit


Bachelor's Thesis
submitted to the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

*by
Alexander Neumann*

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Dr. Ulrik Schroeder

Registration date: 21.07.2016
Submission date: 29.09.2016



Eidesstattliche Versicherung

Name, Vorname

Matrikelnummer

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/
Masterarbeit* mit dem Titel

selbständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort, Datum

Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Ort, Datum

Unterschrift

Contents

Abstract	xiii
Überblick	xv
Acknowledgements	xvii
Conventions	xix
1 Introduction	1
1.1 Thesis Structure	2
2 Related work	5
2.1 Tangibles on Capacitive Touch Screens	5
2.2 Hardware & Software Specification	9
2.3 User Friendly Manual Guidelines	11
3 User Friendly Tangible Construction Kit	15
3.1 Measurements & Current Design	15
3.1.1 Interviews	16

3.1.2	Board Design	17
3.2	Arduino	17
3.3	Tangible Shield	18
3.3.1	Schematics	19
3.4	Tangible Library	20
3.4.1	Light Sensor	21
3.4.2	Field Sensor	21
3.4.3	Bluetooth Communication	22
3.4.4	Communication Protocol	23
3.5	Tangible Case	24
3.5.1	OpenSCAD	24
3.6	User Friendly Manual	25
3.6.1	Quickstart Guide	26
3.6.2	Requirement List	27
3.6.3	Tangible Case and Shield Instructions	27
3.6.4	Arduino Template	28
4	Evaluation	29
4.1	Participants	29
4.2	Task & Procedure	29
4.3	Questionnaire	30
4.4	Results	30

5	Summary and future work	33
5.1	Summary and contributions	33
5.2	Future work	34
A	Arduino Tangible Building Instructions	35
B	User Study Questionnaire	47
	Bibliography	51
	Index	55

List of Figures

2.1	TUIC	6
2.2	ZebraWidget	7
2.3	PERC	8
2.4	Exploded Views	12
3.1	Lab Tangible Board Design	16
3.2	Breadboard Circuits	18
3.3	Arduino Uno Tangible Shield	19
3.4	Arduino Nano Tangible Shield	20
3.5	Tangible Case Parts	25
3.6	Copper Foil Grid	26
3.7	Constructed Tangible	27
3.8	Structure of Instructions	28
4.1	Time Measurement Tangible Case	30
4.2	Time Measurement Tangible Shield	31

A.1	Manual we handed out	36
A.2	Manual we handed out	37
A.3	Manual we handed out	38
A.4	Manual we handed out	39
A.5	Manual we handed out	40
A.6	Manual we handed out	41
A.7	Manual we handed out	42
A.8	Manual we handed out	43
A.9	Manual we handed out	44
A.10	Manual we handed out	45
A.11	Manual we handed out	46
B.1	Consent form	48
B.2	Questionnaire we handed out	49
B.3	Questionnaire we handed out	50

List of Tables

3.1	AT Commands	23
3.2	Communication Possibilities	24

Abstract

In this bachelor thesis we wrote a user friendly manual to build tangibles for capacitive touch screens based on an Arduino board. Tangibles are haptic input devices and there are quite a lot of approaches. Voelker et al. [2015] introduced tangibles which are not filtered after a while by the touch screen. That is why our construction kit focuses on those tangibles. Therefore we created a shield for an Arduino Uno and an Arduino Nano. With this shield, the Arduino has the required functionality such that it will not be filtered out. Some of the required components are already installed on the Arduino, such that the shield needs less soldering than the approach of Voelker et al. [2015]. It is easy to add functionalities and stack other shields on top of the tangible shield. The manual is addressed to everyone who has access to a FabLab and is interested in developing with tangibles.

Überblick

In dieser Bachelorarbeit wurde eine benutzerfreundliche Anleitung ausgearbeitet mit der man Tangibles die auf kapazitiven Touchscreens funktionieren herstellen kann. Die Tangibles basieren auf einem Arduinoboard. Tangibles sind haptile Eingabegeräte und mittlerweile gibt es auch viele verschiedene Ansätze. In dieser Arbeit beziehen wir uns auf die vorgestellte Variante, bei denen Tangibles nach geraumer Zeit nicht von dem Touchscreen herausgefiltert werden, von Voelker et al. [2015]. Um dies zu realisieren haben wir ein Shield für den Arduino Uno und Arduino Nano entwickelt. Mit diesem Shield hat der Arduino die nötigen Funktionalitäten um nicht herausgefiltert zu werden. Einige der benötigten Komponenten sind bereits auf dem Arduino verbaut, sodass man weniger auf das Shield löten muss. Da wir einen Arduino benutzen, ist es leicht das Tangible mit weiteren Funktionen auszustatten. Es ist außerdem möglich, ein weiteres Shield auf dem Tangible-Shield zu stapeln. Die bereitgestellte Anleitung ist an alle adressiert, die Zugang zu einem FabLab haben und an Projekten mit Tangibles Interesse haben.

Acknowledgements

I would like to thank Prof. Dr. Borchers for supervising my thesis and Prof. Dr. Schroeder for being my second examiner.

I want to thank my supervisor Christian Cherek, M.Sc., for providing feedback and his guidance through this thesis.

Special thanks to Jan Thar who helped me a lot while I was working at the FabLab.

And of course I want to thank everybody who supported me during my thesis, like my family and friends.

Conventions

Throughout this thesis we use the following conventions.

Text conventions

Definitions of technical terms or short excursus are set off in colored boxes.

EXCURSUS:

Excursus are detailed discussions of a particular point in a book, usually in an appendix, or digressions in a written text.

Definition:
Excursus

Source code and implementation symbols are written in typewriter-style text.

`myClass`

The whole thesis is written in American English.

Download links are set off in colored boxes.

File: [myFile](#)^a

^ahttp://hci.rwth-aachen.de/public/folder/file_number.file

Chapter 1

Introduction

Tangibles, physical objects which can be placed on the surface of a touchscreen, offer the user haptic feedback. Bigger touch screens do not necessarily have to be placed vertically, they can also be placed horizontally and used as a touch table. With this setup, there are more interaction possibilities for multiple users working with the system. Controls can be placed anywhere on the screen and with multiple users every corner can be reached. Tangibles solve the lack of haptic feedback but they are not used much, although the approach of tangibles is not new. There exist different approaches of tangibles and for different devices.

Multi-touch interaction is widespread. Everyday people are using their smart phones, laptops and tablets which mostly have a capacitive touch screen. Tangibles would give those people the ability of eyes free interaction. Voelker et al. [2013] already presented a type of tangibles, so called PUCs (passive untouched capacitive widgets), which are recognized on various commercial unmodified capacitive touch screens. PUCs are not available in stores but they are easy and fast to build. PUCs have the disadvantage that they are filtered out by most of the capacitive touch screens after a few minutes.

PUCs

A more recent approach from Voelker et al. [2015] solve also this problem. They introduced PERCs, persistent capacitive tangibles, which know whether they are placed on

PERCs are not that easy to build

a capacitive touch screen or not. This is done with a field sensor and a communication to the computer. PERCs offer the opportunity for various new applications and more user friendly interaction with haptic feedback but they are neither available in any stores nor easy to build.

Students often come up with project ideas and these might involve tangibles since touch input is a popular requirement for innovative applications. Tangibles offer haptic feedback and are great for games and educational purposes. The best choice is an active tangibles which is not filtered out by the touch screen, the problem is that students do not know how to produce or obtain such a tangible. We set the requirement that the users are able to use a [FabLab](#)¹ (fabrication laboratory) where they have the necessary tools for building the tangible. With this bachelor thesis we offer them a user friendly tangible construction kit.

1.1 Thesis Structure

Related work

In this thesis we start with chapter 2 “Related work” where we first discuss different approaches on tangibles. For the construction kit we make use of the approach of Voelker et al. [2015]. We also have a look at the Arduino and why it is good for prototyping and also the other components used for this construction kit. At the end of this chapter we filter good design guidelines for writing a user friendly manual.

User friendly tangible construction kit

Chapter 3 “User Friendly Tangible Construction Kit” is divided into five different parts. First We declare some design decisions for the construction kit. The second part deals with the Arduino platform and the library that we provide for the communication with the computer. In the following part, we introduce the tangible shield for the Arduino. We could reduce the parts needed for the tangible functionality to six components. The fourth part describes the case design for the tangible. We provide a single file that can be

¹<http://hci.rwth-aachen.de/fablab>

used with a laser cutter to produce all the parts for the case. The last part of this chapter deals with the user friendly manual, in which we explain the structure and design decisions we made to create a user friendly manual.

Whether the manual is really user friendly can be found in chapter 4 "Evaluation". We conducted a small user study with ten participants who tested the manual and built tangibles. The results can be found in this section. The results are satisfying since the building time could be reduced to about an hour.

Evaluation

In chapter 5 "Summary and future work" we conclude the thesis with an overview of possible extensions of tangibles and improvements for the manual. Additional sensors and motors can be put onto the tangible shield and improve the user interaction with the capacitive touch screen.

Summary and future work

Chapter 2

Related work

2.1 Tangibles on Capacitive Touch Screens

For the construction kit we had to choose the type of tangible we want to use. Therefore, we set as a requirement that the touch screen should be unmodified and commercially available. In this chapter we first start with describing different approaches on tangibles and how they are used on touch screens. Using physical objects as input devices is not a new approach. Fitzmaurice et al. [1995] introduced Bricks, so called graspable user interfaces. With these physical objects the user has direct control to manipulate parameters. Ishii and Ullmer [1997] presented Tangible Bits, which describe a tangible user interface. These tangibles also allow to "grasp & manipulate" virtual objects. This is done by "coupling" the virtual object with the physical object. They also focused on the background information, such as light and sound.

TUIC is an approach which enables the interaction with tangibles on capacitive touch screens. Yu et al. [2011] presented three different approaches to achieve this interaction.

The first approach deals with passive recognition. Multiple markers are required to define an object id which is a disadvantage. It also requires a specific spatial distance.

The second approach improves these circumstances. There-

We search for tangibles which will work on unmodified capacitive touch screens

TUIC provides an active, a passive and a hybrid for capacitive touch screens.

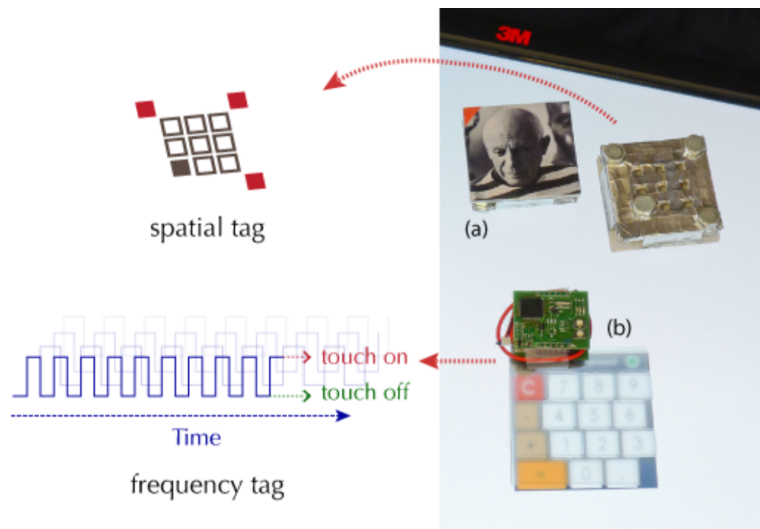


Figure 2.1: TUIIC. a) A passive approach which uses four touch points for the identification. b) Active approach showing the frequency tag used for the identification. Only one touch point is required.

fore an active modulation circuit is used to simulate the touches. The space and the number of touch points could be reduced. The frequency can be changed which enables the opportunity to change the object id. But solving issues opens new ones: The active version does not support orientation and has problems with fast movement. Both approaches are shown in figure 2.1.

The last approach is a combination of those two.

CapStones and ZebraWidgets are passive tangibles which allow different interaction techniques.

CapStones and ZebraWidgets are stackable tangibles introduced by Chan et al. [2012].

CapStones are passive blocks which can be stacked to a specific height and determined by the number of touches. So every stacked block adds a new touch point. After a while, the created touch points are filtered by the touch screen which is a well-known problem of passive tangibles.

ZebraWidgets are also passive tangibles which are made of a special material which are only vertically conductive. By touching this material a touch point is created under this object. A slider implementation is represented in figure 2.2. Using these tangibles it is possible to make more accurate

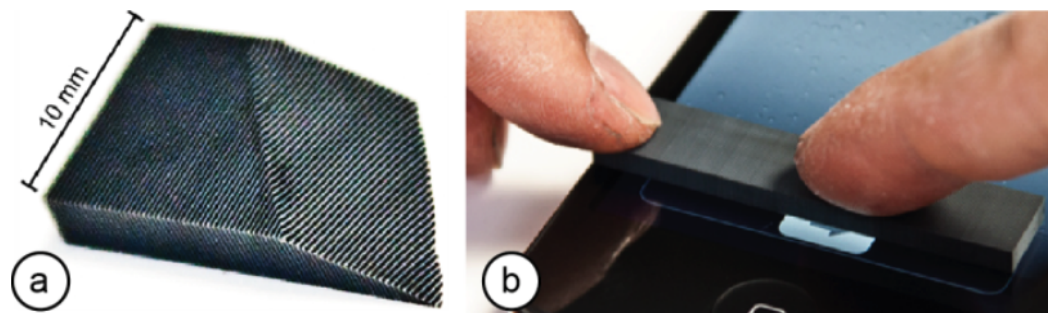


Figure 2.2: ZebraWidget. a) The material consists of alternating layers of conductive and non conductive layers. b) The sliding gesture to unlock an iPhone is transmitted through the material.

touch gestures, but they need continuous contact to create a touch which is a big disadvantage.

Sketch-a-TUI is a very simple approach for tangible prototyping. Wiethoff et al. [2012] present a low-cost technique of building passive tangibles by creating a box out of cardboard and adding conductive ink for the touch points. They offer different templates for paper objects. Designers have the chance to develop different types of tangible prototypes without knowledge of electronics. Missing code for the recognition makes this approach hard to use.

Sketch-a-TUI offers fast and easy prototyping of passive tangibles.

CapTUI is a tangible drawing tool consisting of three main components: a tangible, a recognition system and the drawing software. The tangibles developed in this approach are passive tangibles which work on capacitive touch screens. Blagojevic and Plimmer [2013] tried different conductive materials like aluminum foil, conductive ink, conductive rubber and conductive foam. They have measured the recognition performance of each material and achieved the best results using conductive foam. The recognition is done by using different patterns. Therefore, the distances of the edges must be distinct, exceptions are regular shapes like a circle. The recognition part was the main challenge and can still be improved.

Drawing with tangibles improved through CapTUI.

PUCs are passive untouched capacitive widgets. Voelker et al. [2013] demonstrated that these tangibles are recognized on capacitive touch screens without touching them.

PUCs, passive tangibles without touching them.

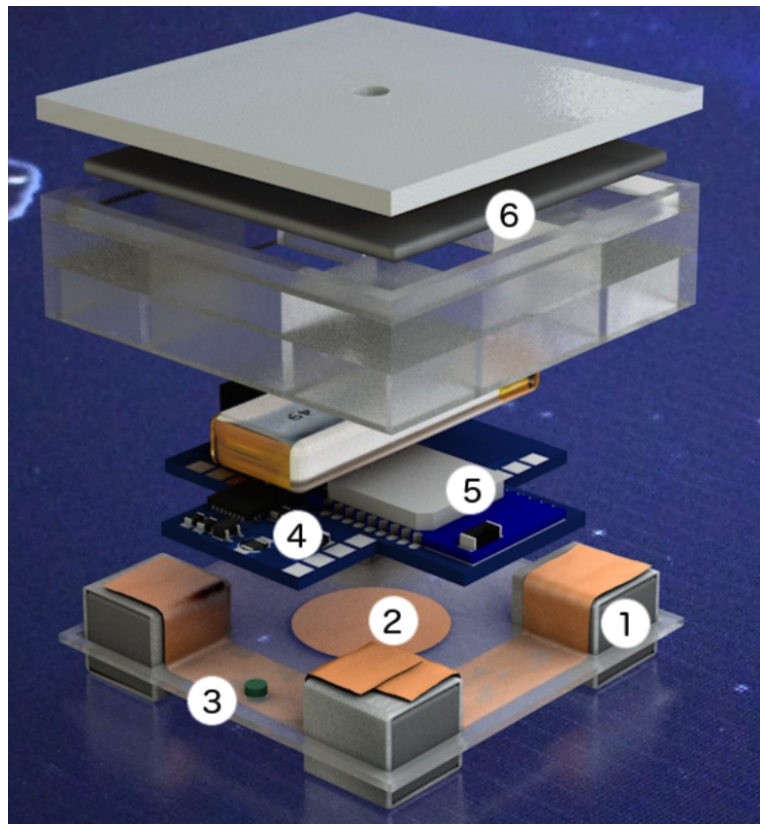


Figure 2.3: Structure of a PERC: 1) marker pattern, 2) field sensor, 3) light sensor, 4) micro controller, 5) BLE module, and 6) a lead plate.

The conductive markers beneath the tangible are connected with each other on a top layer, such that only the touch point marker are on the touch screen. The unused parts of the screen ground the marker and the touch points are created. These tangibles can also be entirely transparent and do not need any extra power or any internal electronic. But PUCs are also filtered out after a while on the touch screen without any interaction.

PERCs, persistent capacitive tangibles, which are not filtered out by the capacitive touch screen.

PERCs are active tangibles introduced by Voelker et al. [2015]. This approach is based on their previous approach PUCs. PERCs have their own electronic and can detect whether they are placed on a capacitive touch screen or not. This is done by a field sensor measuring the scan for touch

points of the touch screen. The information is sent via Bluetooth 4.0 (BLE) to the computer. A small light sensor is also added to ensure the orientation, if one touch point is falsely not recognized. The structure of a PERC tangible is shown in 2.3. The advantage of PERCs is not only that they are not filtered out by the touch screen they also allow stable orientation detection. The problem of fast movement with active TUICs is here also no big problem, since games like tabletop air hockey can be successfully played with PERCs [2015].

Most of the approaches are using passive tangibles, but these are filtered out after a while by most commercial touch screens after a few minutes. The most promising approach of active tangibles are PERCs. In this thesis we focus on the work of Voelker et al. [2015] and try to achieve an ease way of building this type of active tangible.

2.2 Hardware & Software Specification

Since we are building up on PERCs we need a recognition software which can distinguish between different active tangibles and identify them correctly. Linden [2015] introduced the Multitouchkit which is a software framework for touch input. It can detect passive and active tangibles and supports PUCs and PERCs. This framework is based on SpriteKit and can be used on iOS and MacOS. The Multitouchkit supports Windows 7 touch input and other devices. [TUIO](http://www.tuio.org/)¹ which is a protocol and API for tangible multi touch surfaces is also implemented and tracks relevant data.

Multitouchkit detects PUCs and PERCs.

We also have to look for hardware and software which can be used for the tangible to improve the construction process. [Arduino](https://www.arduino.cc/)² is a well-known platform for building electronic projects. Arduino runs on Mac, Windows and Linux. It is a good choice for people who just started working with electronics.

Arduino platform for electronic projects.

¹<http://www.tuio.org/>

²<https://www.arduino.cc/>

The Arduino platform can be used for educational purposes.

Juang and Lurr [2013] presented a two-wheel self-balancing robot. The core element of this robot is a Arduino board. They describe the design, the construction and the control of the robot. They point out that Arduino is a popular platform for education and development processes, including network control. They refer to the "up-to-date discussions" and that the Arduino project is open-source which helps the user to find their needs for their own project.

Arduinos are cheap, open-source and flexible.

An Arduino board was also used to emulate a keyboard and measure response latencies. Schubert et al. [2013] used an Arduino board instead of response boxes and data acquisition cards, because using an Arduino is much cheaper and more flexible. They have listed several reasons why they have used an Arduino. It is easy to write code onto the Arduino via the USB connection with the computer. The Arduino is modular and easy to modify since it is an open-source project. A big advantage is the low price of an Arduino and that there are shields which extend the functionality of the board.

Arduino has a large contributing community.

D'Ausilio [2012] have proofed that an Arduino can be used as a low-budget lab equipment which can be powered by a standard 9 Volt battery. They refer to the large community which has risen over the past years. Their test results show that Arduinos offer "complete independence, portability, and accuracy".

Another lab-based approach was given by Sarik and Kymissis [2010]. They presented a lab kit which uses the Arduino platform and is addressed to students. Arduinos are "low-cost hardware" but offer high-quality experience. Students can use these kits and work at home which is a big benefit for remote students.

Taking these references into account Arduino seems to be a good choice for the construction kit. Detailed explanations can be found in chapter 3.2 "Arduino".

2.3 User Friendly Manual Guidelines

Creating a user friendly construction kit requires a user friendly manual. Therefore, we are searching for principles and guidelines for giving user friendly building instructions.

The most popular approach for building instructions Lego. Lego is known for their manufactured toys for children. The basic parts of Lego are colored bricks, gears, and other small pieces. With those parts, it is possible to build nearly everything imaginable. Another example are assembly instructions. Those are similar to the Lego instructions but have an other target group. Assembly instructions are not for fun purposes and not targeted to children, but they are also supposed to be easy to follow.

Lego instructions are very easy to follow.

A common technique used by those instruction groups is the use of exploded views. Driskill and Cohen [1995] presented a tool for creating exploded views.. These views are easy to understand and do not require any expertise. An example for an exploding view can be found in figure 2.4.

Exploded view is a common technique.

Agrawala et al. [2003] presented design principles for assembly instructions. They claimed that it is difficult and expensive to create instructions which are easy to follow. They filtered two major tasks which are needed to design instructions: Planning and presentation. Planning requires a sense of choosing the right sequence for the building order. Such that it is easy for users to understand and follow. Presentation describes the visual feedback. The operations have to be described in an understandable way.

Two tasks: Planning and presentation.

Agrawala et al. [2003] proposed the following design principles:

Hierarchy and grouping of parts. People prefer parts built simultaneously or one after another if they have something in common or belong together.

Hierarchy of operations. People are used to attach small pieces and fasteners at the beginning and go on with more significant parts depending on their size, symmetry and

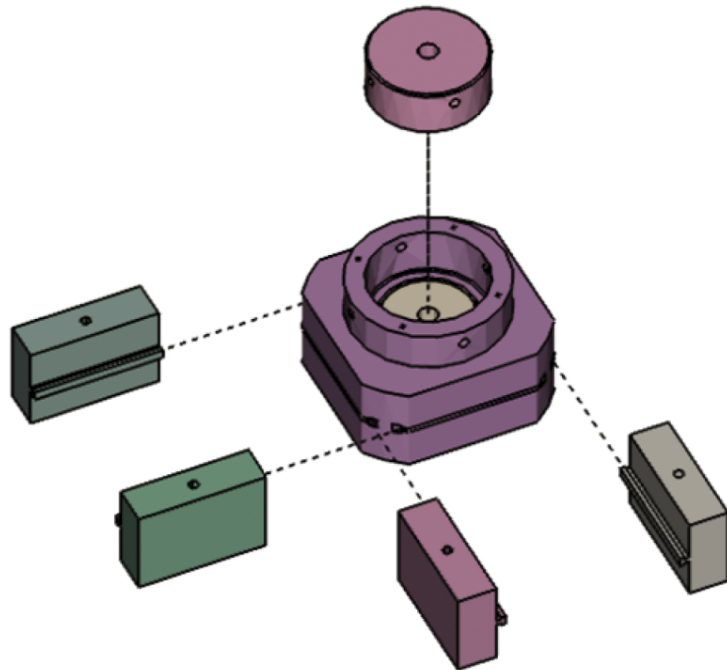


Figure 2.4: Example of exploded views. Image source: Agrawala et al. [2003]

their function. Agrawala et al. [2003] prefer a two level hierarchy using significant parts and less important parts together.

Step-by-step instructions. It is common using a sequence of diagrams describing the building process instead of one diagram containing all operations. Repetitive operations are skipped since the user can use a single diagram multiple times if it is repetitive.

Structural diagrams and action diagrams. Structural diagrams show the parts in their final position. The user needs two different structural diagrams to gather the building operations. Action diagrams are ranked higher than structural diagrams. They indicate the parts that have to be attached with arrows or guiding lines.

Orientation. Objects have natural orientations, but sometimes it is useful to have other orientations regarding the gravity or easier attachment.

Visibility. Maybe the most obvious but also the most important principle. New parts attached to an assembly have to be visible. It is also important that not only new parts are visible because then it is unclear how these parts have to be attached.

Since we are focusing on usability, user experience we have to include human factors and take psychological factors into account. Hodgson listed a lot of guidelines for writing user manuals. These guidelines are based on best practice principles, principles of good information design, human perception, cognition and psychology (reading) and own experience (user testing manuals). We filtered out some guidelines what we think are the most important ones and which were not mentioned before:

Guidelines from a more psychological view.

Physical copy. Provide a physical copy of the manual and not only a pdf.

Quick start guide. Include an accessible one-page quick start guide.

Users. Test the manual with real users.

Colors. Effective use of color.

Sections. Divide the sections due to their chronology, frequency, functional and expertise.

Consistency. The instructions should match the product.

Chapter 3

User Friendly Tangible Construction Kit

Capacitive touch screens are currently widespread. These touch screens can detect tangibles with conductive material on the ground Voelker et al. [2013]. Since passive tangibles are filtered out it is recommended to use active tangibles Voelker et al. [2015]. There is a problem using active tangibles. Currently, you cannot buy them and as a student or a person who has a great project idea involving tangibles you have no idea how to build an active tangible. The goal of this thesis is to fill this gap and provide a user friendly manual which helps people to build an active tangible for fast prototyping. The construction kit contains a case for the tangible, a shield for an Arduino, communication software and a user friendly manual with instruction steps.

3.1 Measurements & Current Design

Before we started with the development of a construction kit we had to measure problems of the current version of active tangibles. In the winter term 2015/16 students participated in a lab course where they had to develop a [tower defense game](#)¹ with active tangibles.

Students had to build active tangibles.

¹<https://www.youtube.com/watch?v=1ogK7QBIPeg>

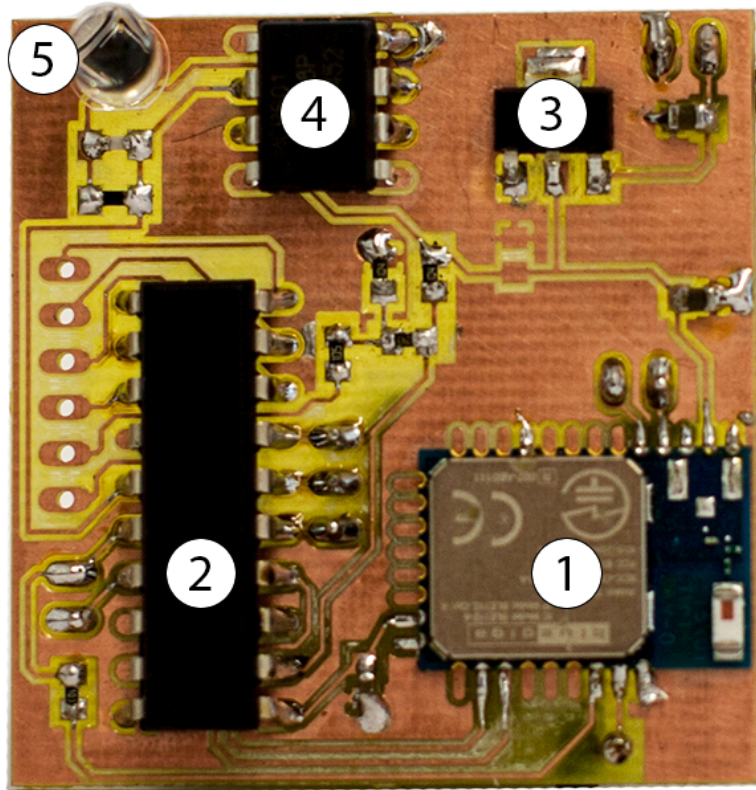


Figure 3.1: Components of the board: 1) Bluetooth module, 2) micro controller, 3) voltage regulator, 4) operational amplifier, 5) photodiode, resistors and capacitors

3.1.1 Interviews

We asked the participants several questions due to their development process. The group who built the tangibles consisted of six people. They all complained about soldering the board was time elapsing and the most difficult part. Another big problem is the software overhead caused by the Bluetooth communication. Out of these six people, one had experience building tangibles and said that it takes about an hour to build an active tangible, the others claimed that it needs at least three hours to build an active tangible and there might be a chance that the tangible does not work.

Building an active
tangible takes a lot of
time.

3.1.2 Board Design

We had a deeper look at the current version of the board. Figure 3.1 shows the board design of the students from the lab course. For a student without soldering experience, this board contains a lot of components which have to be soldered. They are using the [MSP430G2553](#)². The micro controller has to be flashed for writing the software on it. It would be sufficient to just use the Bluetooth module. It would be possible to write the communication software on the module itself, but still, there are quite a lot of components to be soldered.

The board of the lab course consists of multiple components.

3.2 Arduino

We switched to another platform because we want to make the building process easier and the introduced approach uses the MSP430G2553. For a novice user it is difficult and unwieldy to flash a micro controller. Programming an Arduino is quite easy, you just connect the Arduino to your computer and use the Arduino IDE to write your program. But using an Arduino has also other advantages:

Arduino, the easy way.

- Cheap
- Open-source project
- Built-in components
- Fast prototyping

In this thesis we focused on the Arduino Uno and Nano. Both are using the [ATmega328P](#)³. The Arduino Nano is not officially produced anymore but it uses the same micro controller as the Arduino Uno. This way it was easier to create

²<http://www.ti.com/lit/ds/symlink/msp430g2253.pdf>

³<http://www.atmel.com/Images/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P.datasheet.pdf>

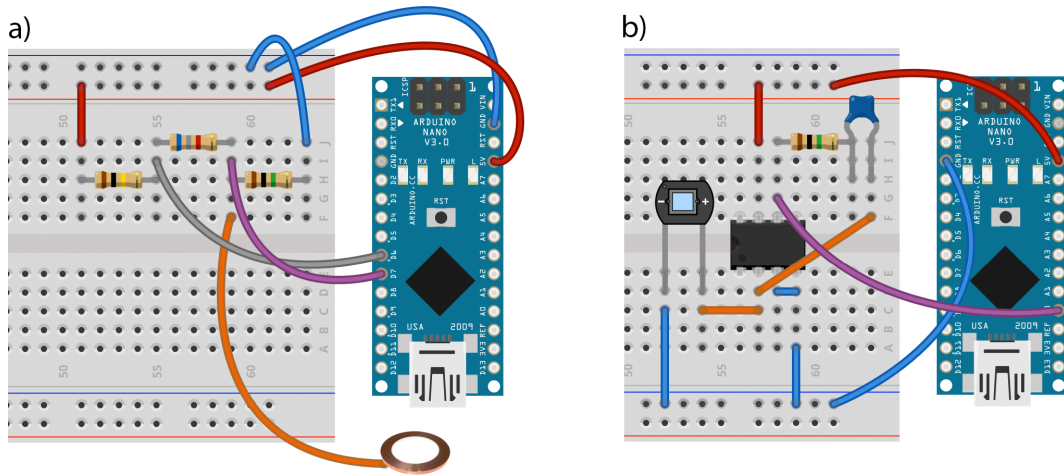


Figure 3.2: a) Circuit of the field sensor. The gray cable is connected with the positive comparator input and the purple cable with the negative comparator input. The orange cable is connected with copper foil which will be placed slightly over the touch screen. b) Circuit for the light sensor. The value can be read out of analog pin 0.

a shield for both devices. One could also choose the Arduino Micro, but it uses the [ATmega32U4](http://www.atmel.com/Images/Atmel-7766-8-bit-AVR-ATmega16U4-32U4.Datasheet.pdf)⁴ which has different settings that you have to look up in the datasheet. A clone of the Arduino Nano can be purchased for about 3 Euro apiece. The ATmega328P has an integrated comparator and the Arduino has a built in voltage regulator. So we do not have to use those components on the shield.

3.3 Tangible Shield

Our shield consists only of six components.

For the tangible functionality we created a shield with the necessary components. We were able to reduce the parts to a count of six. The shield consists of four resistors, one capacitor and an operational amplifier. The rest is done with pin header. We wanted that the shield is easy to create and the whole tangible is modular. Using pin header makes it easy to exchange parts. We started with a simple circuit on a breadboard (see figure 3.2a for the field sensor and figure

⁴<http://www.atmel.com/Images/Atmel-7766-8-bit-AVR-ATmega16U4-32U4.Datasheet.pdf>

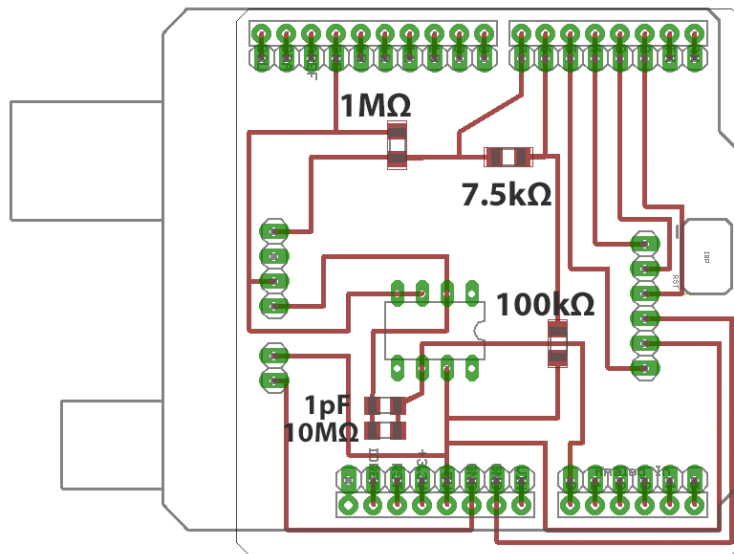


Figure 3.3: Schematic of the Arduino Uno shield created with Eagle.

3.2b for the photodiode). This allowed us to verify our results and write the software which we already presented in 3.4 “Tangible Library”. The Arduino is supplied by a 9V battery which is connected to the shield.

3.3.1 Schematics

Since the breadboard version is working and uses less components, we designed a shield for the Arduino Uno (see figure 3.3) and Nano (see figure 3.4). Therefore, we used a program called [Eagle](https://cadsoft.io/)⁵. Eagle is a cross-platform software developed by CadSoft Computer GmbH. It can be used for designing circuit diagrams. Eagle also provides a PCB layout editor which allows back annotations to the schematic. For the Arduino Shield layouts we used [SparkFun’s public Eagle PCB footprints](https://github.com/sparkfun/SparkFun-Eagle-Libraries)⁶.

We provide schematics for the Arduino Uno and Arduino Nano.

⁵<https://cadsoft.io/>

⁶<https://github.com/sparkfun/SparkFun-Eagle-Libraries>

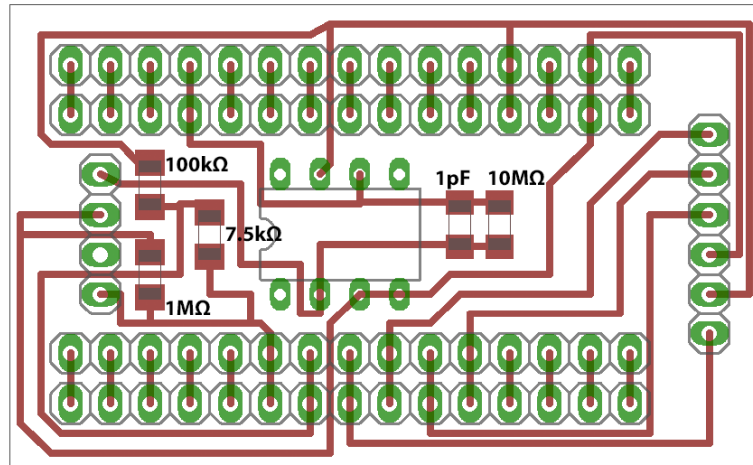


Figure 3.4: Schematic of the Arduino Nano shield created with Eagle.

3.4 Tangible Library

The library is written with the Arduino IDE and consists of a single class. The user just has to import the required library and create the desired tangible object.

```
1 Tangible myTangible(3, 2, 600);
```

The first two parameters are for the RX, TX pin of the Bluetooth module. The third parameter is the threshold for the photodiode. An object of the tangible class needs to be updated in the `loop` by calling the `update` method of the tangible.

```
1 void loop() {
2   tangible.update();
3 }
```

This method checks the value and of the photodiode. For the field sensor we are using the `ISR` routine of the analog comparator, where we have to call the `checkTable` method to check whether the tangible is on the touch screen or not.

```
1  ISR(ANALOG_COMP_vect ) {  
2    tangible.checkTable();  
3  }
```

We provide a template which can be found in chapter A “Arduino Tangible Building Instructions”.

3.4.1 Light Sensor

By using the previously described shield, the photodiode is connected to pin `A0`. Therefore, we just have to make an analog read call and distinguish whether we have a white spot below the tangible or a black spot (used for orientation determination if one touch point is not recognized). To transfer the read analog value into 0 or 1 we use the threshold given as a parameter to the constructor. This is important because there are multiple reasons why we have to adjust this value: light conditions, screen brightness and the material of the tangible case.

We need the light sensor to determine the orientation if one touch is missing.

3.4.2 Field Sensor

In order to use the built-in comparator the `ACD` bit has to be settled down in the Analog Comparator Control and Status Register (`ACSR`). We set the `ACSR` variable with the following value:

The field sensor will tell us whether we are on the table or not.

```

1 ACSR =
2     (0 << ACD) |
3     (0 << ACBG) |
4     (0 << ACO) |
5     (1 << ACI) |
6     (1 << ACIE) |
7     (0 << ACIC) |
8     (1 << ACIS1) | (1 << ACIS0);

```

The negative input of the comparator is located at pin `AIN1` (digital pin 7) and the positive comparator input is at pin `AIN0` (digital pin 6).

3.4.3 Bluetooth Communication

We use a `SoftwareSerial` to not block the communication for writing sketches onto the Arduino.

For the Bluetooth communication we make use of a `SoftwareSerial`. The Arduino has built-in support for serial communication on pins 0 and 1, but those pins are also used for the USB connection with the computer. If we still use these pins for the Bluetooth communication, the user will not be able to upload a new sketch to the Arduino, because the Bluetooth module will block the serial communication. The `SoftwareSerial` allows us to use other pins for serial communication.

Since we only need to tell the computer whether we are on the touch screen or not and the current photodiode value we only need to implement the transmit functionality. This is done by using the `print` function of the `SoftwareSerial`.

We are using the [HM-10](http://www.mikrocontroller-praxis.de/data_article/HM-10/Serial.Port.BLE.Module.Master.Slave.HM-10.pdf)⁷ BLE since this is one of the BLEs with pins. Most of the other modules are regular Bluetooth modules or have to be soldered. There is a small problem using this module: The Multitouchkit uses multiple characteristics for retrieving data of the tangible. Characteristic `0xFFE1` is used for the field sensor and `0xFFE2` is used for the light sensor value [2015]. We are able to change the

⁷http://www.mikrocontroller-praxis.de/data_article/HM-10/Serial.Port.BLE.Module.Master.Slave.HM-10.pdf

characteristic of the BLE using AT commands. The latest firmware running on the BLE offers a [list of AT commands](#)⁸. Table 3.1 “AT Commands” shows useful AT commands like setting the name, test if the module is connected and how to change the characteristic, but changing the characteristic needs a restart of the module. Using multiple characteristics at the same time is with this approach not possible. Therefore, we made slight changes in the Multitouchkit to use just one characteristic.

AT commands for changing the name or the characteristic.

Characteristic:		
Send	Receive	Parameter
AT+CHAR[P1]	OK+Set:[P1]	P1: 0x0001 0xFFFE

Useful functions:		
Send	Receive	Parameter
AT	OK	None
AT+NAME[P1]	OK+Set[P1]	P1: module name

Table 3.1: AT Commands

3.4.4 Communication Protocol

The presented version of the Multitouchkit uses String representations of the decimal digits 0 and 1 to indicate whether something is present or not. Previously we described that we had to make small changes to the Multitouchkit and send the data with one single characteristic. Therefore, we had to implement an easy protocol. We chose the symbol “#” as delimiter to separate the field sensor value from the light sensor value. Anything sent after the light sensor value (also separated with a “#”) can be used as additional data. Table 3.2 “Communication Possibilities” shows the different possibilities of the communication protocol. The case “0#1” is not possible because we do not track the light sensor value if we are not on the touch screen.

We send the data over one characteristic.

⁸<http://www.jnhuamao.cn/showNews.asp?id=14>

Sent	Meaning
0#0	Not on the table and a black. (Default)
0#1	Not on the table and a white. (Not possible)
1#0	On the table and a black.
1#1	On the table and a white.

Table 3.2: Communication Possibilities

3.5 Tangible Case

Using the laser cutter is faster and cheaper.

In a FabLab we usually have the access to a laser cutter and a 3D printer. We wanted a case design that is easy and fast to fabricate. 3D printing produces a case without any work left, but it is more expensive than using the laser cutter and takes much more time. So using the laser cutter leaves us some short building instructions but it is very cheap and fast. That is why we chose the laser cutter for creating the case of the tangible.

Use non-transparent material.

The case can be made of one single material. We suggest using a non-transparent material because the light will shine through the material and disturb the measurements for the orientation. For demonstration purposes we used transparent acrylic glass.

Figure 3.5 shows the different parts of the case. Of course, this figure is not really material saving, we provide an SVG file which is compact (the parts of e) are placed in the gaps of b) and d)).

3.5.1 OpenSCAD

The SVG is generated by a program called [OpenSCAD](http://www.openscad.org/)⁹. OpenSCAD is available for Windows, Linux, and Mac. It allows creating solid 3D CAD objects and also 2D shapes. This program does not focus on artistic aspects. The object is created out of a script where the user can use functions

⁹<http://www.openscad.org/index.html>

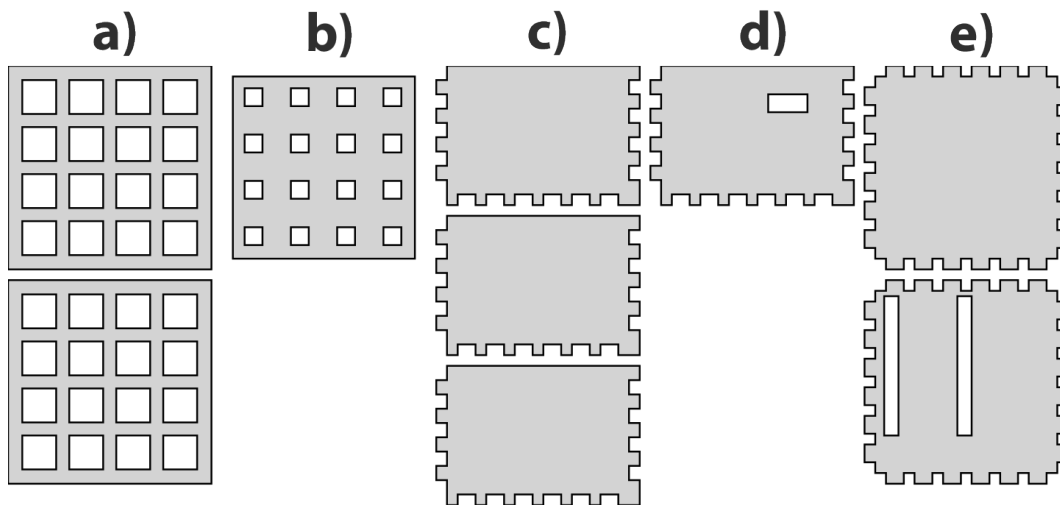


Figure 3.5: Parts of the case. a) Bottom part, b) separation layer, c) side parts d) side part with USB slot, e) top layer (normal or with slots for another shield).

to describe the 3D object. This makes it easy to adjust the tangible case for another Arduino Board by changing a few parameters.

The tangible case is modular.

The marker pattern for the touch points is created by a grid that is surrounded with copper foil (see figure). It is important that one side is not connected to the grid, because we just want to create three touch points [2015]. Thus one magnetic shielding pad has to be placed at this position. It is recommended to use a square layout for the pads since most other combinations will not work that well.

3.6 User Friendly Manual

We created a manual with different parts. Regarding to chapter 2.3 “User Friendly Manual Guidelines” we looked at different types of instructions needed to build the tangible and split the manual into different parts. The manual starts with a quickstart guide. Then we have a list of components needed for the construction. The manual continues with instructions for the case and the shield. At the end there is a template with the Arduino code. The separation

The manual is divided into different parts.

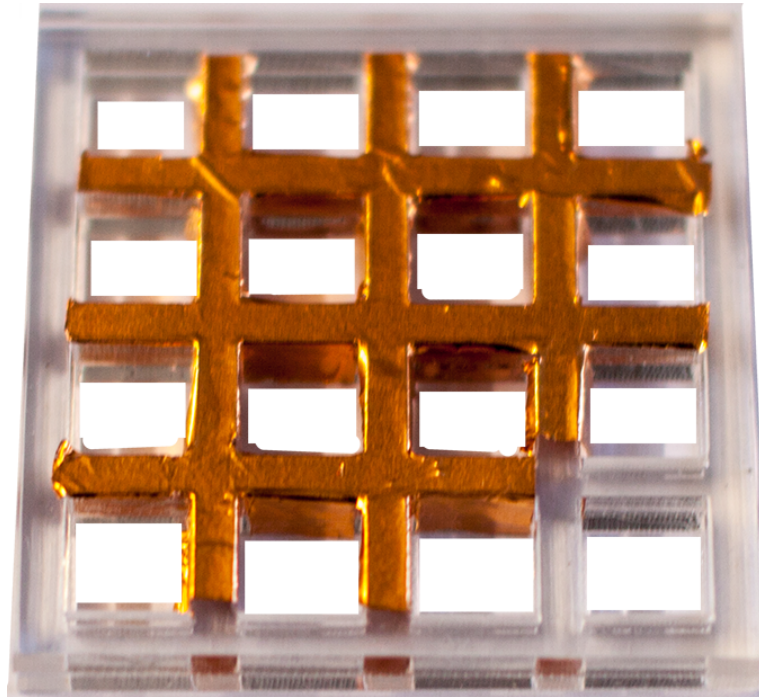


Figure 3.6: Ground layer grid of the case surrounded with copper foil.

is caused by the types of instructions and the order due to their frequency. Figure 3.7 shows an example of a tangible built with the provided manual.

3.6.1 Quickstart Guide

Quickstart guide contains six lower level tasks.

The quickstart guide requires all subcomponents of the tangible and consists of six steps. The user just has to plug in all components and connect them. The quickstart guide contains of structural diagrams. We made stationary pictures of every step and highlighted important parts with red color. We provided also a description for each step, but we tried to remain accurate. It is also placed at the beginning, because this part is usually most frequently used, since the user has built all subcomponents of the tangible once, it is not necessary to built them again and only connect those parts if the user wants to use the tangible.



Figure 3.7: Tangible built with the created manual

3.6.2 Requirement List

The list of requirements is divided into three parts: components, tools and software. We ordered the items in a grid layout and tried to create a look that is not disturbing for the user. Each item is referenced with a picture, its name and the amount needed.

3.6.3 Tangible Case and Shield Instructions

The case and the shield for the tangible structured the same way. At the beginning we have an icon list of the requirements of the instruction part and a time estimation. It might be useful because the user has for example to book a reservation for the FabLab and this might be limited to a specific duration. The first steps are in both cases downloading the

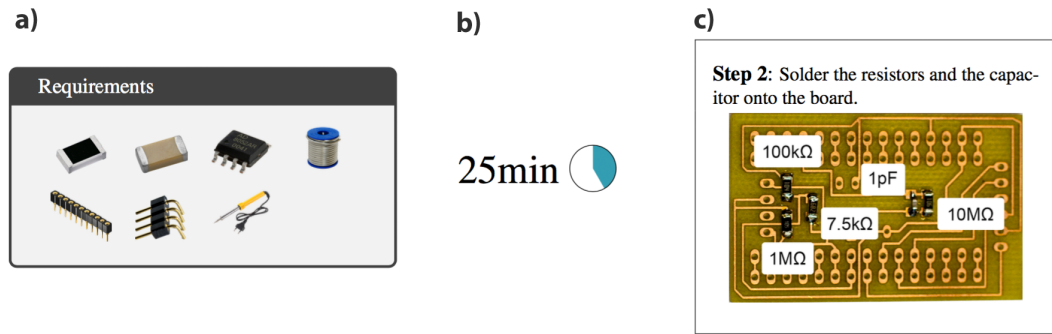


Figure 3.8: Structure of instructions. a) List of requirements. b) Time estimation. c) Structural diagram for step instruction with description.

required files and produce with the laser cutter the parts for the case and with the PCB mill the milled board for the shield.

The tangible case and shield consist of higher level tasks.

The following steps require the user to tinker or to solder components. Here we also make use of structural diagram and provide descriptions for the pictures. Important parts like the orientation are highlighted with red color. Figure 3.8 gives an example how these instructions look like.

3.6.4 Arduino Template

The template is the last chapter of the manual. It contains a template for using the tangible library. Users will probably download the code and will not use this page anymore or at least barely. Therefore, we placed the template at the end of the manual.

Chapter 4

Evaluation

In this chapter we investigate the quality of the manual and encounter improvements. We also measure the time for each building process. The results are based on a small user study.

4.1 Participants

We recruited 10 users (aged 20–25) for our study. 5 have soldering experience and none of them have built an active tangible before.

4.2 Task & Procedure

We asked users to build an active tangible using our manual. We provided a printed version of the manual. Since it is time wasting waiting for the laser cutter and the PCB mill, we prepared those materials. Users had to do multiple tasks. The first task was to build the case of the tangible where they had to glue the parts together and place the copper foil. The next task was to solder the shield. After finishing those parts they were able to use the quickstart

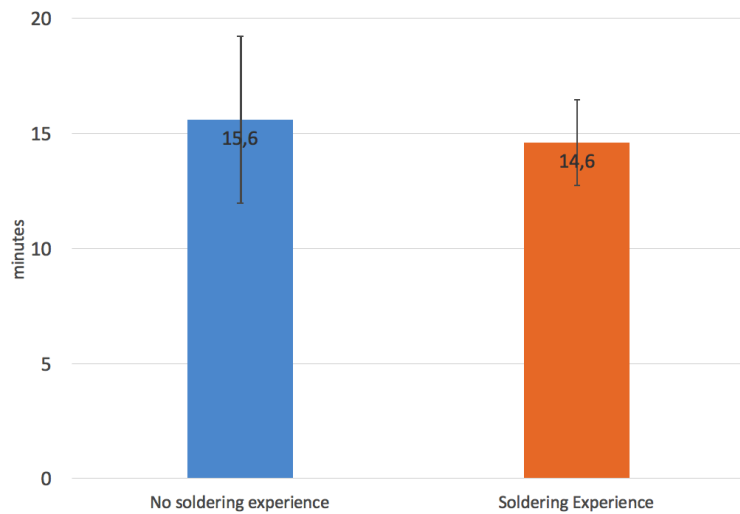


Figure 4.1: Required time to build the case of the tangible.

guide and try the tangible. We provided answers that could possibly come from a FabLab employee.

4.3 Questionnaire

At the end of the study we asked users to fill out a questionnaire. This questionnaire was designed to get some information, if the manual was easy to understand and follow, how users felt during the instruction process and if they are satisfied with their result. These questions were 5 point likert questions and had a scale from totally disagree, to totally agree. At the end there was offered a free text entry box for any additional comments.

4.4 Results

At the end of each task we measured the time, separated from the user group (soldering experience). The time measurements for the case can be seen in figure 4.1. The measurements for the shield are in figure 4.2. 8 out of 10 tangibles

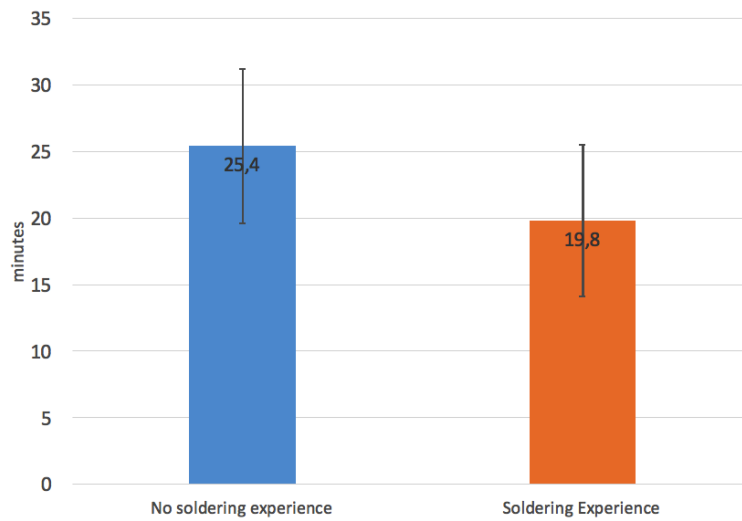


Figure 4.2: Required time to build the tangible shield.

worked at the end. The two tangibles which do not work were both from people with no soldering experience. They soldered both a short circuit. These were the only persons who were not satisfied with the result. All of the participants could follow the manual and said that the manual is very consistent. The results are satisfying because the building time could be reduced to up an hour (building time for those two parts + the time used with the laser cutter and the PCB mill). The only thing all participants with no soldering experience complained was soldering. It was very difficult for them, but 3 out of 5 were able to create a working shield and with the guidance of a FabLab assistant it should be no problem. We used the results an updated the manual. We edited the estimated time and provided extra feedback for parts of the manual where we saw that the users were not really sure what to do next.

Chapter 5

Summary and future work

5.1 Summary and contributions

We created a user friendly tangible construction kit based on the work of Voelker et al. [2015] and the results of the lab course in winter 2015/16. We moved on and transferred the active tangible to the Arduino platform. These tangibles will work on unmodified commercial touch screens.

For the tangible functionality we created a shield with EAGLE for the Arduino Uno and the Arduino Nano. It is easy to adjust these files in order to get a shield for another board. We made it possible to reduce the amount of components used on the shield to six. This makes it much faster to produce and less error prone. The shield also allows stacking other shields on top of it. Therefore, we have to solder another pin header row at each side.

We also provide files for OpenSCAD in order to produce a modular shield. These can be cut with the laser cutter and build within minutes.

In order to make the tangible recognized from the touch screen, we had to make slightly changes in the Multi-

touchkit and write our own communication protocol. It is possible to send other additional data without big problems.

We created a manual which helps students and other users which are interested in building tangibles in a user friendly way. The manual was tested and updated due to the results of the user study.

5.2 Future work

There are many possibilities for future work of this presented approach. The first thing one could do is upgrading the tangible and add different types of sensors/motors to the tangible and try to find the limits.

The Manual could be updated for these purposes. For example how to add a button to a tangible or a specific sensor. Arduino offers a lot of tutorials, but these could be adjusted to work with the Arduino tangible.

The library and the Multitouchkit could also be improved. Currently, there is communication in one way, but it is possible to let the computer send information to the tangible which also enables new possibilities.

Soldering is still a big problem. We could reduce the soldering part to six components, but users still complain about it. Maybe there is a solution that does not require any soldering or a shield for the Arduino will be available in stores.

Appendix A

Arduino Tangible Building Instructions

The following pages contain the building instructions for an active tangible.

Arduino Tangible Building Instructions

Alexander Neumann

September 29, 2016



1

Figure A.1: Manual: Page 1.

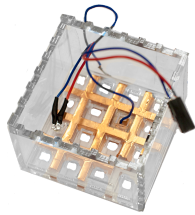
Contents

1 Quickstart Guide	3
2 Requirements	4
2.1 Components	4
2.2 Tools	6
2.3 Software	6
3 Case	7
3.1 Download and Cut	7
3.2 Build	8
4 Shield	9
4.1 Download the circuit	9
4.2 Solder	9
5 Software	11

1 Quickstart Guide

Hint: The side with the slot for the USB connection is the front-side!

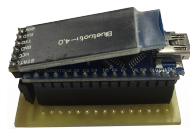
Step 1: Put the smaller grid into your case. Now place the photodiode and the field sensor.



Step 2: Take the battery and place it on the left side.



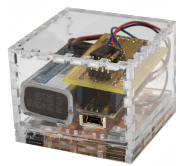
Step 3: Plug the shield into the Arduino. Then plug in the Bluetooth module.



Step 4: Connect the wires with the shield. The photodiode and the field sensor are connected with the outer pins.



Step 5: Place the Arduino such that the USB-port fits to the slot of the case and tape the top plate onto the case.



Step 6: Place the electromagnetic shielding material to your desired positions.

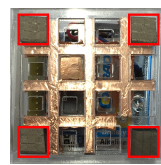






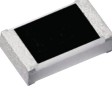
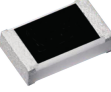
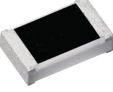
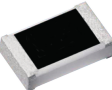




Figure A.3: Manual: Page 3.

2 Requirements

2.1 Components

<p>Arduino Uno/Nano</p>  <p>1x</p>	<p>9V Block Battery</p>  <p>1x</p>	<p>6cm×6cm Copper Foil</p>  <p>1x</p>
<p>HM-10 BLE</p>  <p>1x</p>	<p>Operational amplifier</p>  <p>1x</p>	<p>Photodiode (400nm-1100nm)</p>  <p>1x</p>
<p>7.5kΩ Resistor</p>  <p>1x</p>	<p>100kΩ Resistor</p>  <p>1x</p>	<p>1MΩ Resistor</p>  <p>1x</p>
<p>10MΩ Resistor</p>  <p>1x</p>	<p>1pF Capacitor</p>  <p>1x</p>	<p>Single Sided Copper PCB</p>  <p>1x</p>




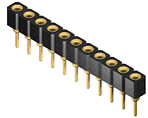
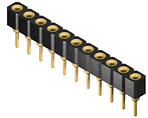


<p>Case material (3mm acrylic glass)</p>  <p>1x</p>	<p>EMS material</p>  <p>1x</p>	<p>Solder</p>  <p>1x</p>
<p>Female pin header</p>  <p>2x 15</p>	<p>Female pin header</p>  <p>1x 6</p>	<p>Male pin header 90</p>  <p>2 & 4</p>
<p>10cm cable</p>  <p>3x</p>		

Figure A.5: Manual: Page 5.

2.2 Tools

<p>PCB mill</p>  <p>1x</p>	<p>Laser cutter</p>  <p>1x</p>	<p>Soldering iron</p>  <p>1x</p>
<p>Cutter knife</p>  <p>1x</p>	<p>Scissor</p>  <p>1x</p>	<p>Glue</p>  <p>1x</p>


2.3 Software



3 Case

3.1 Download and Cut



5min 

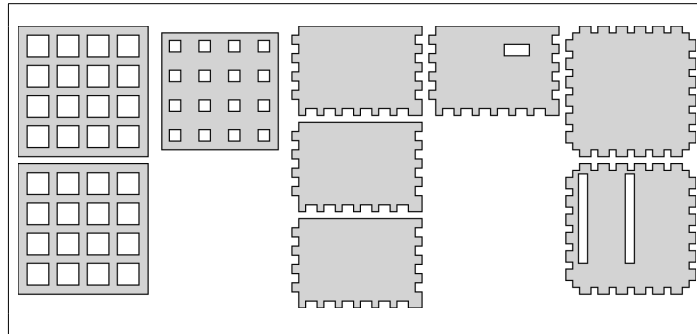
Step 1: Download the files for the case at:

<http://github.com/AlexanderNeumann/Arduino-Tangible/tree/master/Visicut>.

Take the file *case.svg* which contains every part of the case construction. You can use one of the other files if you just want to have a specific part.



Step 2: Use VisiCut and the lasercutter to cut the parts.



3.2 Build

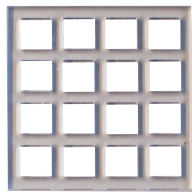
Requirements



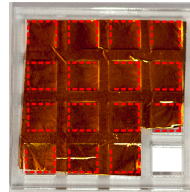
15min 

For this building instructions 3mm acrylic glass was used. If you use other material please take care that the measurements are still the same.

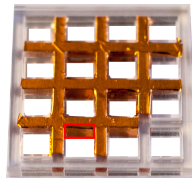
Step 1: Take the two parts of the ground layer and glue them together.



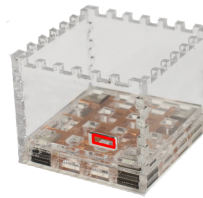
Step 2: Cut out a square of the copper foil and put it on top of the ground layer. Leave one hole of the grid empty.



Step 3: Use the cutter knife to slice the squares of the grid, but leave one side connected. Fold the foil into the grid such that the foil sticks to the walls of the grid.



Step 4: Take the four side parts and glue them together with the ground plate. Take care with the **orientation**.



4 Shield

4.1 Download the circuit



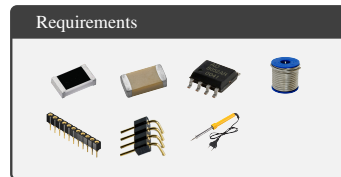
15min

Step 1: Download the job for milling the circuit

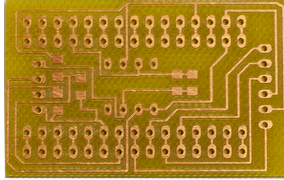
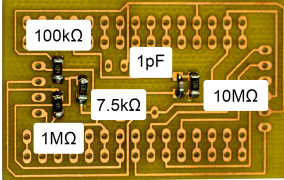
<http://github.com/AlexanderNeumann/Arduino-Tangible/tree/master/BoardMaster/Nano>.

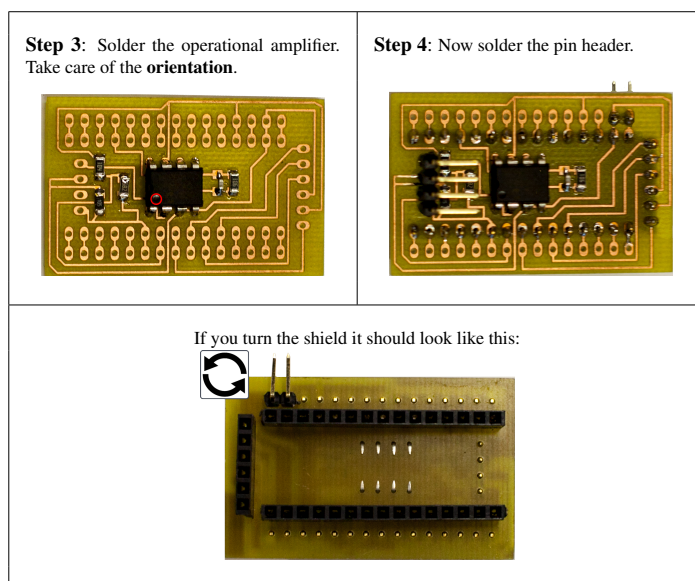


4.2 Solder

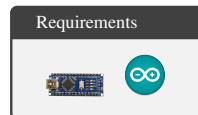


25min

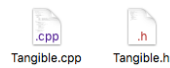
<p>Step 1: Mill the circuit.</p> 	<p>Step 2: Solder the resistors and the capacitor onto the board.</p> 
---	---



5 Software



Step 1: Download the tangible library at:
http://github.com/AlexanderNeumann/Arduino-Tangible/tree/master/tangible_example.



Step 2: Connect the Arduino with your computer.

Step 3: Use the following template and start writing your program.

```
1 #include "Tangible.h"
2 Tangible myTangible(3,2,600); // RX, TX, Photodiode-Threshold
3
4 void setup() {
5 }
6
7 void loop() {
8     myTangible.update();
9 }
10
11 ISR(ANALOG_COMP_vect) {
12     myTangible.checkTable();
13 }
```


Appendix B

User Study Questionnaire

The questionnaire we handed out after the user study.

Informed Consent Form

Evaluating the User Friendly Tangible Construction Kit Manual

PRINCIPAL INVESTIGATOR Alexander Neumann
 Media Computing Group
 RWTH Aachen University
 Phone: 01575-1222792
 Email: alexander.tobias.neumann@rwth-aachen.de

Purpose of the study: The goal of this study is to check the quality of the manual and encounter improvements. Participants will be asked to build a working tangible based on an Arduino. Observation and the questionnaire at the end will be used in the analysis.

Procedure: Participation in this study will involve three phases. In the first phase, you will be asked to build the case of your tangible. You have to glue the individual parts together. In the second phase, you will be asked to solder the shield that will be used for the tangible functionalities. In the last phase you will be asked to put all parts together and test your tangible. This study should take about an hour and a half to complete.

After the study, we will ask you to fill out the questionnaire about the tested manual. In this questionnaire, we will ask some general questions about your procedure and feelings during the building process.

Risks/Discomfort: You may become fatigued during the course of your participation in the study. You will be given several opportunities to rest, and additional breaks are also possible. There are no other risks associated with participation in the study. Should completion of either the task or the questionnaire become distressing to you, it will be terminated immediately.

Benefits: The results of this study will be useful for improving the manual and make it more user friendly.

Alternatives to Participation: Participation in this study is voluntary. You are free to withdraw or discontinue the participation.

Cost and Compensation: Participation in this study will involve no cost to you. There will be snacks and drinks for you during and after the participation.

Confidentiality: All information collected during the study period will be kept strictly confidential. You will be identified through identification numbers. No publications or reports from this project will include identifying information on any participant. If you agree to join this study, please sign your name below.

_____ I have read and understood the information on this form.

_____ I have had the information on this form explained to me.

Participant's Name	Participant's Signature	Date
Principal Investigator		Date

If you have any questions regarding this study, please contact Alexander Neumann at 01575-1222792
 email: alexander.tobias.neumann@rwth-aachen.de

Figure B.1: Consent form

User Study: Tangible Construction Kit

Participant Information

Age: _____

Gender: male
female
N/A

I have soldering experience

Questionnaire

1. The manual was well structured and understandable.

totally disagree totally agree

1 2 3 4 5

2. The building time of the tangible was appropriate.

totally disagree totally agree

1 2 3 4 5

3. The manual was consistent over the different steps.

totally disagree totally agree

1 2 3 4 5

4. I felt confident during the building process.

totally disagree totally agree

1 2 3 4 5

5. Building the case of the tangible was...

very difficult very easy

1 2 3 4 5

6. Soldering the shield was...

very difficult very easy

1 2 3 4 5

Figure B.2: Questionnaire: Page 1.

7. The Quickstart Guide was easy to follow.

totally disagree	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	totally agree
	1	2	3	4	5	

8. How difficult was the whole building process?

very difficult	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	very easy
	1	2	3	4	5	

9. I am satisfied with the result.

totally disagree	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	totally agree
	1	2	3	4	5	

10. Do you have any additional comments?

Figure B.3: Questionnaire: Page 2.

Bibliography

Maneesh Agrawala, Doantam Phan, Julie Heiser, John Haymaker, Jeff Klingner, Pat Hanrahan, and Barbara Tversky. Designing effective step-by-step assembly instructions. *ACM Trans. Graph.*, 22(3):828–837, July 2003. ISSN 0730-0301. doi: 10.1145/882262.882352. URL <http://doi.acm.org/10.1145/882262.882352>.

Rachel Blagojevic and Beryl Plimmer. *CapTUI: Geometric Drawing with Tangibles on a Capacitive Multi-touch Display*, pages 511–528. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013. ISBN 978-3-642-40483-2. doi: 10.1007/978-3-642-40483-2_37. URL http://dx.doi.org/10.1007/978-3-642-40483-2_37.

Liwei Chan, Stefanie Müller, Anne Roudaut, and Patrick Baudisch. Capstones and zebrawidgets: sensing stacks of building blocks, dials and sliders on capacitive touch screens. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems, CHI '12*, pages 2189–2192, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2208276.2208371. URL <http://doi.acm.org/10.1145/2208276.2208371>.

Alessandro D’Ausilio. Arduino: A low-cost multipurpose lab equipment. *Behavior Research Methods*, 44(2):305–313, 2012. ISSN 1554-3528. doi: 10.3758/s13428-011-0163-z. URL <http://dx.doi.org/10.3758/s13428-011-0163-z>.

Elena Driskill and Elaine Cohen. Interactive design, analysis, and illustration of assemblies. In *Proceedings of the 1995 Symposium on Interactive 3D Graphics, I3D '95*, pages 27–34, New York, NY, USA, 1995. ACM. ISBN 0-

89791-736-7. doi: 10.1145/199404.199409. URL <http://doi.acm.org/10.1145/199404.199409>.

George W Fitzmaurice, Hiroshi Ishii, and William AS Buxton. Bricks: laying the foundations for graspable user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 442–449. ACM Press/Addison-Wesley Publishing Co., 1995.

Philip Hodgson.

Hiroshi Ishii and Brygg Ullmer. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, CHI '97*, pages 234–241, New York, NY, USA, 1997. ACM. ISBN 0-89791-802-9. doi: 10.1145/258549.258715. URL <http://doi.acm.org/10.1145/258549.258715>.

H. S. Juang and K. Y. Lurr. Design and control of a two-wheel self-balancing robot using the arduino microcontroller board. In *2013 10th IEEE International Conference on Control and Automation (ICCA)*, pages 634–639, June 2013. doi: 10.1109/ICCA.2013.6565146.

Rene Linden. Multitouchkit: A software framework for touch input and tangibles on tabletops and. Master's thesis, RWTH Aachen University, Aachen, September 2015.

J. Sarik and I. Kymissis. Lab kits using the arduino prototyping platform. In *2010 IEEE Frontiers in Education Conference (FIE)*, pages T3C-1–T3C-5, Oct 2010. doi: 10.1109/FIE.2010.5673417.

Thomas W. Schubert, Alessandro D'Ausilio, and Rosario Canto. Using arduino microcontroller boards to measure response latencies. *Behavior Research Methods*, 45(4):1332–1346, 2013. ISSN 1554-3528. doi: 10.3758/s13428-013-0336-z. URL <http://dx.doi.org/10.3758/s13428-013-0336-z>.

Simon Voelker, Kosuke Nakajima, Christian Thoresen, Yuichi Itoh, Kjell Ivar Øvergård, and Jan Borchers. Pucs: Detecting transparent, passive untouched capacitive widgets on unmodified multi-touch displays. In *ITS '13: Proceedings of the ACM International Conference*

on *Interactive Tabletops and Surfaces*, pages 101–104, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2271-3. doi: 10.1145/2508468.2514926.

Simon Voelker, Christian Cherek, Jan Thar, Thorsten Karer, Christian Thoresen, Kjell Ivar Øvergård, and Jan Borchers. Percs: Persistently trackable tangibles on capacitive multi-touch displays. In *UIST '15: Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*, UIST '15, pages 351–356, New York, NY, USA, November 2015. ACM. doi: 10.1145/2807442.2807466. URL <http://dx.doi.org/10.1145/2807442.2807466>.

Alexander Wiethoff, Hanna Schneider, Michael Rohs, Andreas Butz, and Saul Greenberg. Sketch-a-tui: Low cost prototyping of tangible interactions using cardboard and conductive ink. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction*, TEI '12, pages 309–312, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1174-8. doi: 10.1145/2148131.2148196. URL <http://doi.acm.org/10.1145/2148131.2148196>.

Neng-Hao Yu, Li-Wei Chan, Seng Yong Lau, Sung-Sheng Tsai, I-Chun Hsiao, Dian-Je Tsai, Fang-I Hsiao, Lung-Pan Cheng, Mike Chen, Polly Huang, and Yi-Ping Hung. Tuic: Enabling tangible interaction on capacitive multi-touch displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 2995–3004, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9. doi: 10.1145/1978942.1979386. URL <http://doi.acm.org/10.1145/1978942.1979386>.

Index

3D printer, 24

abbrv, *see* abbreviation

action diagram, 12

analog comparator, 21

Arduino, 9–10, 17

assembly instructions, 11

AT commands, 23

ATmega328P, 17

ATmega32U4, 17

BLE, 22

Bluetooth Communication, 22

board design, 17

Bricks, 5

CapStones, 6

CapTUI, 7

cardboard, 7

Communication Protocol, 23

evaluation, 29–31

exploded view), 11

eyes free interaction, 1

FabLab, 2, 24, 27, 29

Field Sensor, 21

frequency tag, 5

future work, 34

haptic feedback, 1

HM-10, 22

laser cutter, 24

Lego, 11

Light Sensor, 21

manual, 25

Manual guidelines, 11–13

MSP430G2553, 17
Multitouchkit, 9, 22

OpenSCAD, 24

PCB mill, 27, 29
PERCs, 1, 8
photodiode, 21
PUCs, 1, 7

questionnaire, 30
Quickstart Guide, 26

RX, 20

Schematics, 19
Sketch-a-TUI, 7
SoftwareSerial, 22
structural diagram, 12, 26
SVG, 24

Tangible Bits, 5
Tangible Case, 24
Tangible Library, 20–23
Tangible Shield, 18
tower defense, 15
TUIC, 5
TUIO, 9
TX, 20

User Friendly Manual, 25
user study, 29

ZebraWidgets, 6

