originalMachines:

Developing Tools and Methods for Object-Oriented Mechatronics

Peter Schmitt

MS, Media Arts & Sciences, MIT, 2007
Akademiebrief, Kunstakademie Düsseldorf, 2005
Meisterschüler, Kunstakademie Düsseldorf, 2003

Submitted to the Program of Media Arts and Sciences, School of Architecture and Planning in partial fulfillment of the requirement for the degree of Doctor of Philosophy at the Massachusetts Institute of Technology, June 2011

Author
Program in Media Arts and Sciences
April 2011

Certified by
Cynthia Breazeal
Associate Professor of Media Arts and Sciences
Massachusetts Institute of Technology

Accepted by
Mitchel Resnick
LEGO Papert Professor of Learning Research
Academic Head, Program in Media Arts and Sciences

originalMachines:

Developing Tools and Methods for Object-Oriented Mechatronics

Peter Schmitt

MS, Media Arts & Sciences, MIT, 2007
Akademiebrief, Kunstakademie Düsseldorf, 2005
Meisterschüler, Kunstakademie Düsseldorf, 2003

Submitted to the Program of Media Arts and Sciences, School of Architecture and Planning in partial fulfillment of the requirement for the degree of Doctor of Philosophy at the Massachusetts Institute of Technology, June 2011

Abstract

The digital revolution has fundamentally changed our lives by giving us new ways to express ourselves through digital media. For example, accessible multimedia content creation tools allow people to instantiate their ideas and share them easily. However, most of these outcomes only exist on-screen and online. Despite the growing accessibility of digital design and fabrication tools the physical world and everyday objects surrounding us have been largely excluded from a parallel explosion of possibilities to express ourselves. Increasingly, web-based services allow professional and non-professional audiences to access computer-aided manufacturing (CAM) tools like 3D-printing and laser-cutting. Nonetheless, there are few (if any) design tools and methods for creating complex mechanical assemblies that take full advantage of CAM systems. Creating unique mechatronic artifacts or "originalMachines" requires more specific and sophisticated design tools than exist today. "Object-Oriented Mechatronics" is a parametric design approach that connects knowledge about mechanical assemblies and electronics with the requirements of digital manufacturing processes. Parametric instances like gears, bearing and servos are made available as objects within a CAD environment which can then be implemented into specific projects. The approach addresses the missing link between accessible rapid-manufacturing services and currently available design tools thereby creating new opportunities for self-expression through mechatronic objects and machines.

The dissertation matches mechanical components and assemblies with rapid manufacturing methods by exploring transferability of conventional manufacturing techniques to appropriate rapid manufacturing tools. I rebuild various gearing and bearing principles like four-contact point bearings, cross roller bearings, spur and helical gears, planetary gears, cycloidal and harmonic gear reducers using the laser cutter, the CNC-mill and the 3D-printer. These explorations lead to more complex assemblies such as the PlywoodServo, 3DprintedClock and 3-DoF (Degree of Freedom) Head. The lessons from these explorations are summarized in a detailed "cook book" of novel mechatronic assemblies enabled by new fabrication tools. Furthermore, I use the results to develop a CAD tool that brings together several existing software packages and plug-ins including Rhino, Grasshopper and the Firefly experiments for Arduino, which will allow animation, fabrication and control of original machines. The tool is an example of an object-oriented design approach to mechatronic assemblies. A user calls a DoF (Degree of Freedom) object (parametric servo) with specific parameters like gearing and bearing types, motor options and control and communication capabilities. The DoF object then creates the corresponding geometry which can be connected and integrated with other actuators and forms. A group of roboticists and designers participated in a workshop to test the tool and make proposals for original machines using the tool.

The dissertation has contributions on multiple levels. First, the actuator assembly examples and parametric design tool present a body of novel work that illustrates the benefits of going beyond off-the-shelf actuator assemblies and kit-of-parts for robotic objects. Second, this tool and the accompanying examples enable the design of more original machines with custom actuator assemblies using the latest digital fabrication tools. Finally, these explorations illustrate how new CAD/CAM tools can facilitate an exchange between more design-oriented users and more engineering-oriented users.

Thesis Supervisor: Associate Professor of Media Arts and Sciences, Cynthia Breazeal, Massachusetts Institute of Technology

originalMachines:

Developing Tools and Methods for Object-Oriented Mechatronics

Peter Schmitt

Leah Buechley
Assistant Professor of Media Arts and Sciences
Massachusetts Institute of Technology

originalMachines:

Developing Tools and Methods for Object-Oriented Mechatronics

Peter Schmitt

Axel Kilian
Assistant Professor, School of Architecture
Princeton University

originalMachines:

Developing Tools and Methods for Object-Oriented Mechatronics

Peter Schmitt

Chris Bangle
Chris Bangle Associates, S.R.l.

Acknowledgments

No man is an island! For all of your support, friendship and warmth I would like to thank:

Susanne Seitinger

Edith Ackerman, Zoe Adamovicz, Paula Aguilera, Jason Alonso, Stephanie Anderson, Paula Anzer, Ed Baffi, Catherine Bangle, Chris Bangle, Derek Bangle, Ute Meta Bauer, Anne Beamish, Andreas Bee, Matt Berlin, Leonardo Bonanni, Joost Bonsen, Marcel Botha, Adam Boulanger, Cynthia Breazeal, Ruth Breitenstein, Karen Brennan, Zoz Brooks, Leah Buechley, Falko Burkhardt, Helmar Burkhardt, Thorsten Burkhardt, Leah Anne Cappello, David Carr, Donna Catelldo, Erni Catelldo, Angela Chang, Mano Chappelier, Wayne Cherry, Kenny Cheung, Ryan Chin, Marcello Coelho, James Coleman, Tony Cragg, Kevin Davis, Bruce Deffenbaugh, Anthony DeVincenzi, Alex Diensberg, John DiFrancesco, Laura Dingwall, Kelly Dobson, Kristopher Dos Santos, Richard Dunn, Mark Feldmeier, John Ferguson, Jeannie Finks, Sean Follmer, Christian Forsen, Chad Foster, Elise Foster, Natalie Freed, Joerg Gattwinkel, Stephanie Gayle, David Gerber, Neil Gershenfeld, Will Glesnes, Max Gordon, Rab Gordon, Tanja Gordon, Dirk Gottwald, Tom Greene, Jesse Grey, Polly Guggenheim, John Helmes, Gamaliel Herrera, Wayne Higgens, Guy Hoffman, Garett Hwang, Hiroshi Ishi, Mitch Joachim, Dale Joachim, Markus Kaiser, Corry Kidd, Axel Kilian, Laura Kim, Cornelle King, Tony Knopp, Patrik Kuenzler, Franklin Labarbara, Jean-Babtiste Labrune, Berthold Langnickel, Vanessa Lark, Will Lark, Sherry Lassiter, Jin Joo Lee, Hyejin Lee, Daniel Leithinger, Phil Liang, Jeff Lieberman, Lisa Lieberson, Ulrich Lorang, Oliver Lutz, Tom Lutz, Pattie Maes, Fiorella Marengo, John McBean, David Mellis, Bill Mitchell, Frank Moss, Julian Hernandez Munoz, Lira Nikolovska, Laura Noren, Sigurour Orn, Neri Oxman, Nick de Palma, Andy Payne, Nicholas Penycooke, Hanna Perner-Willson, Linda Peterson, Arthur Petron, Peter Pflanz, Daniel Piker, Retro Poblano, Mitch Resnik, Adam Rinke, Andera Rinke, Klaus

Rinke, Phillip Robbel, David Robert, Tamar Rucham, Thomas Ruff, Sajid Sadi, Phil Salesses, Larry Sass, Katja Schechter, Susanne Schindler, Alfons Schmitt, Daniel Schmitt, Helga Schmitt, Tobias Schmitt, Doro Schoenenstein, Max Schulze, Mark Schwindenhammer, Heinz Seitinger, Karin Seitinger, Adam Setapen, Martin Seymour, Mikey Siegel, Stacie Slotnick, Aaron Solle, Stefan Sous, Dan Stiehl, Praveen Subramani, Amy Sun, Bob Swartz, Jazec Szubert, Alex Taylor, Gottfired Teil, Richard The, Andrea Thomaz, Skylar Tibits, Matt Trimble, Isabella Tsao, Greg Tucker, Franco Vairani, Jani Vanter, Cati Vaucelle, Noah Vawter, Nick Villar, Ryan Walsh, Adam Whiton, Bettina Wickert, Cynthia Wilkes, Kenton Williams, Ryan Wistort, Jane Wojcik, Jane Wolfson, Jamie Zieglbaum, Amit Zoran

Contents

Chapter 1

Introduction

The digital revolution has fundamentally changed our lives by giving us new ways to express ourselves through digital media. For example, accessible multimedia content creation tools allow people to instantiate their ideas and share them easily. However, most of these outcomes only exist on-screen and online. Despite the growing accessibility of digital design and fabrication tools the physical world and everyday objects surrounding us have been largely excluded from a parallel explosion of possibilities to express ourselves. Increasingly, web-based services allow professional and non-professional audiences to access computer-aided manufacturing (CAM) tools like 3D-printing and laser-cutting. Nonetheless, there are few (if any) design tools and methods for creating complex mechanical assemblies that take full advantage of CAM systems. Creating unique mechatronic artifacts or "originalMachines" requires more specific and sophisticated design tools than exist today. "Object-Oriented Mechatronics" is a parametric design approach that connects knowledge about mechanical assemblies and electronics with the requirements of digital manufacturing processes. Parametric instances like gears, bearing and servos are made available as objects within a CAD environment which can then be implemented into specific projects. The approach addresses the missing link between accessible rapid-manufacturing services and currently available design tools thereby creating new opportunities for self-expression through mechatronic objects and machines.

Motivation: From where do robots come?

We are increasingly surrounded by kinetic objects that have embedded electronics which allow them to move and interact with people and their surroundings. Tofu and TofuMini are two examples of companion robots from the MIT Media Lab's Personal Robots group. They are built with a combination of off-the-shelf parts and custom-designed mechanical components and materials. In general, there is a spectrum for the origin of robots from completely off-the-shelf robots through kits-of-parts to fully customized examples. Examples and descriptions of each approach follow.

Figure 1.1 (Right) Tofu and TofuMini, MIT Media Lab, Personal Robots Group, vimeo.com/6409030

Approach 1: Ready-made Robots

The ready-made approach refers to using complete robots that are used in the exact configuration in which they were built. For example, the now defunct Sony AIBO robotic dog or Robosapiens do not require the user to add any hardware features or create new code (though they can be used as research platforms in which case there would be some development work). There are also sophisticated, custom-design robots that I also consider ready-made because they are not being built by the person commissioning them. In other words an expert robot builder creates the machine and another research specialist then develops software and usage scenarios for it. In this approach, users have no power al all to intervene in the object's shape, form, design, or mechanical functions.

Figure 1.2 (Left) Sony Aibo and Robosapiens

Approach 2: Kits-of-Parts

The kits-of-parts approach enables user to assemble machines from prefabricated components like LEGO bricks. The Bioloid Robotics Kit is a prim example. In this approach, users have slightly more control over the appearance and functionality of the machine within the constraints of the modular system. Still, they are bound by the particular design choices made by the creators of the kit potentially limiting the diversity of possible outcomes and biasing users towards certain kinds of creations.

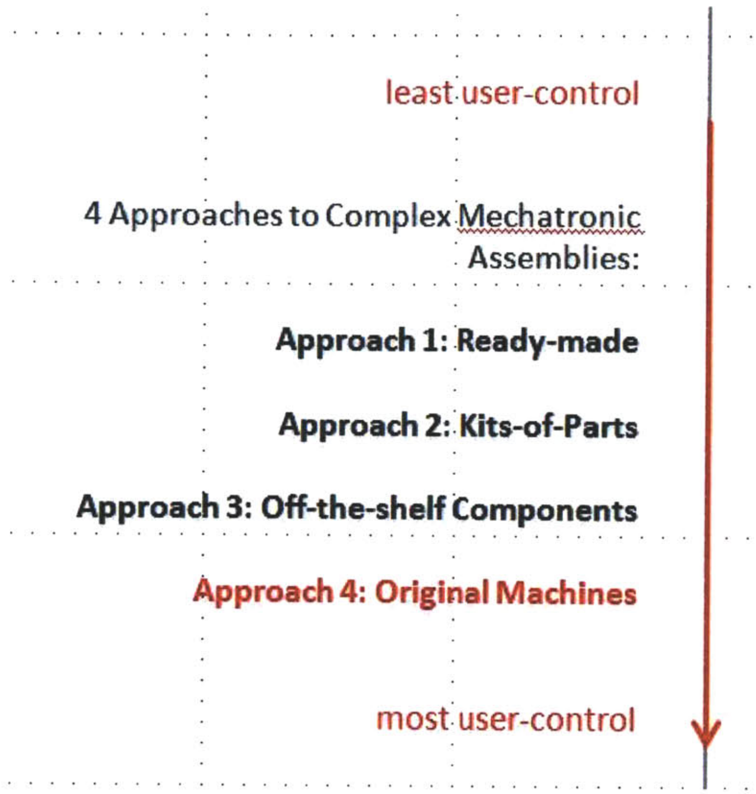
Figure 1.3 (Above) Bioloid robots and dynamixel, Source: robotis.com

Approach 3: Off-the-Shelf Components

In the third approach (e.g. Tofu and TofuMini above or Domo from MIT CSAIL), users create their own customized robot from low-level off-the shelf components such as RC servos. Users have significant control over the appearance and functionality of the machine depending on their skill-level and the application domain in question. Many of the examples in this domain are extremely expressive and particular. Kinetic creations of all kinds would fall under this category. However, users are still limited to the need for certain basic parts that might constrain specific features of their creation.

These three approaches do not provide enough flexibility and freedom to truly design and animate kinetic artifacts from the ground up. Alternatives exist such as Perner-Wilson's Kit-of-No-Parts approach where she guides users through crafting their own fully customize electronics. Still, most existing examples of kinetic objects fall under one of the three approaches described above.

Figure 1.4 (Right) Examples of user-created kinetic sculptures, objects and robots. Source: Letsmakerobots.com



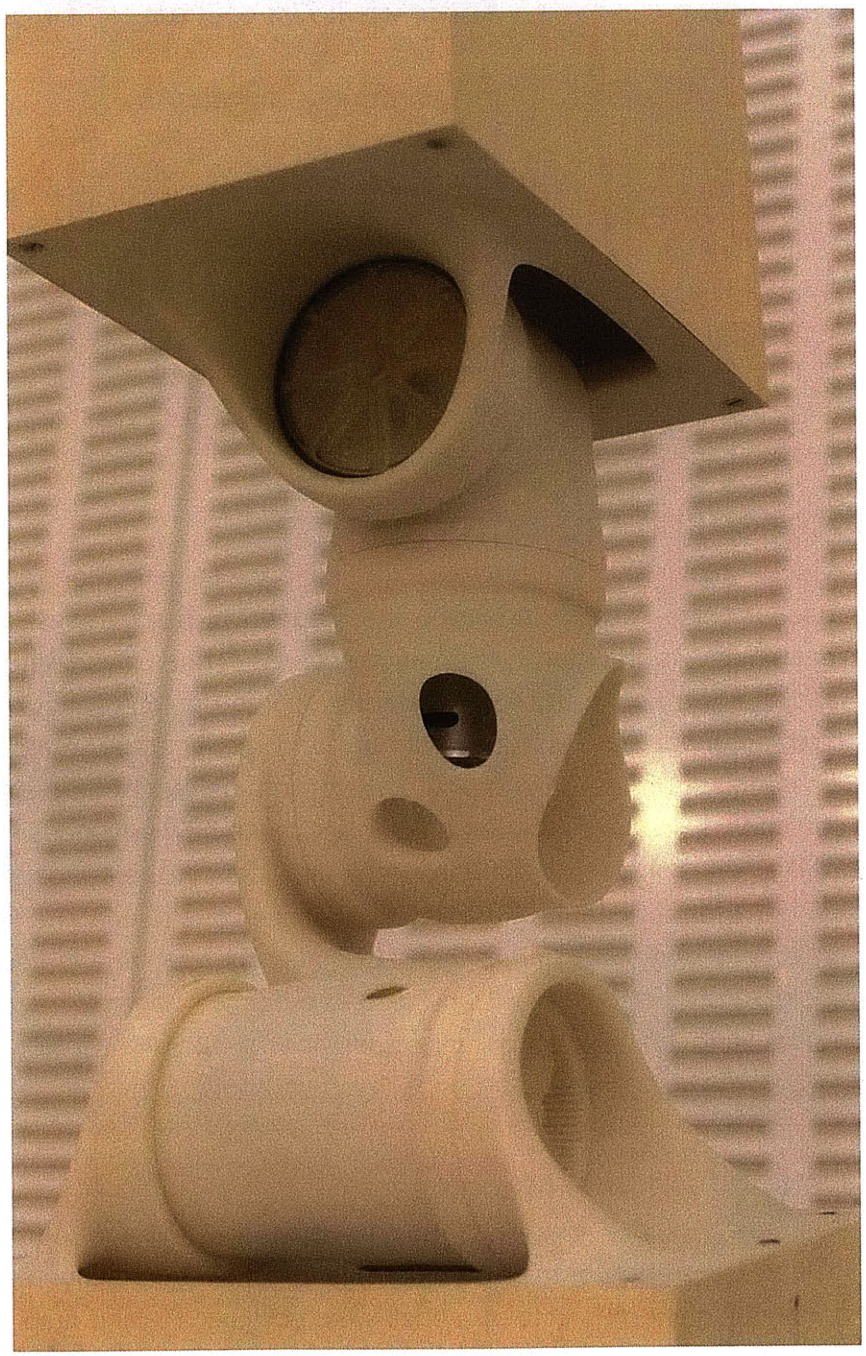
Research Question

originalMachines is an interdisciplinary investigation of mechatronics, which underpins the creation of many types of machines. Rapid manufacturing processes like 3D-printing, laser cutting and CNC milling are investigated with respect to their potential for developing servos and power transmission elements such as gears and gear-boxes as near-complete assemblies. In this thesis the question is asked how advances in CAD/CAM processes will enable the integration of the concept, design and fabrication of mechatronic assemblies to enable the creation of originalMachines within domains such as the arts and design. Furthermore, how can these approaches become more available to an interdisciplinary audience of expert and non-expert users of all ages?

Defining Original Machines

Whereas typically machines consist of interconnected parts lacking integration of their components and a holistic design an originalMachine consists of a shell which organically and fluidly blends and integrates actuators, bearings and in some cases sensors and other components. These original machines can be conceptualized and designed using a parametric actuator tool built into a CAD/CAM environment and made including all mechanical components necessary for the servos and bearings using rapid manufacturing especially 3D printing. Similar to programming and coding strategies the build in tool represents a knowledgebase comparable to a library enabling users to integrate mechanics without having to develop them first. Also

Figure 1.5 (Right) Idling, an originalMachine example, see chapter 5 for detailed description.



like object-oriented programming each mechanism can be called upon with specific parameters and will then build itself within the user's project. The geometric representation of the mechanism and actuators can be modified by the user in order to design the desired original machine which will then be made using 3D printing. By calling for tools to create original machines, this thesis highlights how many underexplored design opportunities exist for increasing the diversity of machines made.

The term "originalMachines" serves as an overarching category for a broad range of mechatronic objects. The artifacts targeted by this research are related in size, form, shape and complexity to the following: domains such as toys, consumer electronics, RC and robot servos like Dynamixel, robotic kits like Bioloid and Lego Mind Storms; and applications such as animatronics, academic research in computer science and robotics, product design process and end-user applications like DIY, tinkering, art, play and learning.

While robots are strongly associated with human companions and workers most likely in humanoid form, originalMachines describe any actuated artifact involving sensing, actuation and human interaction not covered under the robot paradigm. Examples for originalMachines would be a coffee mug which turns its handle towards the person reaching for it (Bangle, every day magic concept for kitchen and household); or a fruit bowl that changes its shape according to the number of items it is supposed to hold (Schmitt 2010). As these examples show, new kinds of objects are specialized for certain, fairly simple interaction, which makes them a particularly interesting field for exploring original machines. The goal is not to make the actuators invisible but to celebrate them an integral part of the objects for which they were designed.

A summary list of the qualities of originalMachines is as follows:

- **parametric**: shareable, encoded knowledge in a set of design constraints, use of digital fabrication, integrated form (overall shape) and mechanical components (bearings, gear reducers), project specific component creation

- **integrated**: electronics integrated into parametric definition of mechanical elements, no seams between components, all components 3D-printed, fewer components in total, easy of assembly or no assembly
- **one-of-a-kind**: expressive qualities of motion, new kinds of mechanical assemblies and integrated parts, no mass manufacturing, constrains beyond function

Building Blocks: The Servomechanism

The core building block or DNA of every robot or mechatronic artifact consists of servos in different shapes and sizes. Because of its central importance it is not surprising that there are many off-the-shelf versions of the servomechanism. However, in this dissertation the goal is to break open this black box in many different ways and demonstrate how it can be integrated more fluidly with the overal objects.

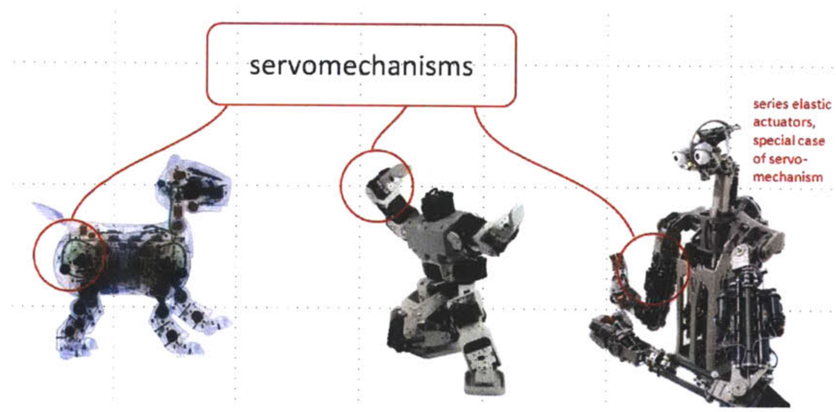


Figure 1.6 (Right) Diagram showing the location of the servo in three different robots. From left to right: Sony Aibo, Robosapiens, Domo (MIT CSAIL).

Methods

Object-Oriented Design Tool

A dominant, existing approach in the domains listed above is the kit-of-parts approach. For example, all the parts needed for a robotic project arrive presorted and fully compatible with each other. All mechanical hardware and electronics are pre-made and guaranteed to be compatible with each other. The system connects to a computer where behavioral programming can be done with the help of specialized software like visual programming language.

As an alternative to the kit-of-parts approach, creators have the option of selecting their own off-the-shelf components and integrating them. Even though these parts are designed to connect, the user will be forced to create custom elements like connection brackets between servos. Most users in this category will already be using CAD/CAM tools for design and fabrication.

In this dissertation a proposal for another approach is made, which tries to combine the benefits of a parametric tool with an open-ended design process. It allows the user to gain control over all aspects of the object including not only its shape, form, size, and material properties, but also its functionality, control, and animation. It uses the computer and computational tools holistically to integrate design, fabrication and controls overcoming the limitations of both the kit-of-parts approach and the free-form process.

Addressing New Communities

In this thesis a broad user base is addressed with a common interest in the types of machines and robots described above as well as their applications, control and making. Expert and non-expert users like engineers, designers, architects, artists, DIY, and tinkerers of all ages are addresses. These users engage with robotic kits, animatronics, research and product development involving actuation and motion enabled projects, actuated toys and playful learning, but they seek to go beyond the constraints of off-the-shelf products.

Existing examples such as the LilyPad Arduino (Buechley & Hill 2010) have demonstrated the potential for technologies with different affordances to attract new communities of users. The LilyPad is an adaptation for wearable computing of the popular embedded electronics platform Arduino. Even though it technically provides the same capabilities it is intended for applications that involve fabrics and conductive thread. This simple change has enabled a host of new creations created with the platform that may not have been built with the generic Arduino platform. The reason is connected to the affordances of the system. However, most importantly the new creations are the result of previously unengaged users becoming involved in a new medium. They bring their skills from quilting, sewing, weaving and so on and combine it with the capabilities of embedded electronics to generate novel and meaningul (to them especially) projects. Similarly, the originalMachines tool and examples are intended to inspire existing and new makers to become involved with kinetic and actuated artifacts.

Structure of the Dissertation

In the first part of the research, I match mechanical components and assemblies with rapid manufacturing methods by exploring transferability of conventional manufacturing techniques to appropriate rapid manufacturing tools. Ongoing explorations such as the PlywoodServo, 3DprintedClock and 3DprintedGearBoxes will inform the structure of this typology and allow me to specify constraints and parameters.

In the second part of the research, I enclode this knowledge in a novel parametric design and fabrication tool. This tool will not just allow for parametric actuator design but rather target assemblies and objects as a whole. It should not only allow the user to specify details about each piece of a mechanism but also the form, shape and geometry connecting them. It should also allow animation and control of robots or other original machines. The tool will consist of an integrated environment that brings together several existing software packages and plugins including Rhino, Grasshopper and the Firefly experiments.

The dissertation is divided into six chapters that are roughly divided into the two parts described above. Chapter 2 provides both the inspirational context for originalMachines - my own work as an artist - as well as examples from the diverse field of digital design and fabrication. Chapter 3 distills the lessons from a series of design explorations into a cook-book for using the latest CAD/CAM machines to create mechatronic components, near-complete assemblies, and assemblies. Chapter 4 describes the parametric design tool that is grounded in the lessons from Chapter 3. Chapter 5 summarizes the design examples various users and myself created using the parametric design. Finally, Chapter 6 summarizes the contributions of the dissertation and provides an outlook on future work.

Chapter 2

Background and Motivation

This chapter provides the background and motivation for creating originalMachines and developing a design tool for object-oriented mechatronic. The motivation for this work comes from two primary areas. First, the arts, design and architecture provide the driving force behind creating holistic mechatronic artifacts. Second, the ongoing revolution in digital fabrication makes it possible to reconsider how kinetic and mechatronic artifacts are designed, manufactured and used. The chapter is loosely divided into two sections according to these two themes.

Personal Motivations for originalMachines

As an artist trained at the Academy of Fine Arts in Düsseldorf, Germany prior to coming to MIT I focused on kinetic sculptures and installations that draw people in through their unique motion. In order to create the effects I desired, I had to develop unique mechanisms and electronics, different from anything available off-the-shelf. The work led me to undertake many different investigations, especially a series of pieces inspired by the movements of plants in nature.

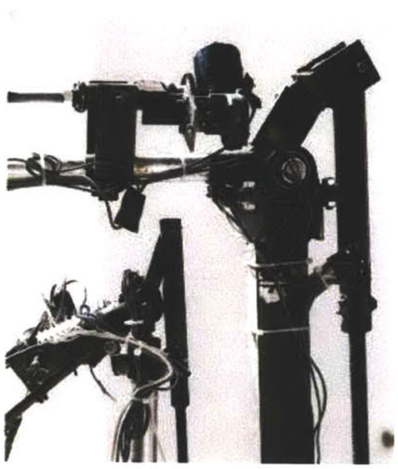


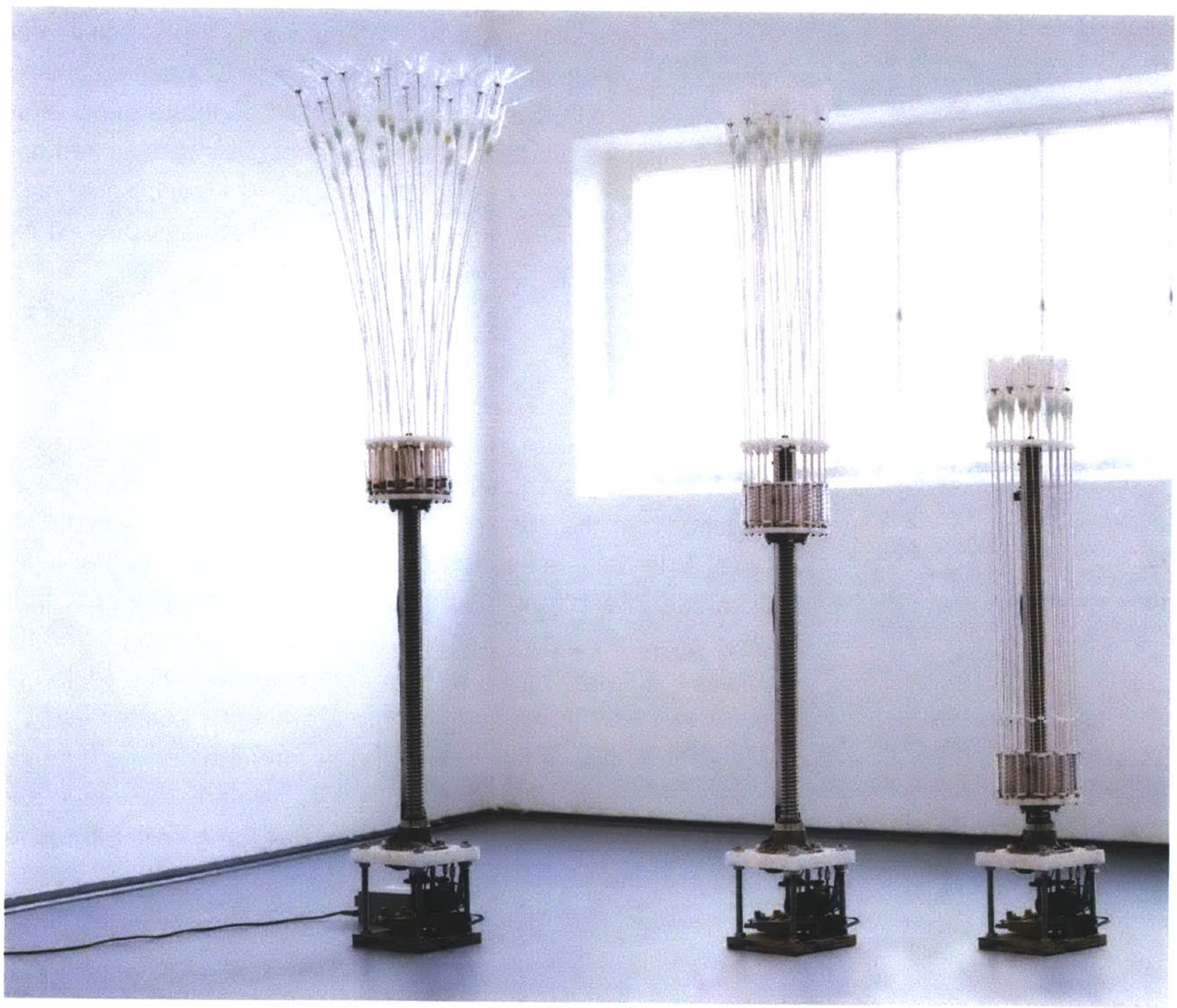
Figure 2.7 (Top) 001#00 Photo montage showing the sculpture in its different stages
Figure 2.8 (Above) 001#00 Detail of primary folding mechanism made from industrial bearings, bicycle bearings, windshield wiper motoers, threaded rods, wires and end switches.

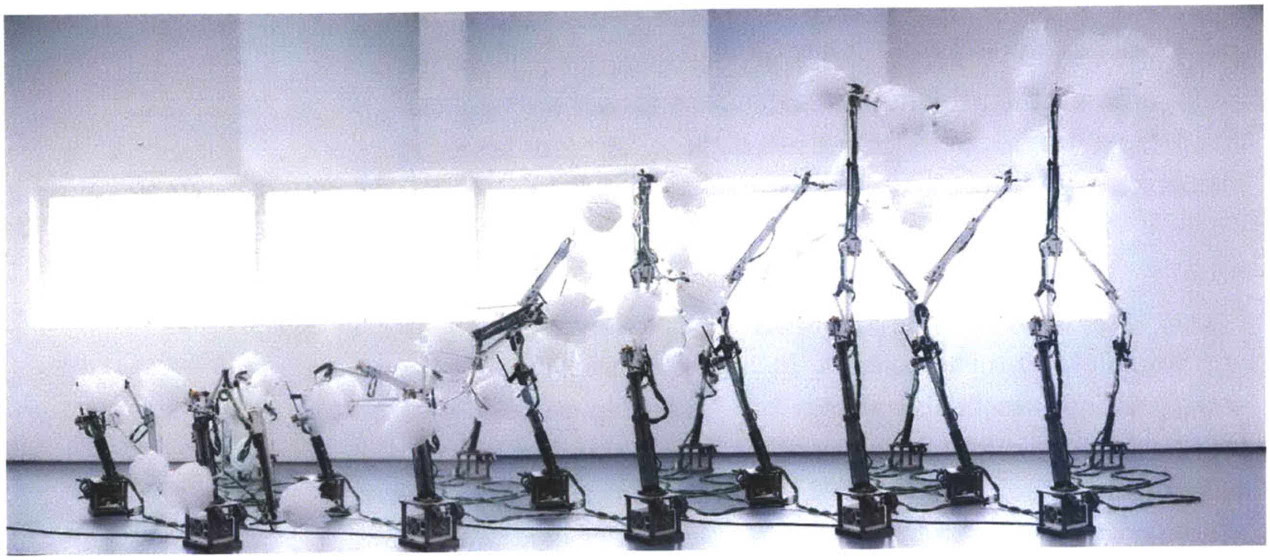
001#00 (Fig. 2.7 and 2.8) is a 4.8-meter sculpture that gradually folds and unfolds like ferns in nature. The piece consists of four joints topped with a light-weight unfolding umbrella in three segments. It is powered by windshield-wiper motors driving threaded rods acting as linear actuators. A winch mechanism in combination with springs unfolds the umbrellas. The full sequence of motion (from closed to open and back to closed) runs for 40 minutes. All the parts were assembled from found pieces such as car and bicycle parts repurposed to create something completely new.

Another piece, 002#01 (Fig. 2.9), similarly has different stages of movement. 24 blossoms mounted on spring-steel rods which are individually mounted on a plate move up and down a linear actuator to a maximum height of two meters. The very last inches of motion towards the top hold back the enclosures of the spring-steel rods which causes the blossom to open. The blossoms are custom-made cast epoxy resin in plastic film with hand-wound rotary springs. The motor sits in the base of the piece and weights the sculpture. The drive electronics pause the motor at the top and bottom of each cycle for several minutes. A full cycle (from lowered to raised back to lowered) takes 20 minutes.

003#02 (Fig. 2.11 and 2.12) is the first ensemble of machines that I built. There are three orchid-like robots that open and close in approximately 30-minute cycles. The orchids combine the linear and rotational mechanisms from the preceding pieces. They turn and unfold at the same time. Each sculpture has its own personality through motion as each one moves slightly differently according to the mechanical friction in its components (which were also salvaged). At 2.5 meters height, the objects fill a room up to the ceiling.

Figure 2.9 001#02 Photo montage showing the sculpture in its different stages in motion





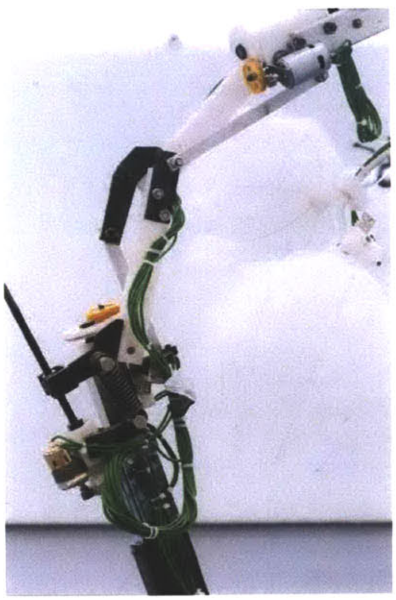


Figure 2.10 (TOP) 003#02 Three kinetic scultpures imitating orchids in motion.
Figure 2.11 (Above) 003#02 Detail showing wiring, gears, motors, spring, joints and structure of the machine.

In each of the examples described above, I was motivated to create my own custom mechanisms by a mix of aesthetic, technical and social concerns. Without access to sophisticated prototyping technologies, I was forced to use found mechanisms and off-the-shelf parts in creative ways to generate the desired effects. Still, the available parts and resources were limited and did not allow me to create fully integrated kinetic sculptures. The desire to go beyond existing components has pushed me towards integrating CAD/CAM into my artistic practice.

Experience and Materiality

Artists and makers of all kinds have long juxtaposed the imagery with the materiality of their creations. For example, Dieter Krieg's painting from the "Spiegelei" series (German for: fried egg or sunny-side-up) is an enormous painting (7'6 by 16'3) that looks like a naturalistic rendering of an egg from a distance, but from close proximity the viewer only sees paint. The material on the canvas – pigment mixed with an acrylic binder – has sculptural properties and is applied roughly like a thick plaster using a trowel rather than a brush. The piece's strength derives from the tension between the material qualities, painterly technique and imagery only visible at a distance. (www.stiftung-dieter-krieg.de/index.php?menuid=43andreporeid=53)

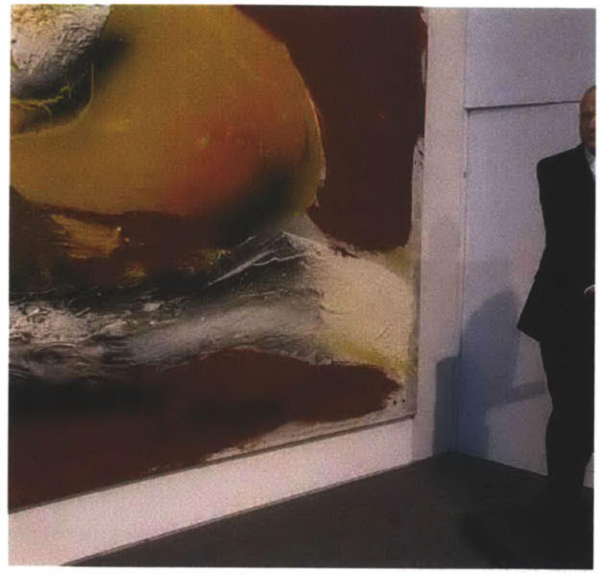


Figure 2.12 (Right) Dieter Krieg, ohne Titel, 1999, Acrylic paint on canvas. Source: http://www.rottweil.de/ceasy/modules/cms/main.php5?cPageId=1983&view=publish&item=article&id=501

Figure 2.13 (Left) Dieter Krieg, ohne Titel, 1999, Acrylic paint on canvas. Source: http://www.monopol-magazin.de/kalender/ort/2010550/duesseldorf/Galerie-Wolfgang-Gmyrek.html

This dissertation argues that a similar tension exists between the kinetic, mechatronic experience of machines and their emotional, evocative characteristics. Rather than designing hidden mechanical devices, I focus on the types of objects that celebrate mechanics, materials and behaviors holistically. Rather than thinking about machines as something to be operated, this view turns machines into something to be enjoyed along with other objects in our environment. Digital fabrication will enable a more wide-spread use of actuators in design creations which will increase the number of mechatronic artifacts for everyday interactions and experience. These objects will require novel actuator assemblies that unite material affordances with movement and behavior.

Challenges of Digital Fabrication

Supporting actuated motion in technological objects increases complexity and cost significantly compared to static objects because they require many different electronic and mechanical components as well as interfaces, software and algorithms. The diversity and multiplicity of these components leads to complex system architectures. A common approach has been to create discrete components and componentized assemblies available off-the-shelf. This approach leads to a high barrier to entry and a strong set of constraints:

- Physical constraints: form factor, dimensions, proportions, attachment points, interfacing surfaces, electrical connections location and connector type, weight, noise level, thermal exchange surfaces.

- Functional constraints: actuation force, actuation time, actuation reach or envelope, actuation precision, the algorithmic implementation of the actuation function, interfacing protocol, communication modes.

While there are benefits to Lego-brick approaches the constraints on new creations are considerable. Even a very large number of discrete components will limit the possibilities for designers, especially the aesthetic appearance of new original machines.

As the examples of Dynamixel, Bioloid, Lego Mindstorms and Arduino show many professionals have tried to address the challenges of creating more sophisticated actuated objects and infrastructures supporting them. The disciplines include robotics, architecture, automotive and manufacturing industries, art and the DIY or tinkering communities. As more and more actuated objects are created across multiple domains, it will be essential to increase access to domain-specific knowledge for a broader base of practitioners.

One important aspect dividing disciplines can be found in the design development methods. On the one hand, engineers and roboticists create initial constraints for their designs and implement them at the final scale. On the other hand, architects, designers and artists often develop models at different scales before settling on a final design.

Designing Across Scales

Transferability of solutions across scales and contexts is also limited by the kit-of-parts approach described above. Under this paradigm, each specific context and scale requires a specialized solution. Therefore, a shift in scale implies a re-specification of components. For example, all servos consist of a motor, a gearbox, electronics and sensors. However, different models

Figure 2.14 (Above) Diagram of the servo motors and their applications raising the question of moving across scale indicated by the red line. Every iteration involving a change in scale requires going back to the root of the diagram and re-implement the servo at the specific scale

and sizes imply very different implementation requirements. Thus transferring a robot design from a small scale to a larger one will mean redesigning the entire object because the large servo requires different cables and connectors, other software protocols and a customized hardware solution.

The servomechanism remains the underlying principle and is split into different implementations mainly based on scale and size. Depending on the path of implementation, a specific mode of communication and pattern of control has to be chosen in order to make use of particular servo-hardware. This particular servo-hardware is designed for a specific and limited field of applications. The diagram (Fig. 2.15) shows the Bioloid humanoid robot as application for the RC-Dynamixel Servo, the research robot Domo as an application for mid-size servo-hardware and various industrial robot arms as an application for large-scale servo-hardware. Although the same principles and core components apply at different scales, they are rendered in radically different hardware and software implementations. As indicated by the red arrow in the diagram, it is impossible to cut across scales directly without traveling back to the origin of the design tree and selecting a different path of implementation. Some domains such as product design, object design, auto motive design and architecture are more likely to require moving from one scale to another during an iterative design process because prototypes and final implementation often differ scale. Other domains are more likely to build prototypes and final objects or artifacts at same scale. In the first case, development starts with a small model for testing and evaluating rough aspects, moves on through different iterations and scales to solve further details and finally reaches full scale. In the second case, initial design constraints are immediately translated into the final scale and each iteration is likely to take place at that final scale as well.

Promise of Digital Fabrication

Advanced CAD/CAM design and fabrication techniques have become increasingly ubiquitous and accessible. (Gershenfeld 2005) In particular, 3D-printing and other rapid manufacturing tools have enabled a shift towards fine-grained customization of mass-produced objects. Customers are thereby more closely connected with the design of their purchases. Web-based parametric CAD software and design tools are expanding people's access to the underlying tools for design in a way that goes beyond styling to include engineering questions such as gear-box typologies. Global, web-based manufacturing resources such as mfg.com allow end-users and makers to easily manufacture and test their creations by connecting them directly to fabricators.

The design and application spaces spanned open through these recent developments in CAD/CAM are underexplored.

Figure 2.15 (Above) Ealry exploration involving CAD software and digital fabrication in conjunction with mechatronic.

For example, more specialized and diverse designs can be created. Not all behavioral objects must fulfill the same set of constraints or need to be optimized towards the same criteria. Not every robot needs to do heavy lifting or cut and weld metal in extreme precision. Relying on off-the-shelf components or being limited by domain-specific standards, force designers to select components that are not perfectly matched to their vision. Having a behavioral object smile at you or grow or breath does not necessarily require high power metal servos but can be achieved very gracefully and coherently with the use of some custom components.

New fabrication methods will also enable the creation of fully functional shapes or even whole assemblies which would not have been possible using previous manufacturing techniques. Lower rapid-manufacturing costs will enable more iterations and streamline the integration of components. They will also enable the creation of more diverse original machines.

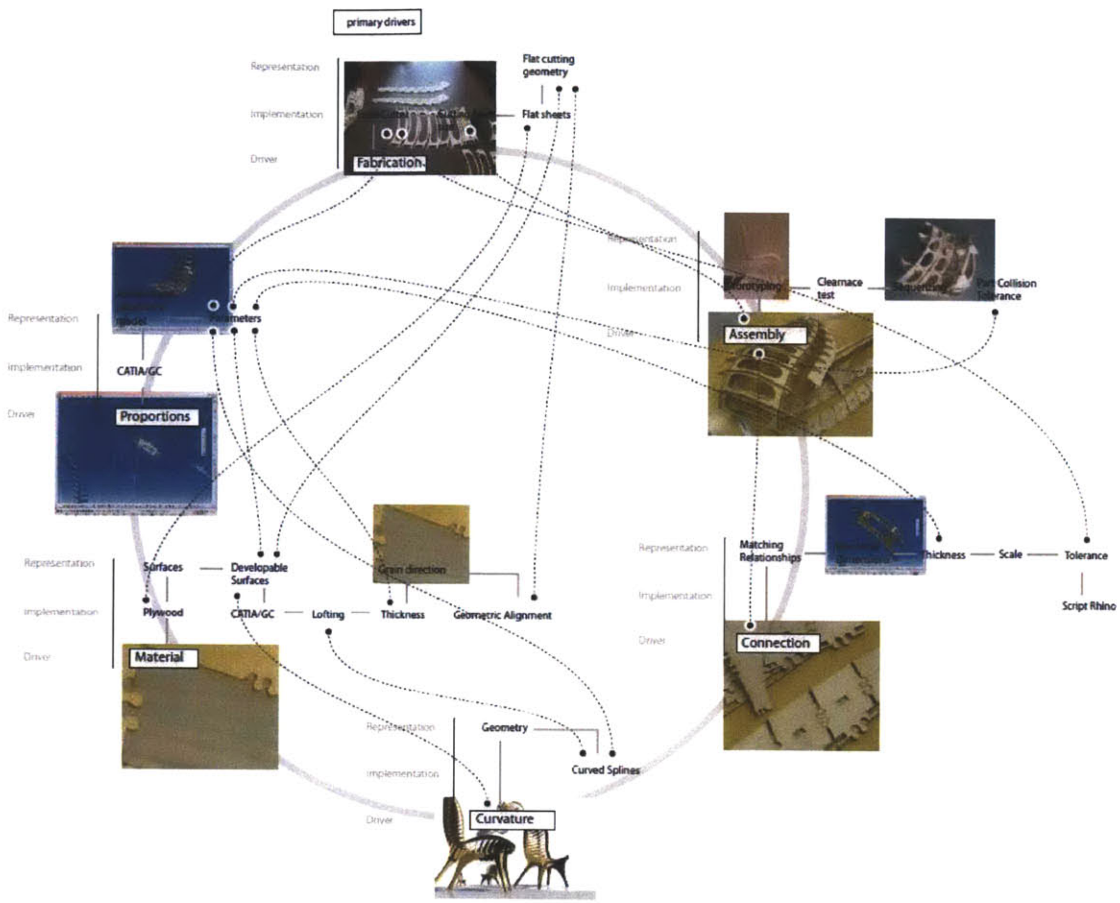
These emerging technologies also enable more individualized solutions because one-off pieces become financially feasible. Unlike industrial designers in the past, creators have more freedom to design and build different iterations thanks to various rapid manufacturing processes. Therefore, these tools provide us with new opportunities to focus on the tension between materiality and experience described above.

The opportunities for digital fabrication blur the boundaries between prototyping and manufacturing. As materials become more robust and certain machines like 3D-printes more affordable and accessible, it no longer makes sense to distinguish a priori between a rapid prototyping and rapid manufacturing machine. What matters is the intention of the designer or creator engaging with it. Throughout this thesis, I focus on users who are interested in creating robust objects that can last for a significant period of time and function at full-scale.

Precedents

The precedents described here place originalMachines in the broader context of design and mechatronics (robotics). Inspirations are drawn from computer science, robotics and mechanical engineering, on the one hand; DIY, tinkering, art and design on the other. I also describe the tools and processes which accompany CAD/CAM technologies in order to potential areas of opportunity for integrating these systems more effectively into the design, fabrication and control of original machines for domains such as toys, consumer electronics, RC and robot servos like Dynamixel, robotic kits like Bioloid and Lego Mind Storms.

The examples are subdivided into three groups. First, an array of projects by artists, designers and architects show the range of scales and criteria used in digital fabrication processes. Second, more consumer-product related products derived from parametric manufacturing processes and on-demand fabrication show how a more general audience is accessing new types of creations. Finally, there are precedents within the domain of mechatronics and robotics that focus specifically on the integration of complex objects with electronics to enable interactive experiences.



Artists, Designers and Architects

Figure 2.16 (Above) Axel Kilian, Design Explorers, example of a chair illustrating bidirectional dependencies of all aspects involved with the chair activle driving the overall design or passively being driven by the overall design.

Digital fabrication has become a widely used term within design and architecture. It is often used to imply cutting-edge processes and new forms. The following overview does not aim to be exhaustive, but rather focuses on examples where digital design methods are understood as one holistic process. The examples are representative of different approaches to this challenge within current architectural and design practice and research.

Design Explorers

In Design Exploration through Bidirectional Modeling of Constraints Kilian (2006) describes a design method which takes into account the various relations among "design drivers" and their resulting bidirectional constraints. With the example of a chair, Kilian shows that idea, concept, form, material, connection details, fabrication methods and assembly have to be dealt with as a whole rather than individually and sequentially. By choosing the idea of a chair, the material of plywood and a laser cutter as

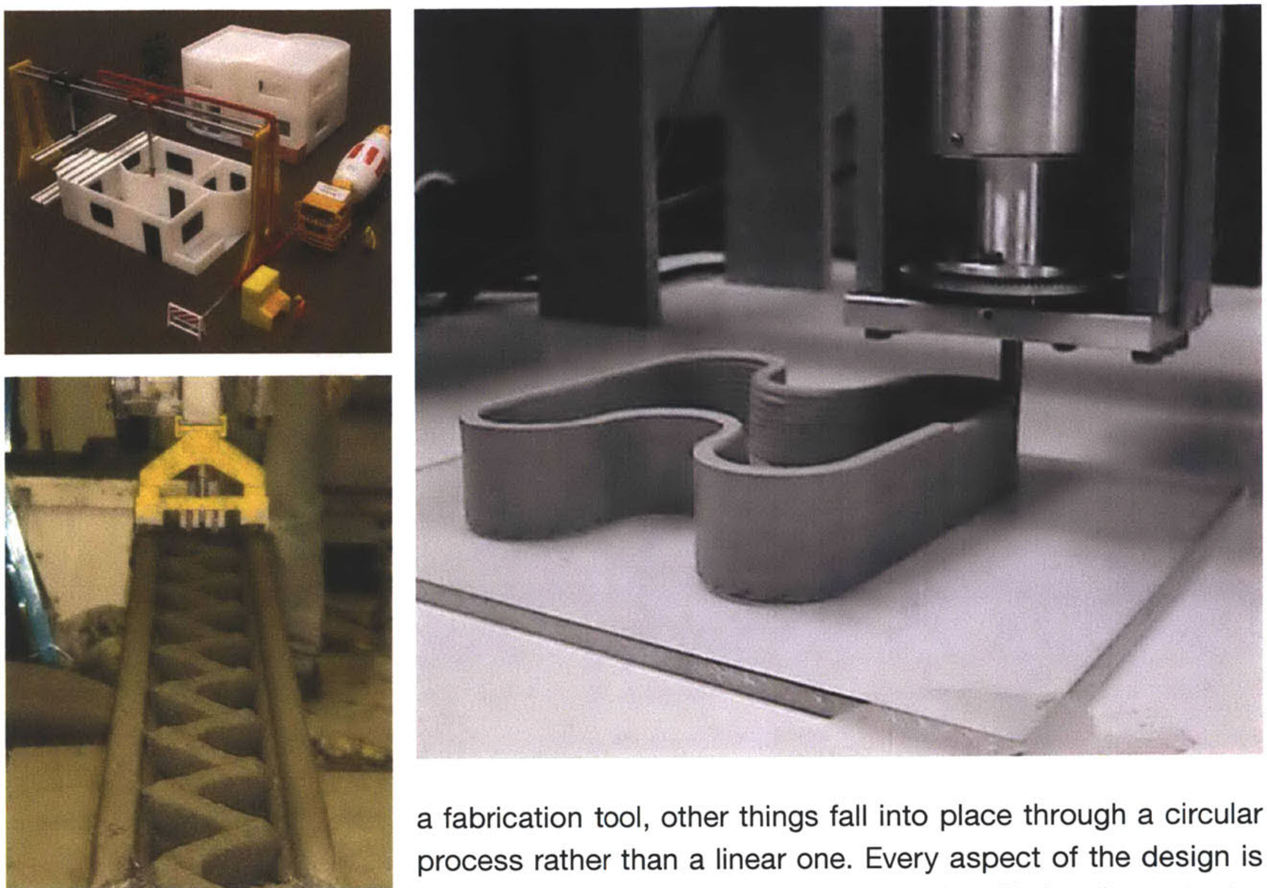


Figure 2.17 (Above) Contour Crafting, automating construction of whole structures and structural elements by Behrokh Khoshnevis USC (http://www.contourcrafting.org). Detail of wall structure created with Contour Crafting. Detail of Contour Crafting nozzle in action

a fabrication tool, other things fall into place through a circular process rather than a linear one. Every aspect of the design is driven by every other aspect of the design. Today, few actuator assemblies are selected with this kind of process, but there is great potential to create a host of new original machines using the design driver approach.

Contour Crafting

Contour Crafting is a research project at the University of Southern California headed by Dr. Behrokh Khoshnevis. The project focuses on using an XYZ Cartesian robot for layered fabrication of extruded fast-setting concrete. A structure rigid robot moves a nozzle along a path corresponding with the walls of a section drawing (contours). Each time the nozzle moves along a certain track fast-setting concrete is extruded with a layer thickness of several inches and straighten out on the sides with trowels mounted next to the nozzle. Doors and windows are spared out and bridged with prefabricated elements and finished with additional layers of concrete. The same applies to the ceiling, prefabricated elements are added by a second actuator supported by the same structure rigid robot and finalized with additional layers of extruded fast setting concrete.

This system allows for concrete fabrication of structural elements and complete assemblies needed for buildings. The size of the fabrication depends on the size of the robot which could be as big as construction cranes capable of spanning the dimensions of high-rise buildings.

Though extremely innovative, this method for automating the construction of entire structures has several limitations. First, it is nearly impossible to build horizontally. Second, the size of the final structure is limited by the size of the machine. Finally and most importantly, the technique limits the possibilities of formal architectural vocabularies. The material, scale and formal limitations make it difficult to envision new types of forms such as those envisioned by cutting-edge designers today.

DFab, ETH

The architects Gramazio and Koehler head the Architecture and Digital Fabrication program at the ETH in Zuerich. Since 2006, an industrial robot arm, Kuka CAMRob system, has been in use there for pick-and-place applications of componentized assemblies, milling based subtractive machining and several projects exploring additive processes involving foam. The scope of the research spans from custom brick patterns used for wall components and ornaments to custom perforated sheet and wall elements to custom patterns created from foam as well as timber structure assemblies. The projects shown in Fig. 2.18 try to resemble an overview of the research scope. The typical process involves some kind of script-generated geometry which interfaces to the CAMRob through the Kuka Robot Language (KRL). The industrial robot arm which is mounted on rails allowing for several meters of linear motion will then cut, glue and assemble the segments and components. As the working envelope of the robot is still fairly limited compared to architectural scale, sub-assemblies are created and transported to the construction site where they are finally installed.

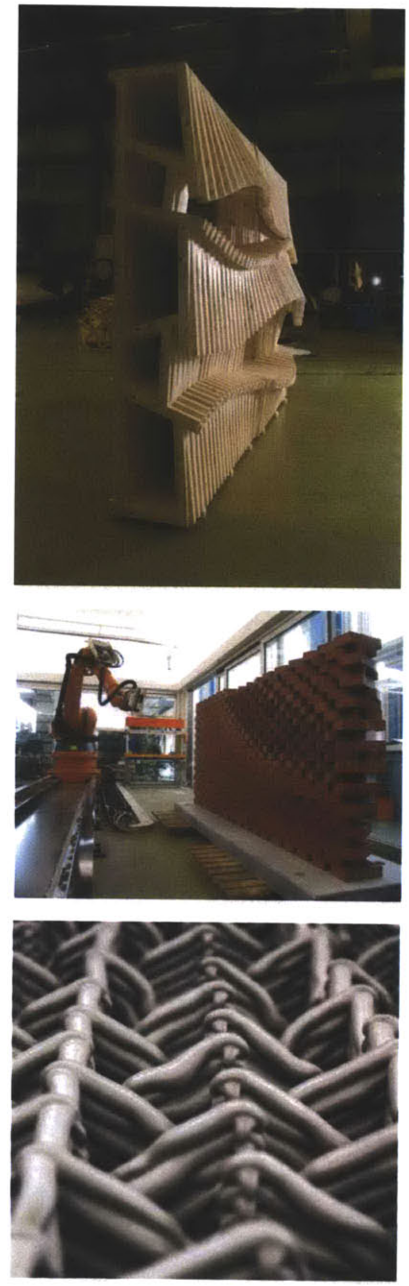
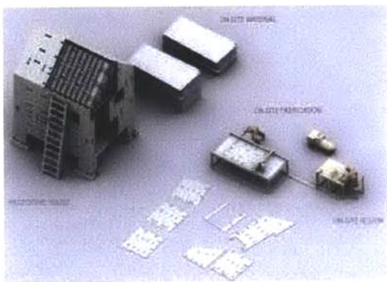


Figure 2.18 (Above) DFab at ETH using Camrob, Gramazio & Kohler, (http://www.dfab.arch.ethz.ch). Camrob robot arm assembling a wall. Additive foam extrusion process.

Another approach involves reversing the need for transportation by shipping the robot arm on-site as in the R-O-B or Pike Loop projects. Still, the working envelope of the robot is limited and mostly enables the creation of sub-assemblies rather than



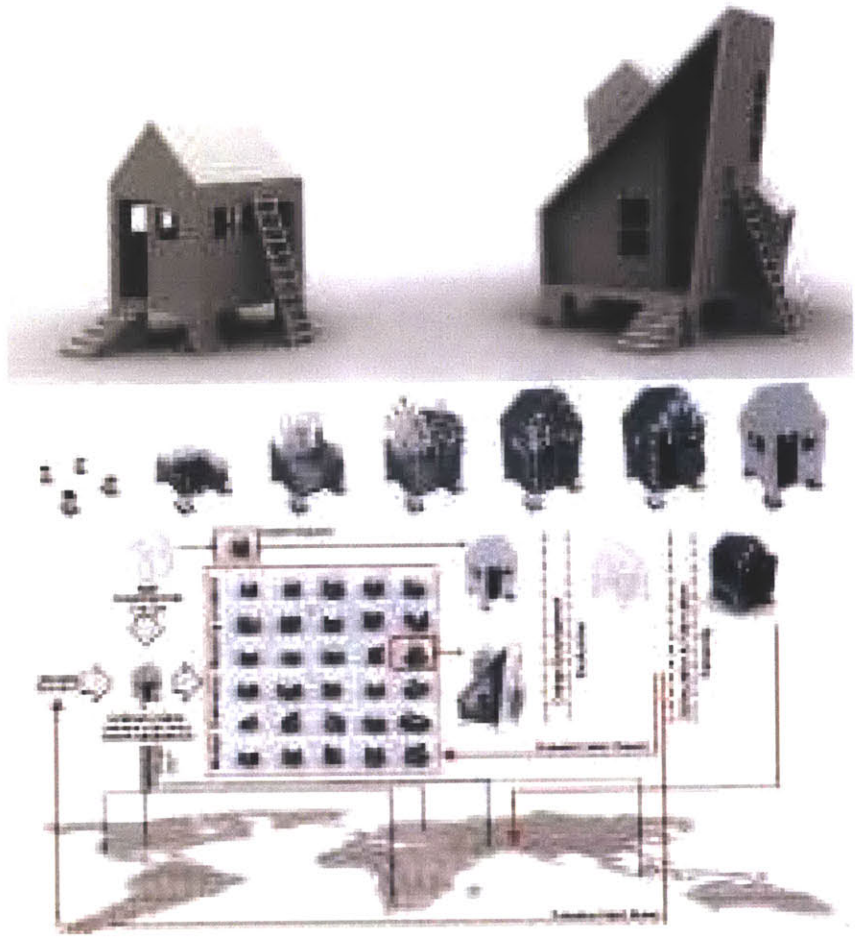


Figure 2.19 (Above and Right) yourHouse shown in New York City MoMA exhibition. Diagrams of on-site fabrication, Marcel Botha SMArchS Thesis MIT, 2006). Diagrams of on-site fabrication, Marcel Botha SMArchS Thesis MIT, 2006)

whole structures. Even with a bigger span the issue of track-based linear motion of the robot arm would limit the system in terms of its ability to create large scale structural assemblies. From a software perspective, each project depends on a large amount of custom script that will make it difficult to scale up the system towards projects on a different scale or projects involving multiple robots.

yourHouse

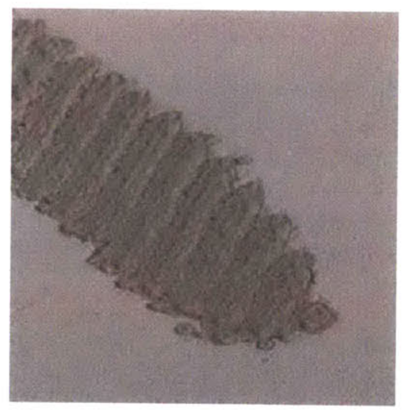
yourHouse is a project which originated at the MIT Department of Architecture under Prof. Larry Sass. It focuses on prefabricated home construction using a software based parametric description of houses allowing the reinterpretation of typical New Orleans "Shotgun" houses. In a second step, the software breaks down the entire house into components which can be cut out of flat sheet-stock material (timber, plywood, plastic) using a CNC router. The software also takes care of all connection details and parts labeling. The parts cut by the CNC machine then need to be assembled manually. The router and sheet-stock material can be transported onto the site which makes this process relatively mobile and flexible.

yourHouse is a successful example of digitally fabricating the components of homes. In this model, the solution space for expressive architectural form and language is closely related to the chosen process, parameters and constraints. Using flat sheet material and a router, any 3-dimensional objects will be reduced and approximated by 2-dimensional representations which themselves are constrained by their connection details and structural properties. The overall result closely resembles the intended style and features but does not necessarily convey the richness and flexibility of the existing New Orleans architecture. Unlike the previous examples, this project focuses on the fabrication of parts rather than on digital construction of houses as it creates parts and components which need to be assembled by hand.

Dshape

Using sandstone powder and inorganic "structural ink" as a binder, the Dshape Company developed a large scale 3D printer. Similar to a Z-Corp 3D printer, Dshape used a XYZ Cartesian

Figure 2.20 (Below and Left) Dshape 3D printer. Detail of the 3D-printing material (similar to Z-Corp plaster). Radiolaria Structure made with the Dshape 3D printer



robot to impregnate powder with a binder in a layer-based process. Dshape has taken this principle to a new scale of 6 by 6 by 6 meters. The finished 3D printed parts are rough and require manual finishing. The company claims to be capable of developing this process to an even bigger scale that can serve as a cheap replacement for traditional construction methods. To print a building-like structure, the Dshape process requires sandstone powder to fill the entire working envelope of the machine while it impregnates only the cross section area of walls on each layer. This leads to a huge overhead of scaffolding, mold making and material handling. In addition, each printed building needs to be extracted out of the sandstone powder. The structural properties of the finished sandstone composite are not known yet and in combination with the huge efforts of initializing a printed construction and the poor detail and feature quality this approach still needs to come a long way to fulfill its promise of replacing existing construction methods.

Hyperbody Research Group, Kas Oosterhuis

The Hyperbody Research Group at the TU Delft under Kas Oosterhuis mainly focuses on exploring interactive architecture. Some of the applied research projects explore industrial muscle actuators and demonstrate how they could be implemented into structural assemblies. Though this research does not primarily focus on digital fabrication and construction, actuated structures could inform and assist digital construction. Structures with implemented actuators present a whole new opportunity for the process of erection and construction for example the idea of assisted self erection.

Figure 2.21 Muscle Tower II of the Hyperbody Research Group at the TU Delft (http://www.bk.tudelft.nl/live/pagina.jsp?id=42d12e00-5d78-42d1-afe0-262352934565&lang=en)

Since their introduction CAD CAM methods have shaped architecture, fabrication and construction significantly. Almost all steps involved in the process of conceiving, designing and making a building can be originated in software. The same software package allows an architect to conceive and design a building, the structural engineer to analyze and adjust the structure, the general contractor to subdivide and price the building and the subcontractors and fabricators to operate machines creating the components. The same software package can also be used to coordinate the workflow on site and manage the erection and assembly of all components as well as maintaining the building after its completion. CAD/CAM methods have taken over the traditional architecture and construction process and streamlined it towards an integrated digital methodology.

An important enabler for this process is the concentration of computational resources on prefabricated components. The software enabled standard method implies efficiently breaking down any 3D object into standardized or unique components as well as their outsourcing to workshops. The process of slicing down the building into sub-assemblies and components is very precise as is the computer numerical controlled making of them. The huge coordination and transportation efforts resulting from the large number of parts and components can also be managed using software. The process than calls for precise on site assembly which from case to case might become difficult as it brings together all components created by different workshops with structures erected manually on site. An example of a building project undergoing this process described above is the MIT Stata Center.

In a project like the MIT Stata center, the building is subdivided into components which can be outsourced to shops and factories where they can be made conveniently using all kinds of specialized machine tools. Most of these machine tools are stationary equipment rendering many different jobs throughout the day while other construction equipment like cranes, man lifts and power tools are rented out and located at the construction site. Precision machines used in fabrication and construction are typically robots and CNC machines creating parts, components and sub-assemblies in great precision, fidelity and quantity which

Figure 2.22 MIT Stata Center under construction, Frank Gehry

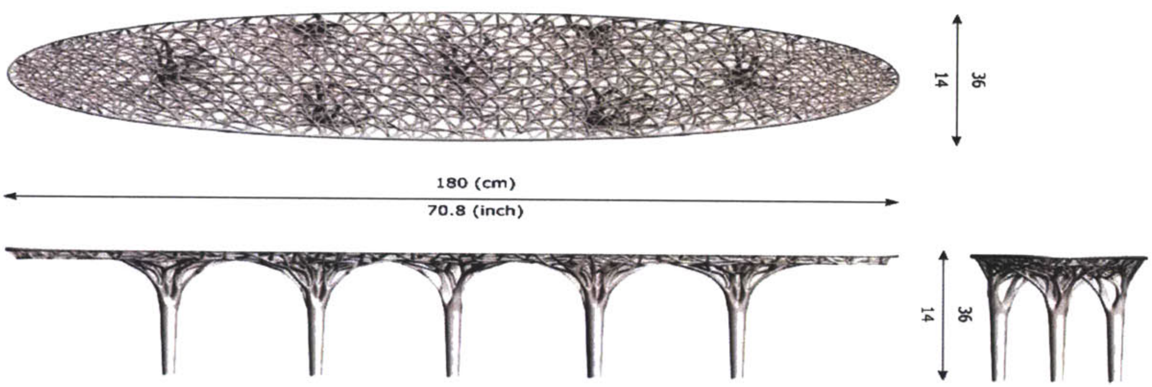
also creates many constraints which drive back the process itself. A simple example is that components must not exceed road traffic safety requirements in size, weight or shape because they must be shipped to the construction site. This means although components and sub-assemblies could be made bigger they will have to be cut down in order to be transported. This constraint adds more connections details which need to be engineered, documented, rendered and assembled. Also a larger number of components will increase overall complexity enormously which needs to be compensated by more sophisticated software and labor. As a whole, this process is growing increasingly complex which leads to a growing preoccupation among professionals and academics to find alternative processes and technologies.

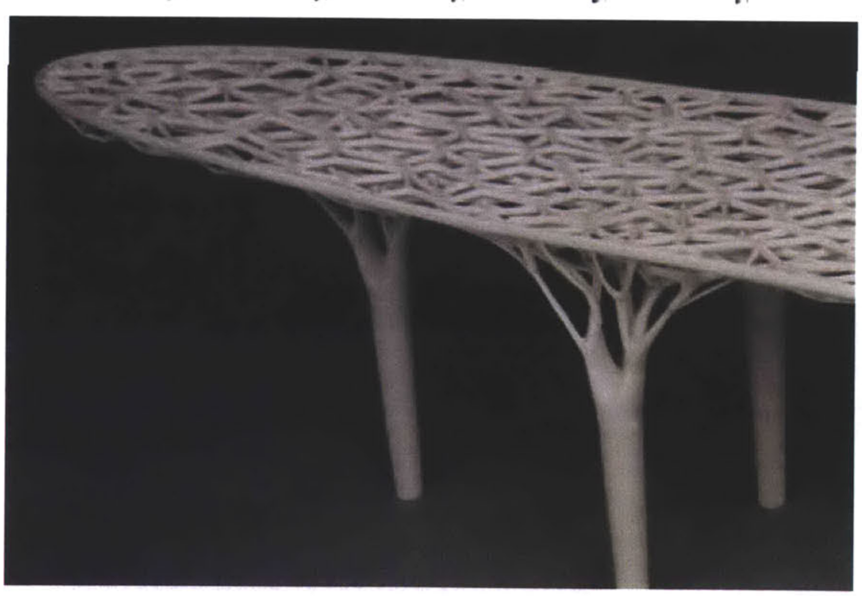
3D-printing has received growing interest from the architectural and construction scenes for full-scale projects. As shown above Contour Crafting and Dshape are building upon this technology and trying to refine its feasibility for construction purposes. But the field of 3D printing is broad and spans more innovative and pioneering applications. The company Freedom of Creation is pioneering 3D printing as a rapid manufacturing process for design objects.

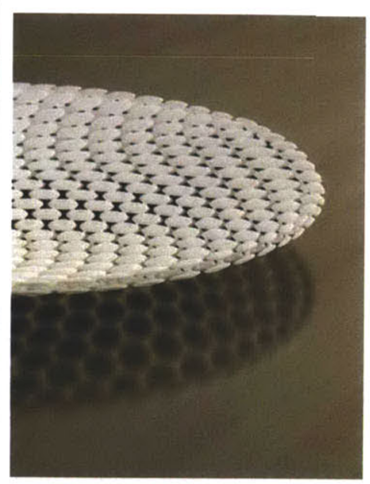
Freedom of Creation

The adjacent images show some of the products created by Freedom of Creation ranging from design objects to furniture to 3D printed fabrics. Originating from a design and product design based background a small group of designers started out to pioneer 3D printed design as a business model. The freedom of creation web page informs, shows and sells 3D printed products reaching from lighting over futniture to assecories. The webpage offers products which will be 3D printed upon ordering and mailed to the customer. Different material options are available one of them being 24 carat gold.

Figure 2.23 (All Page) made by Freedom of creation only utilising laser sintered Nylon 3D printing: Lamps, chain mesh gloves and fabrics, furniture up to the size of a bench and fruit tray.







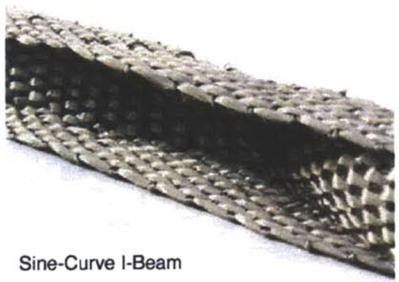


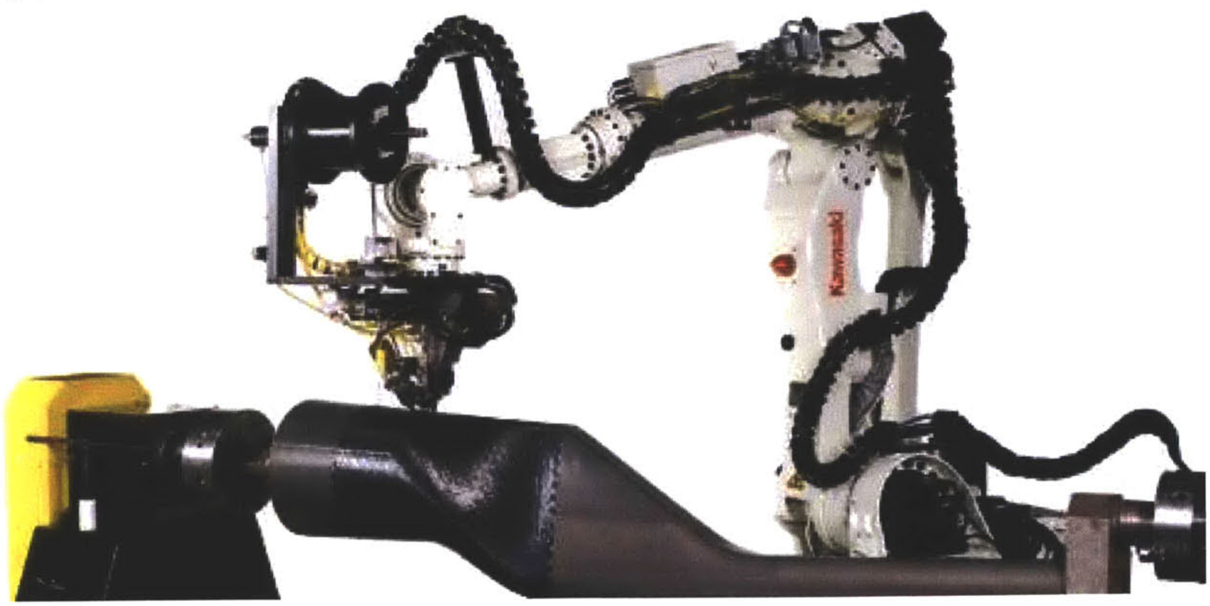
Figure 2.24 (Above) 3D weaving examples by Shape 3 and Bi Team

3D-Weaving Techniques

Another field in the landscape of emerging techniques with relevance to digital construction is 3D weaving. The companies Shape 3 and BiTeam are pioneering using looms and other weaving devices in order to create curved and double curved shapes rather than flat fabrics.

This technique becomes important in combination with fiber reinforced composite. Typically molds are required in order to shape the composite while 3D weaving allows any composite to have almost any shape without the use of a mold. This flexibility cuts costs, complexity and time involved in composite making tremendously. Again the machines used for this technique are looms and mostly stationary meant to produce parts for supply chains of products. The importance for digital construction lies in the likelihood of making the stationary equipment mobile. As some of the weaving and knitting machines are fairly small in size they could be mounted as tool-heads on the end of robot arms or construction cranes. In fact, automatic tape laying machines

Figure 2.25 (Below) Automatic tape Laying Machine



are already close to this vision although they require molds as their only purpose is the placement of fibers for composite.

An overview of contemporary high-end engineering applications for textiles was given by the exhibition "Extreme Textiles" at the Cooper-Hewitt NY. Fig. 2.26 shows the ISO-Truss which was featured at the exhibition. The ISO-Truss is an example of light-weight, efficient composite construction which could easily be extended towards digital construction.

Figure 2.26 (Above) ISO Truss lightweight carbon fiber structure supporting the weight of a person climbing it.

The Knitting Machine, David Cole at Mass MOCA

From June till December 2005 David Cole's "The Knitting Machine" was performed and exhibited at Mass MOCA. The artist led a team driving two excavator machines using utility poles as giant knitting needles to create a 30 by 20 foot American Flag. The performance was inspired by the idea of national pride and America's role in world affairs. Aside from the artists intention, the performance provides an example of construction-machine-made, architectural-scale fabric evoking the idea of creating inhabitable space with a similar process.

Figure 2.27 (Below) "The Knitting Machine", David Cole at Mass MOCA

Parametric Manufacturing and On-Demand Fabrication

In response to the increasing availability of CAM, more services and design studios are emerging with a focus on customized, on-demand manufacturing. There is a range of services for design and manufacturing with different takes on the role of the creators. In some cases, people design their own objects and rely simply on the manufacturing services. In others, the company offers design tools that allow for different degrees of customization on the part of the user.

i.materialise and Nervous Systems

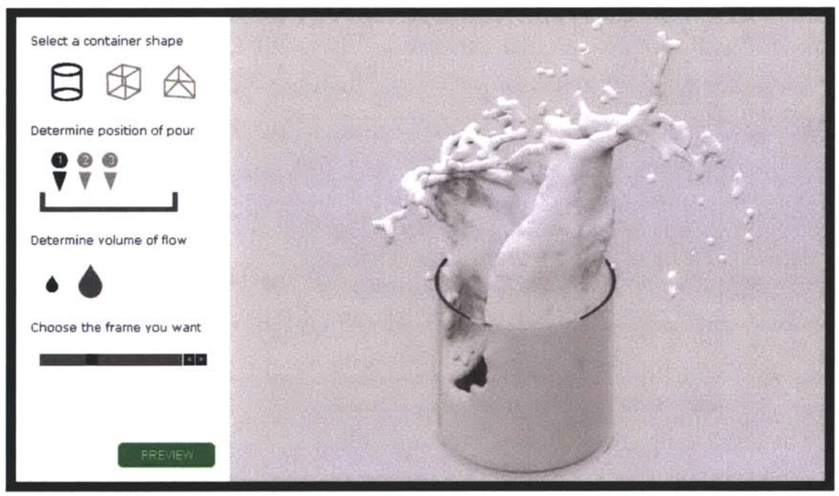
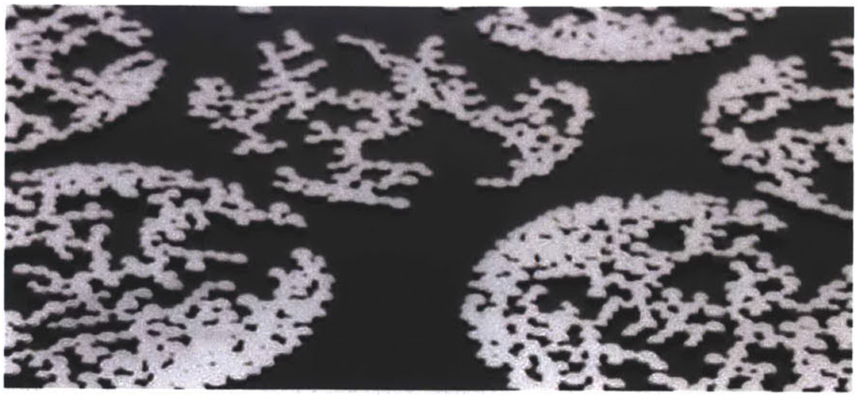


Figure 2.28 Fluid Vase, http://freshbump.com/graphics/image_, Screenshot of the Fluid Vase Design Tool, http://www.solidsmack.com/wp-content/uploads/2010/11/liquid-vase-3d.jpg files_480x400/480x400_fluid-vase.jpg

i.materialise offers 3D-printing services and tools for creating new objects. (http://i.materialise.com) For example, they offer design studio Supabold's (Fung Kwok Pan and Chong Han) "Fluid Vase". The studio created a parametric tool that simulates a water drop splashing into containers of different sizes. The user selects a preferred position within this dynamic process and freezes it. This form becomes the final shape of the vase, unique every time. In addition to specialized tools such as the vase, i.materialise offers a 3D print lab for finishing 3D models. Nervous Systems is a design studio based in Massachusetts specializing in parametric lasercut and 3D-printed jewelry. (http://n-e-r-v-o-u-s.com) They also offer parametric tools for customer to specify patterns and shapes of jewlery.



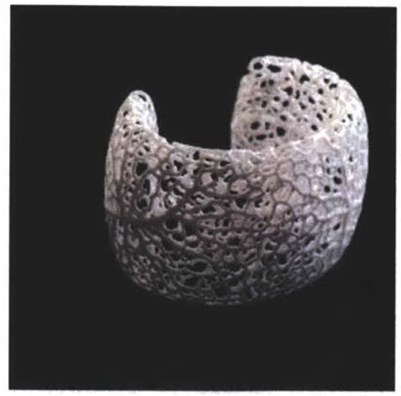
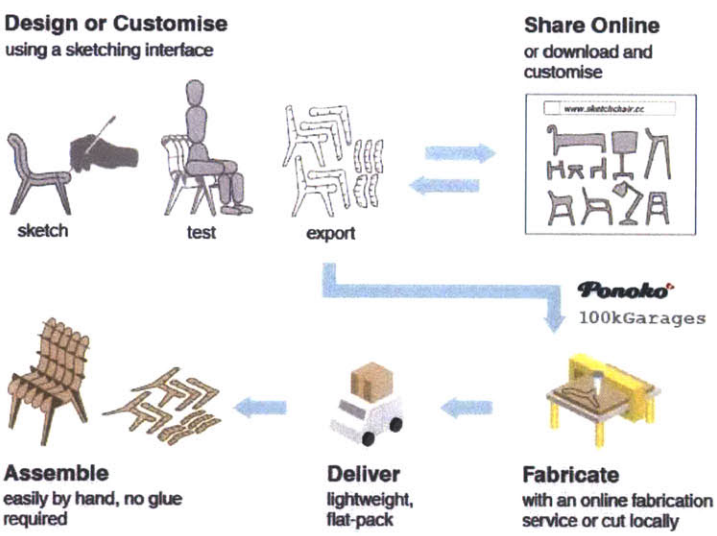


Figure 2.29 (Above) Examples of jewelry by Nervous Systems.

SketchChair

SketchChair is a software-based project aiming to support the concept, design, testing and fabrication of seating furniture. Through a simple four-step process (sketch, test, cut, fabricate) seating furniture can be created inside a computer and constructed with the use of digital fabrication tools such as a laser cutter. SketchChair is an easy-to-use tool aiming to be used by end-users assuming access to digital fabrication resources. It is a great example of user driven innovation and the users desire



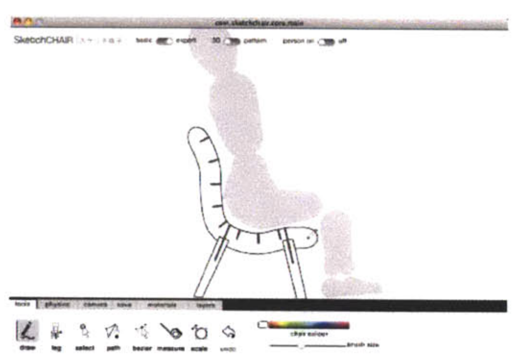


Figure 2.30 (Above) SketchChair

to gain control again over the physical objects surrounding us. (http://gregsaul.co.nz/SketchChair, http://diatom.cc/sketchchair)

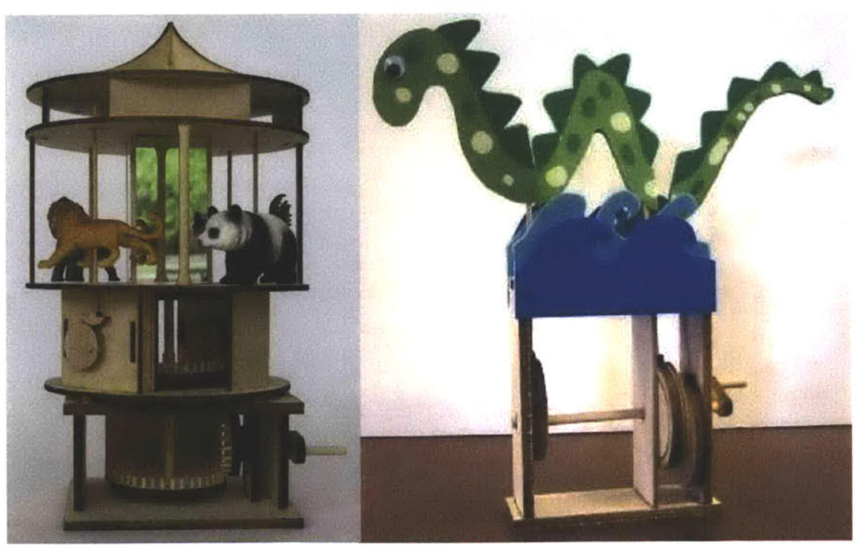
Shapeways and Ponoko

In the fast growing and changing landscape of digital fabrication service providers Shapeways and Ponoko stand out by not just offering access to various resources like laser cutter, cnc machines and 3D printer but also through their community focused web platform. Both companies allow for users to open accounts through which they can share their designs and products on the web platform gallery where they can be ordered. This model allows for individuals to start up small business cases involving digital manufacturing at no start up costs. Ponoko recently started to focus on what they call a digital product which is a data container allowing a product made from different parts to stay digital as long as possible. This digital product can also contain user manual, assembly instruction and packaging specs enhancing it on a holistic level to be comparable to a standerd consumer product

Glen Blauvelt, Machine shop

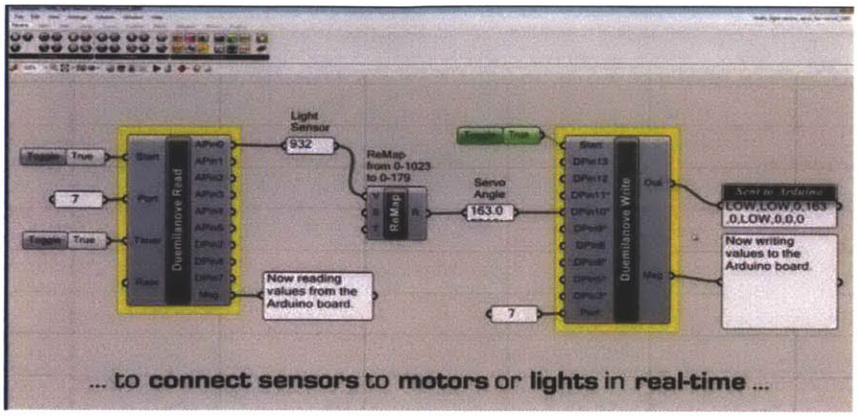
MachineShop is software developed by Glen Bauvelt as part of his PhD research. It is tailored towards middle-school aged children and allows them to create mechanical automata. It combines computational resources with the knowledge of mechanical craft objects towards a design tool and demonstrates how these systems can be expanded for more users.

Figure 2.31 (Right) Examples of creations made with the Glen Bauvelt's Machine Shop. Images from http://l3d.cs.colorado.edu/~zathras/



Mechatronics and Robotics

Firefly Experiments



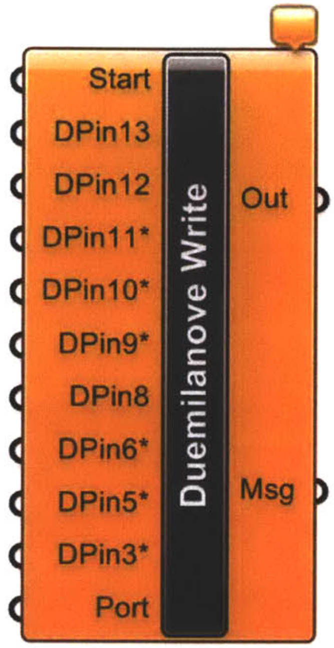


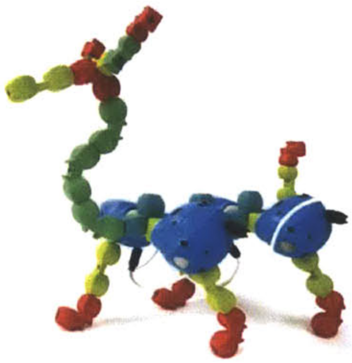
Figure 2.32 (Above) Screenshots of the Firefly experiments. Images from http://www.fireflyexperiments.com/

Grasshopper (David Rutten, ongoing) is a free plugin for the commercial CAD software Rhino that allows visual programming for all Rhino related commands, tasks and functions. (www.fireflyexperiments.com) Firefly (Andy Payne, Jason K Johnson 2010) Experiments extends Grasshopper with a real-time Arduino representation. It allows for a visual representation of the Arduino board and its real-time IO readouts. This allows for linking any Arduino based real world project with CAD software. For example, a joint modeled and described within Rhino and Grasshopper can be linked with its real world representation and any manipulation within the CAD environment will affect the physical object and vice versa. This powerful link creates a pipeline for concept, design, fabrication, actuation and animation in one software environment tool.

Molecubes

Molecubes (Zykov, Chan, et al. 2007) is a research project at Cornell University in the field of self-assembling and reconfigurable robots and programmable matter. Using metaphors from nature of self-repair and self-reproduction, the researchers have attempted to create an evolutionary robotic system that can be adapted to many needs. The project has an open-source component that has led to further developments, for example by the German company FESTO.

Figure 2.33 (Right) Molecubes developed further by FESTO, http://www.festo.com/rep/en-us_us/assets/CC_08_09_Molecubes_2_500px.jpg



Topobo

Topobo by Raffle, Parkes and Ishii is a toy and construction kit with embodied kinetic memory. Consisting of different snap-fit modules children can assemble animal-like or biomorphic robots which can be animated by recording physical motions. First, the user moves the creature by hand to record a motion. Then, it will repeat the movement autonomously. This project represents a

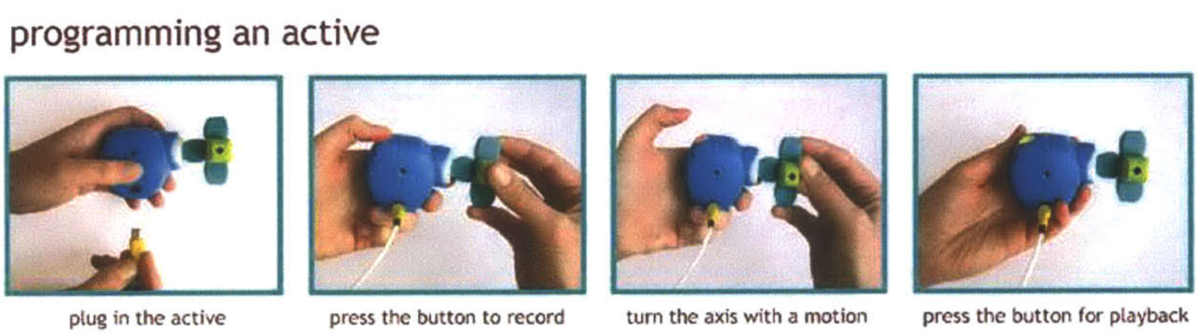


Figure 2.34 (Above) Griffin Walking created with Topobo kit, http://www.topobo.com/topobo-photos/pages/2_griffinWalking.html. Programmin an active servo element in Topobo, http://www.topobo.com/topobo-photos/pages/5_programming.html

simple and elegant approach towards the usually complex task of dealing with kinetics. (www.topobo.com)

Modular Robotics

A spin-off from Carnegie Mellon University, Modular Robotics (www.modrobotics.com, Gross et al.) creates robotic construction kits for kids. Their initial products are entitled cubelets that snap together using magnets. Similar to Topobo, different elements have different capabilities so that children can watch complex behaviors emerge (rather than programming them on a computer and uploading code to the blocks).

Conclusion: Reigniting the Relationship between Making and Objects

The preceding examples show different degrees of connection between the fabrication technique and the resulting forms. I am most inspired by those projects with a tight coupling between construction technique, material and form because they reignite the relationship between making and objects. In his essay "On Weaving a Basket," Tim Ingold (2000) overturns conventional distinctions between natural organisms and man-made artifacts. Ingold considers "making as a modality of weaving" rather than vice versa. The emergent nature of the weaving process blurs the distinctions between making and growing. For Ingold weaving is a special case within craft because the form results from a field of forces rather than from a predetermined representation of the object: "The artifact, in short, is the crystallization of activity within a relational field, its regularities of form embodying the regularities of movement that gave rise to it." (Ingold 2000: p. 345)

Translating from Ingold's weaving metaphors to other areas of making implies a freer dialog between construction processes, forms and a design's evolution. "If forms are the outcomes of dynamic, morphogenetic processes, then their stability can be understood in terms of the generative principles embedded in the material conditions of their production." (Ingold 2000: p.346) Combining new fabrication techniques with this holistic (and generative though that is not the focus here) approach could support new types of makers in creating unexpected forms for a new category of objects.

Chapter 3

Cook Book for Digitally Fabricated Mechatronics

The specifications and features for an object-oriented mechatronics design tool are informed by a series of design explorations that investigate the boundaries of available rapid manufacturing techniques for use with mechatronic assemblies. The main focus is on power transmission elements like gear boxes and bearings as they represent a primary building block for any actuated object. The typology investigates a range of commonly used power transmission assemblies: planetary gear boxes, cycloidal gear reducers, harmonic drives and worm/ ball worm drives. In connection with the planetary gear box, I include spur and helical gears. In addition to power transmission elements, I also explore joints and bearings as they play a key enabling role for any kind of motion. The following chapter is a typology of these elements rendered in different materials like wood, plastic and metal with different rapid manufacturing tools like the laser cutter, cnc mill and various 3D-printers. I've entitled the chapter "cook book" because it includes specific techniques and strategies for creating mechatronics with the latest CAD/ CAM technologies.

The chapter also highlights the challenges involved in adapting existing knowledge about gearboxes and mechanical assemblies for new types of fabrication. The challenge for the typology is to find appropriate translations, settings and strategies depending on the intention for the desired mechanism, the material and the tool. For example, some laser cutters cut on the line rather than including an offset and they all create tapered edges like shown in

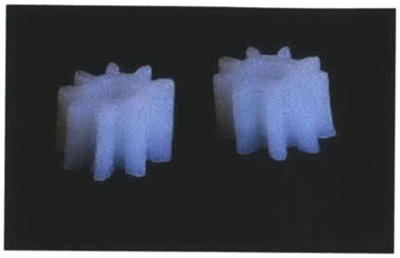


Figure 3.35 (Above) Spur Gears cut on the laser cutter showing the taper and some additional leaning effects created by the laser cutter.

the adjacent image. This presents challenges for cutting precise shapes and precision assembly of laser cut parts like gears and gear boxes. While going through the different rapid prototyping machines I point out particular features and their implications.

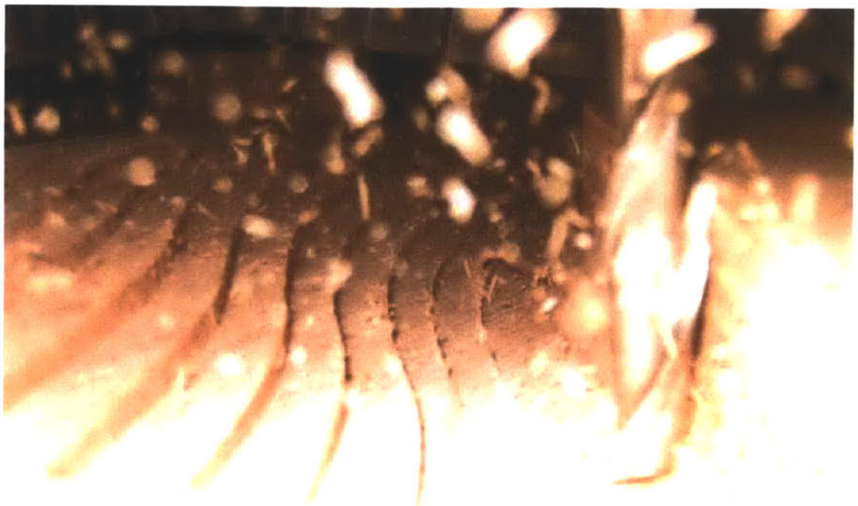
The chapter begins with an overview of three rapid prototyping and digital fabrication tool categories. Then I review a series of components. After describing the building blocks, I describe several integrated assemblies that overcome the traditional assembly requirement in manufacturing.

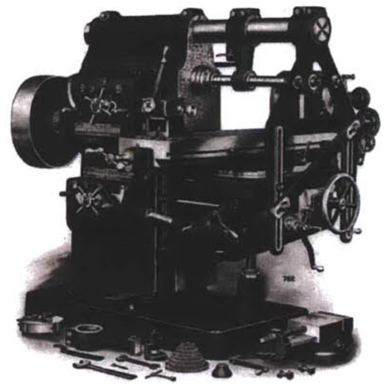
Overview of Rapid Prototyping and Digital Fabrication

Digital fabrication and manufacturing machines root back to the early CNC machines developed at MIT in the 1950s. An entirely new category of computer numeric controlled (CNC) machines was created and mainly used for industrial and military routing and milling purposes. Today CNC is the underlying principle for an even greater variety of machines for automation, fabrication, prototyping, assembly and also consumer electronics. Devices like laser cutters, industrial robot arms, automation equipment and 3D printers belong to this category. Through the digital revolution, many more users have access to CNC machines through web-based service providers. For the explorations shown and described in this cook book, I used the following machines available among others in the MIT Media Lab Fabrication Lab (Gershenfeld 2005): Universal laser cutter, Dimension FDM 3D-printer, Invision SI2 PolyJet 3D printer, Shopbot (three-axis with a 5 by 10 foot work area) CNC router, Modela mini mill, Omax water jet cutter, manual milling machine, drill press, belt sander and various manual tools.

CNC Milling and Routing Machines

The oldest form of automated cutting is the CNC milling machine. Manual milling machines have been in use for almost two centuries and belong to the domain of subtractive machines meaning they remove material from a bigger block (Fig. 3.36). In a milling machine a spindle rotates a cutting tool while a table capable of moving in two or three directions holds a work piece





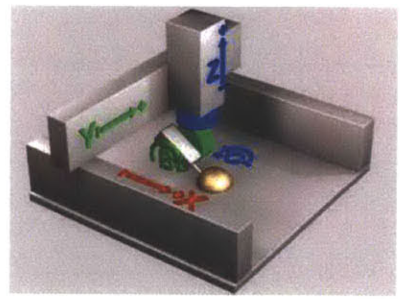


Figure 3.36 (Left) manual miling machine 1900-1920. Source: Cincinnati Milling Machine Company, Wikipedia. (Right) 5 axis gantry style CNC milling machine abstraction illustrating the machine frame and axis configuration. Source: http://cnc-toolkit.com/support.html Author: Rab Gordon.

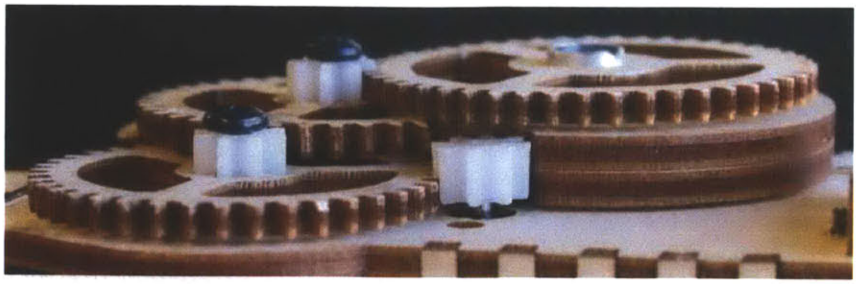
and moves it along the cutting tool Fig. 3.36. Each of these axes of motion has a positioning mechanism. Instead of a manual positioning mechanism like screws and dialing indexers numeric control machines add a motor controlled with a sensor. Special motor drive electronics are needed to start and stop the motor precisely which enables a computer to execute numerical motor control commands on the motor driver. Having motors drive each axis for multiple axes at the same time augments the envelope of machinable shapes from simple straight cuts to complex curved and double-curved results.

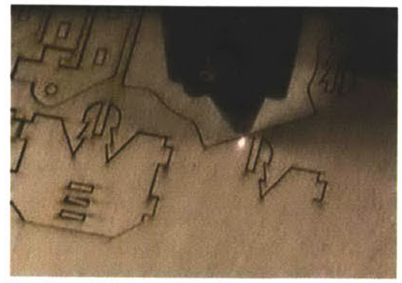
CNC milling machines exist in almost any size and specialization. Different machines are made for different cutting tools and materials like wood or metal. Shopbot offers an entry level machine for wood and plastic routing from 2 by 2 feet to 5 by 10 feet size. Metal milling machines are a much bigger domain and brands like Bridgeport, Haas and Hurco are the ones the MIT Media Lab shop is equipped with. CNC Milling machines are subdivided into their motion and machining capabilities depending on the number of axes. There are two-axis machines only automating X and Y axis while Z (the spindle height) has to be operated manually. More common are 3-axis machines where the X, Y and Z axess can be driven through a computer. These types are mostly Cartesian style gantry machines. 4-axis machines add a rotational axis on which the work piece is mounted within the framework of a 3-axis gantry. This additional feature makes it possible to machine the outside of a cylinder for example while rotating the cylinder. A 5-axis machine either adds two rotational axis holding the work piece or two rotational axis added to the Z axis holding the milling spindle (Fig. 3.47). The number of axes and their orientation determines whether the machine is capable of 2D machining versus 2 1/2D machining versus 3D machining. 2D machining means cutting out shapes from a flat stock material. 2 1/2D machining would be capable of creating surfaces without undercuts. For example ripples of water waves would be possible to machine while something like a human head can only be machines in a 3D 5-axis machine capable of reaching into and under every detail like nose, chin and ears.

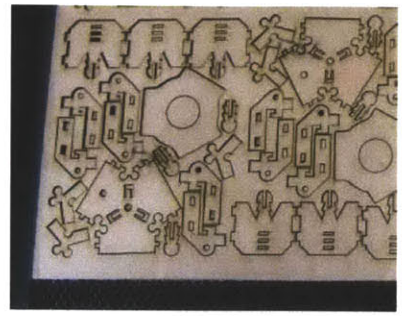
One common constraint to all milling machines results from the tool being round. Cutting out a rectangle will create sharp corners on the outside/positive rectangle but when the rectangle is a cut out like a hole in a surface the inner corners of the rectangle will be filleted with the radius of the tool. This comes into play when trying to notch sides of flat parts in order to connect them. A common way to avoid this problem is to add a pocket to all inside corners which will enlarge the cutting paths by the radius distance. Cutting tools exist as flat bottom or ball nose or V-groove bits. The flat bottom tool is best for cutting shapes and creating pockets. The ball nose tool is used for 2 1/2D and 3D surfaces. The V-groove tool is mostly used for engraving text and line patters but can also be used to cut out shapes having a 45 degree taper on all edges. This becomes very useful for making boxes or ball bearing raceways.

Laser Cutters

Laser Cutters also belong to the domain of subtractive machines and can be used to etch or cut different materials. Again for a variety of materials there are different laser cutter types as the wavelength of the laser light has to resonate with the material in order to heat, melt or vaporize the material. The differences in laser cutter machines mainly relate to the laser type. The general working principle involves a laser light source mostly stationary, sometimes air cooled but mostly water cooled. A pulsed laser light beam is emitted and delivered through mirrors and a focusing lens to the work piece. The laser energy heats up,







burns, melts or evaporates the material while accompanied by cutting gas which purpose it is to blow out the molten material and prevent fumes to condense on the lens which could cause the laser to cut the lens. The laser source does not emit a precise thin laser beam which makes it necessary to focus the laser before hitting the work piece (Fig. 3.37). This also causes the beam to be cone shaped. This cone shape accounts for the cut of the laser to create tapered edges as the cone diameter on the top side of the work piece is bigger than the cone diameter of the bottom size (Fig. 3.37). The smaller the focal lens the bigger the taper. This means two laser cut pieces can never be assembled into a perfect 90 degree angle by them selves but only in a bigger assembly. It also means gears do not mesh perfectly and the gear center holes will not be straight holes to accommodate an axle. A trick regarding assembly of two laser cut pieces is to flip one of the pieces as the taper created by the lase is constant and the angle of one taper will find its perfect counter-angle with the flipped second piece. Laser cutting wood will create burned sides depending on the cutting gas used from the natural resin of the wood and the glue used to laminate wood into plywood. Different sorts of Plywood created different burn marks the worst case being like graphite powder rendering the parts almost useless.

Laser cutters as well as CNC router and milling machines play a big role in digital manufacturing. For example wood as a material cannot be 3D printed and will always require lasers or end mill bits. 3D printers will also not eliminate the need for flat parts used in assemblies. Especially with 3D printers being fairly expensive flat parts still demand cutting rather than printing. Also size and working envelope of 3D printers are still very limited making laser cutter and milling machines a valuable choice for a hybrid approach to digital manufacturing.

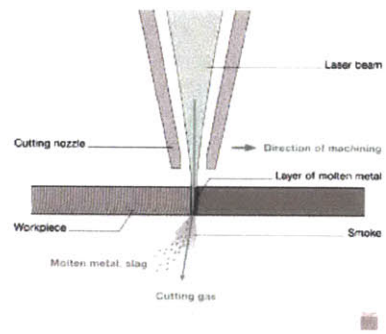


Figure 3.37 (Above)Lasercutter principle showing a beam of laser light being focused with the focal point being on top of the work piece. The cutting gas blows out the molten or vaporized material. Source: http://www.trumpf-laser.com/en/solutions/applications/laser-cutting.html (Top Row Left) Laser cut plywood and delrin gears showing tool marks of the laser like the burnt cutting surfaces and the vertical lines on the cutting surfaces. Also the taper created by the laser can be observed. (Top Right and Below) Cutting plywood parts using the laser cutter.

Figure 3.38 Fused deposition modeling (FDM) utilizes small nozzles through which molten plastic is extruded. Layer by layer this molten plastic bonds to itself and the layer below. A second nozzle extrudes support material which serves as scaffolding and can easily be removed afterwards. Source: http://www.pddblog.com/tag/fused-deposition-modeling/

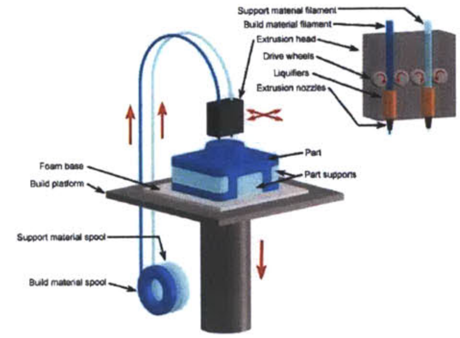


Figure 3.39 Selective Laser Sintering (SLS) and Direct Metal Laser Sintering (DMLS) is a process where powder from a reservoir is spread out in a thin layer while a laser melts the powder particles into a solid. The next layer is applied and the powder melts to the layer underneath and itself. This method does not require a different support material. Source: http://www.pddblog.com/tag/fused-deposition-modeling/

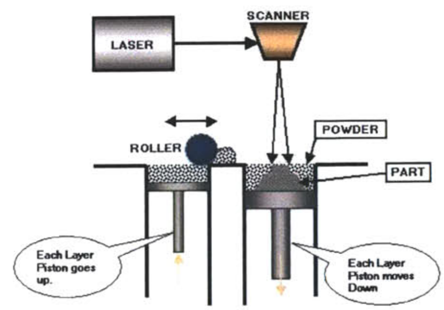


Figure 3.40 Stereo Lithography (SLA) depends on specially developed UV-curable resins. Typically a laser source around 350-365 NM is required to cure the resin. The laser energy needed for curing the resin is lower than with SLS. Recently digital light processing DLP projectors are used to expose a full layer a time. Once a layer is cured the build bed drops down allowing new resin to flow on top where it will be cured into the next layer. Source: http://www.pddblog.com/tag/fused-deposition-modeling/

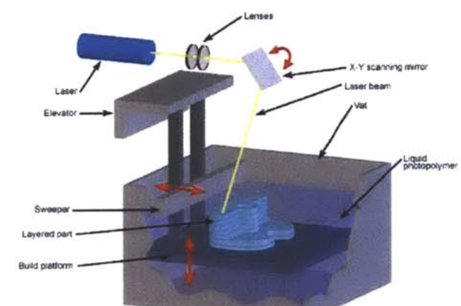


Figure 3.41 PolyJet/ProJet is similar to SLA. Here a photo curable resin is printed through an ink yet printer head. Once a layer is layed down a flash light photo cures the resin. A wax support material is printed also using ink jet printer heads and serves as the support material. The support material can be removed after the print finishes by dissolving it in an ultrasonic water bath. Source: http://www.me.vt.edu/dreams/Facility/Machines/MachinesObjet.html

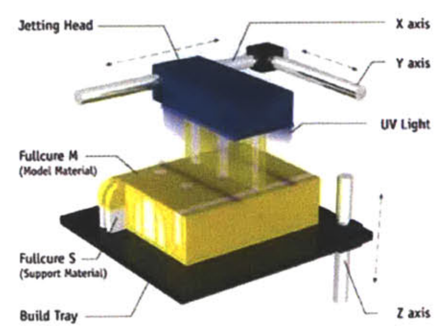
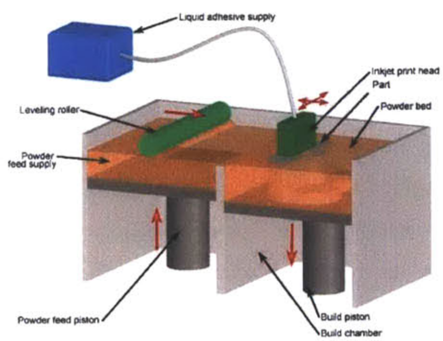


Figure 3.42 ZCorp has patented a 3D printing method using an ink jet printer head to depose glue like binder onto a plaster like powder. This process is similar to the SLS method but is based upon adhesive forces keeping the powder like material together rather than melting it into a solid. One advantage in this method is that the glue like binder can be pigmented just like ink used for ink jet printing which results in full color 3D prints. Source: http://www.pddblog.com/tag/fused-deposition-modeling/



3D Printers

Instead of subtracting material like a mill or a laser cutter a 3D printer belongs to the domain of additive machines. A print head or laser beam is positioned using CNC actuation in order to deposit material which is then fused, glued, or sintered. The adding of material happens layer by layer as shown in the adjacent images. The layer thickness can be as small as some microns depending on the printing method. Support material is needed in some cases in order to scaffold complex shapes. There are five major methods for 3D printing: (1) fused deposition modeling (FDM), (2) laser sintering (SLS) and direct metal laser sintering (DMLS), (3) stereo lithography (SLA), (4) PolyJet and ProJet and (5) the method developed by ZCorp where an ink jet print head infuses a bider into a plaster like powder. Depending on the requirements of the 3D printed parts different methods have different advantages. The most robust method is laser sintering and direct metal laser sintering. The finest details are created using PolyJet ProJet and stereo lithography. The cheapest method is the ZCorp.

In general 3D printing and rapid manufacturing which utilizes 3D printing has several advantages over conventional manufacturing methods. Shapes which are otherwise impossible to make can easily be 3D printed. An example would be a part inside of a part inside of a part or filigree bone interior meshes, patterns and ornaments in any 3 dimensional configuration. Also inside surfaces or cores for casting can be made together with the outside surface.

Another advantage is no up front tooling costs. While injection molding requires tools to be made and machines to be specialized which results in large quantities of parts to be made 3D printing can do any part at any number without affecting the costs in the same way as conventional manufacturing methods. Assembly of products can also be positively effected as the number of components for a product will go down using 3D printing. It is actually possible to create products which are completely assembled by a 3D printer as shown in the example of the 3DprintedClock later in this chapter.

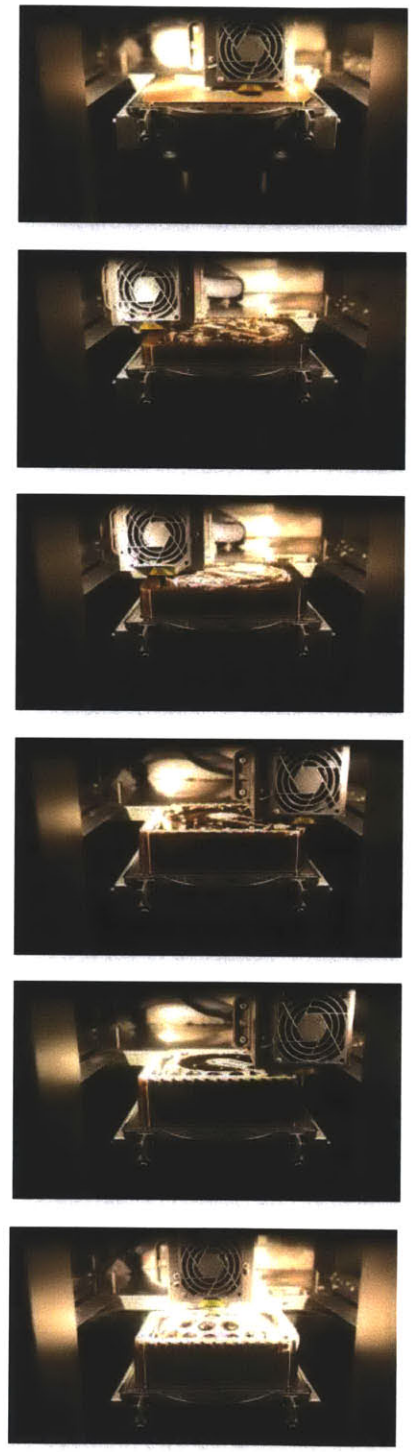


Figure 3.43 (Above) Dimension Elite (FDM) printing the 3DprintedClock layer by layer while inserting support material in between separate parts and around parts in need of support.

One last advantage to mention is the capability of overcoming mass manufactured products towards customized ones in combination with new expressive qualities resulting from advantages mentioned above. Symmetry is no longer a requirement for products and seams resulting from breaking a product down into manufacturable instances will disappear. At the same time user control over form, shape, material, color even functionality increases.

Today 3D printing is limited to one material at a time which will create homogeneous parts in terms of material, color and surface finish. Also parts will be created in homogenous ways meaning the precision of the printer will be the same throughout the printing process while some parts might only need to be precise in certain areas. One way of addressing efficiency involved with 3D printing involves current development of algorithms replacing solid sections in 3D printed parts with bone-like interior mesh structures to save material and weight while maintaining structural stability. Also in development are methods of post processing 3D printed parts towards a more predictable and homogenous behavior regarding failure and breakage which becomes a key topic for more integrated parts and products. With 3D printers developing to become cheaper and more available the paradigm of "built to last" can be reframed as "rebuild on demand" where breakage no longer renders a part useless. Instead, it can break and be replaced immediately. One could even think further towards a kind of living and breathing relation of material and its formal incarnation where on a daily bases the same material is one part today and a different part tomorrow.

Components

Making Gears

Gears span open a wide domain within which I focus on the most common types of involute spur and helical gears. The images below show various explorations using laser cut plywood gears adjacent to laser cut delrin gears, 3d printed helical and stacked spur gears and some meshing examples. The overall performance of the gears and assemblies created cannot yet be compared to conventional state of the art gears and power transmission assemblies. Especially high performance and efficiency applications are out of reach in the context of materials like plywood and plastic. Nevertheless the freedom gained in regards to integrating gearboxes with more holistic assemblies while controlling the degree of integration and visual appearance is significant. A new field of applications is spanned open located between professional and industrial high performance and efficiency applications and the very low end of the spectrum where gears and bearings would not be used because they are expensive, inaccessible and complex. However, not every mechanical project needs to lift heavy weights or move as fast as possible leaving lots of space to be explored by digital fabrication tools offering professional solutions like gears, bearings and

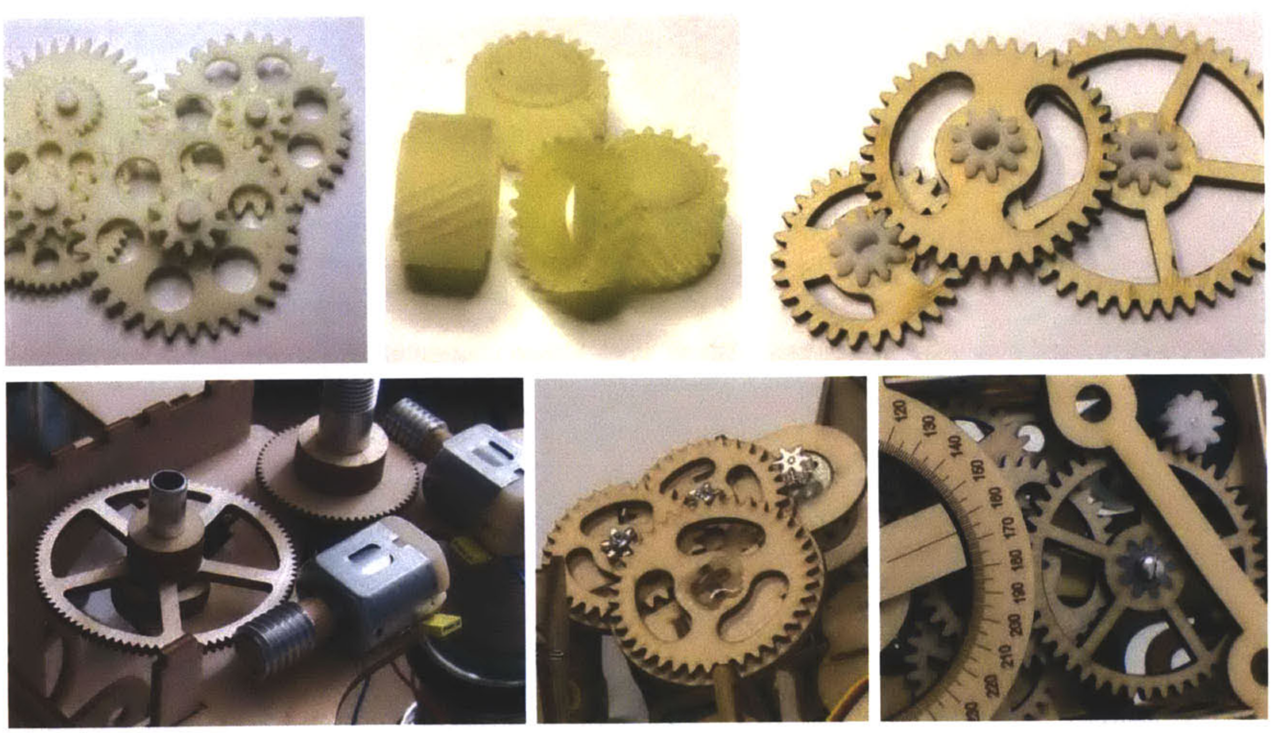


Figure 3.44 (Below) Various gears created on laser cutter and 3D printer (Bottom Row) applications and meshing examples. Involute spur gear and involute helical gear.

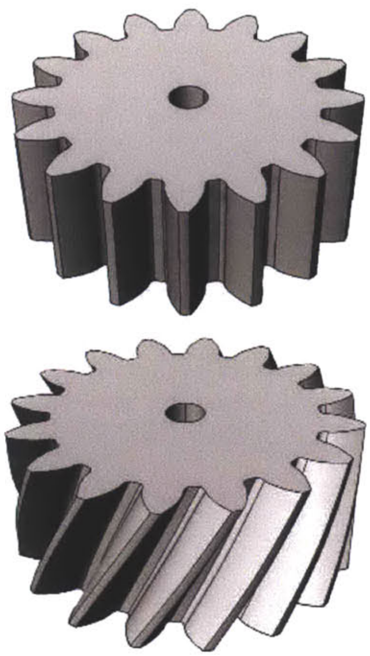


Figure 3.45 (Above) straight spur gear and helical gear

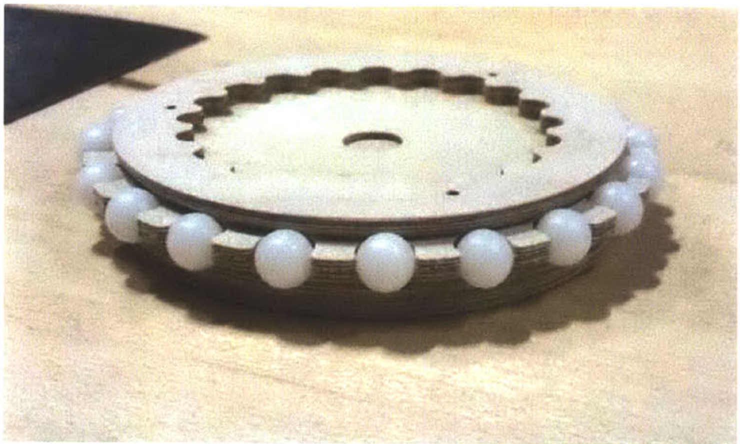
power transmissions at an almost as high level as professional industrial solutions.

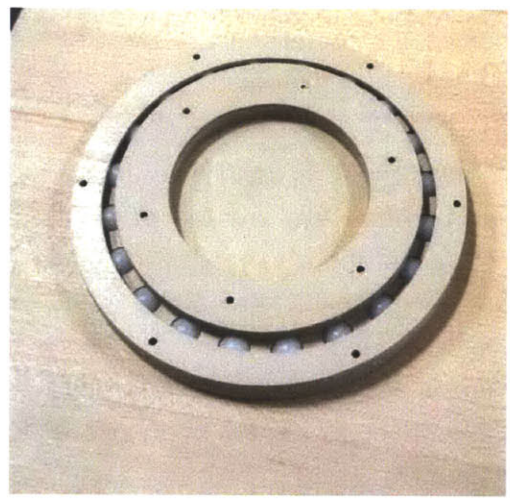
Gears basically consist of a circular outside featured with teeth meant to mesh with a neighboring gear. This allows for rotation to be transmitted from one gear to another but also to be conditioned in terms of rotation ratio and torque. When gears with the same number of teeth mesh with each other both rotate at the same ratio. If one gear has more or less teeth than the other one the rotation ratio changes. A gear with double as many teeth as another one will need two rotations of the smaller one in order to be fully turned or vice versa the gear with double as many teeth will turn the smaller gear twice. Together with this rotation ratio change comes a change in torque. Increasing the gear diameter also increases the lever arm between the contact point with the small gear and the center of the one twice as large. Compared to the smaller gear the larger one will have half the rotation but double the torque of the smaller gear. These are the main two aspects important for understanding gearboxes.

Gears are used for power transmission purposes which become necessary for various reasons. One reason can be found looking into the source of actuation which often is some sort or motor. Motors no matter whether combustion or electric rotate at a rate related to the nature of their working principle which unfortunately does not match most of their applications like powering vehicles. Power transmission becomes necessary to speed up the revolutions or reduce them and make the motor input more or less powerfull. All of this happens in gearboxes which are explained in more detail below. A second reason for power transmission is related to spatial configuration. For economical reasons only one power source is used in a vehicle to power multiple outputs like two wheels which requires routing the power from the source to the desired output location involving power transmissions. Or for example aircraft wing and flap actuation where for weight distribution reasons the power source cannot be located next to the actuators.

Gears have to roll on each other rather than slide which would cause huge friction, heat and power losses. For this reason gear teeth have to be designed in a special way relating to their size and diameter. The involute curve of the gear's pitch circle is used to create the specific tooth shape. Only one half of one tooth needs to be created this way and can be mirrored and circularly arrayed around the pitch circle. Chapter 4 will describe a tool to create gears which can then be laser cut or 3D printed. The resulting gear outline can then be extruded in a straight way and will form a spur gear. A more optimized version of the spur gear is the helical gear where the gear outline is not extruded straight but with an angle twisting the gear around its axle. This change increased the length of the contact line between gears and across teeth which makes meshing and rolling smoother and more quiet. Helical gears are more difficult to make in a conventional way compared to spur gears which makes them rare. 3D printing offers the opportunity to overcome the difficulties involved with making helical gears.

Milling gears is definitively a good way for making gears as the end mill always created perfectly straight and smooth cutting surfaces and a variety of materials are available for milling. The only set back is related to the end mill diameter driven minimum feature detail which determines the size of the milled gear. Often times laser cutting and 3D printing will create smaller gears with the additional benefit of an increased ease of making.





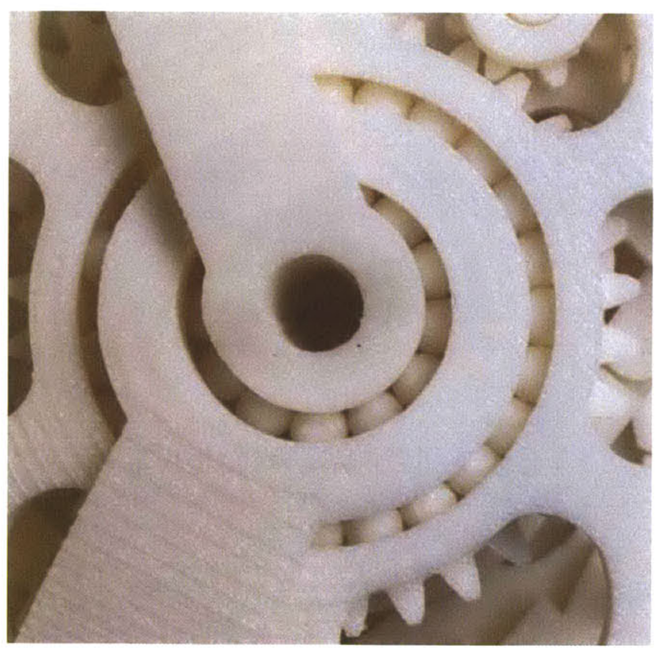


Figure 3.46 (All page) Various ball bearings, four-point contact bearings and crossed roller bearings made from milled plywood and 3D printed plastic.

Making Bearings

Bearings have much in common with gears as they also try to reduce friction and slippage by introducing rolling motions. In the case of a bearing balls, rollers and needles are put in between a rotating shaft and its static housing to allow the shaft to rotate smoothly, precisely and at low friction. Bearings are precision parts requiring an inner and outer raceway for the balls which has to be exactly round and provides only one or two contact points with the balls. The balls have to be precisely round too. If the raceway is simply constructed from an arch of the balls outside diameter the bearing will not work properly as the balls slide and rub on the raceway rather than roll. As shown in the adjacent Fig. 3.47 the raceway should provide either one contact point on each raceway side or two symmetrical contact points on each raceway thus resulting in a four-point contact bearing. These contact points will be the equivalent of rails upon which the balls run and spin. With a roller or needle bearing this contact point can become a line riding on the outside of the roller or needle thus increasing the load bearing capacities of the bearing. All the forces applied to the bearing will be transmitted through the contact points causing the balls to be compressed and to press into the raceway. Conventionally very hard materials are used for both the balls and the raceways. With a wooden or plastic bearing the way to maximize the load bearing capacity and prevent run out of the raceways is to insert as many balls as possible to increase contact area. Conventionally this is a problem as a bearing is made from only two parts, an outer and an inner ring in between which the balls are assembled. The assembly process is fairly simple by just positioning both parts towards each other so they touch which will cause a large gap on the opposite side. This gap can be filled with as many balls as it fits. In the next step the balls will be spread out while simultaneously moving the bearing parts into their final position. Once the balls are spread out evenly a retainer is inserted to keep the balls in their position. With wooden and plastic bearings as shown in the adjacent images it is recommendable splitting a plywood bearing in more than two parts or creating a feeding channel with a 3D printed bearing which allows the raceway to be filled completley with balls. Using a V-grove cutter on a milling machine creates perfect 45-degree tapered edges which can

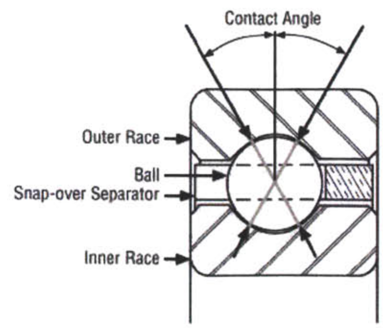


Figure 3.47 Section of a four-point contact bearing. Source: http://www.kaydonbearings.com/

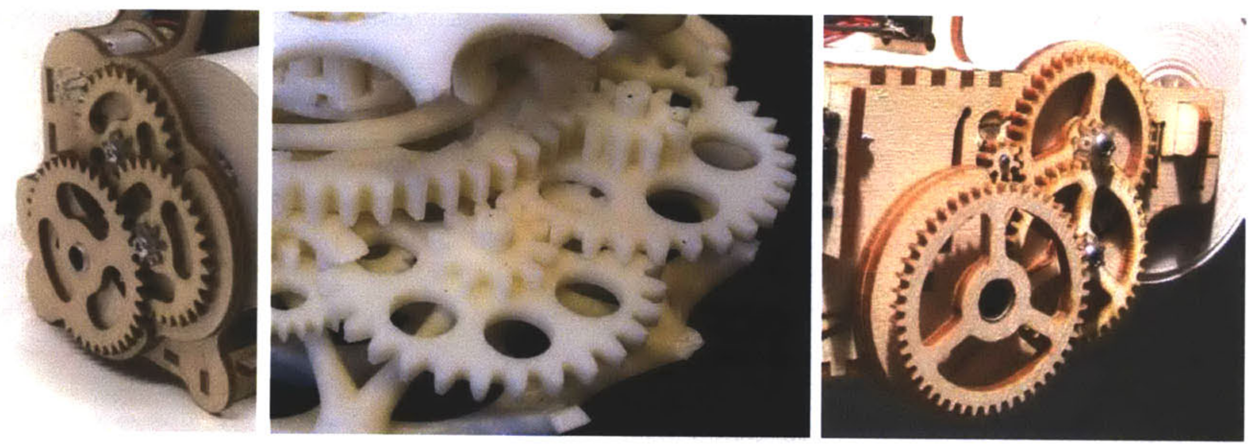
be used as raceways for a plywood four-point contact bearing. Sealing a feeding channel with a custom cork for a 3D printed bearing also temerged as a method for assembling bearings with as many balls as possible.

Unless printed in an already assembled ("fab-sembled") configuration all the steps described above require manual assembly. Ball insertion tools can be developed to increase easy of assembly while the necessity of assembling 3D printed integrated components at the seam of their bearings is a good choice and should always be maintained. Looking into component boundaries of products involving movable parts the bearing is the minimum mutual boundary such a product or assembly must have. While gears can be printed fab-sembled for applications neglecting the mechanical backlash, bearings should not be compromised in this way. A gearbox involving backlash can still drive an output while a bearing involving backlash makes it simply impossible to be operated especially inside a backlashing gear box. The precision of 3D printers is sufficient to create good raceways while printing the balls at the same time always involves a gap depending on the printers precision thus compromising the ball bearing. Utilizing precision raceways to insert precision balls seems to provide the most significant benefits. Avoiding extensive and complicated manual assembly in combination with post ball insertion appears to be a suitable hybrid approach towards 3D printed bearings.

Making Gear Boxes

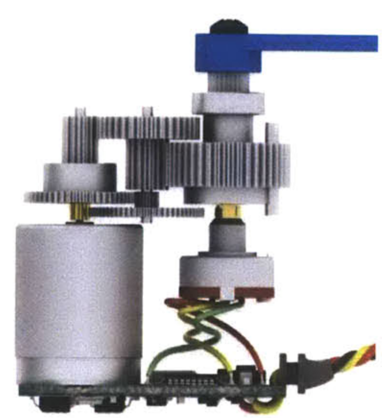
Gearboxes as already mentioned above fulfill the purpose of transmitting and conditioning mechanical force and rotation. The most common types of gear boxes include staggered super gears, planetary gear boxes, worm drives, cycloidal gear reducers and harmonic drive gear reducers. They all mainly serve the purpose of reducing a motor's rotational speed while increasing torque. The cycloidal and harmonic gear reducers offer high precision, high torque and small volume. They cannot be back driven by an outside load which qualifies them for robotic and other position control applications. Staggered spur gear boxes and planetary gear boxes are the most common ones and can be found in cars and power tools.

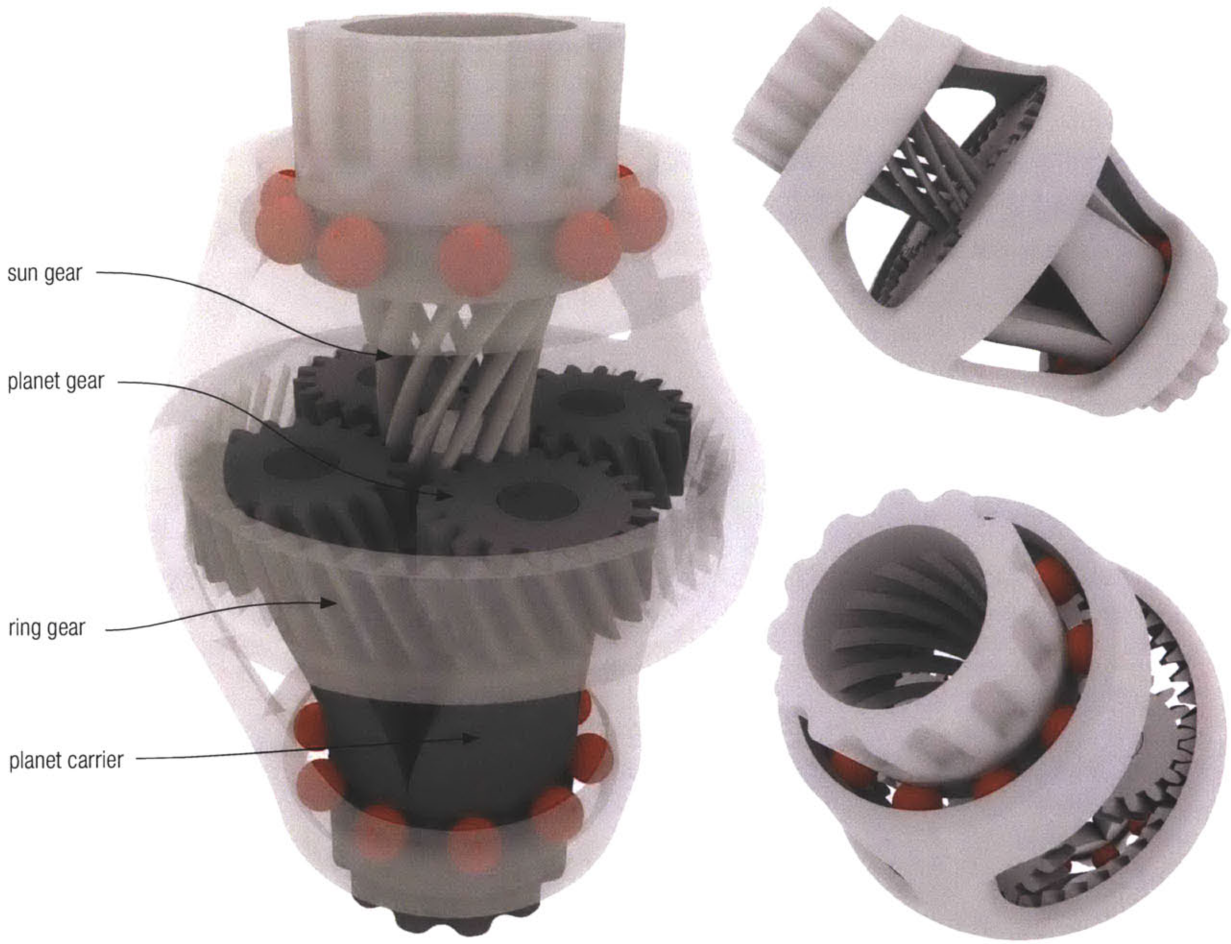
In a staggered spur gear box like shown in Fig. 3.59 an input shaft is geared through multiple stages into an output shaft. Each stage consists of two gears with the same module (size) and pressure angle so that they mesh with each other. Each stage's output gear is connected to the input gear of the next stage which could be of the same module or one step higher in terms of its module as the torque increases. The images in Fig 3.48 show plywood and 3D printed plastic examples of staggered spur gear boxes. Chapter 4 also discusses the topic of staggered spur gearboxes and specifically creating 3D printable staggered gears.



In a planetary gear box a sun gear is located at the center surrounded by two, three or four planet gears which themselves are surrounded by the ring gear. All gears mesh with each other. The planets are connected by a planet carrier. This arrangement allows for three different gear ratios depending on which part is used as input, output and stationary. The most common case allowing for the highest gear ratio is where the sun is driver as the input, the ring gear is stationary and the planet carrier is the output which might be connected to a second stage sun gear. In this case the gear reduction is calculated by number of teeth in the ring gear divided by number of gears on the sun gear plus one. Staggered planetary gearboxes achieve a high reduction ratio to volume which makes them a preferred choice over staggered spur gears boxes. They are used in a variety of applications spanning cordless power drills, vehicles and robotics.

Figure 3.48 (Above) Staggered spur gear boxes in plywood and 3D printed plastic using the FDM printer. The plywood gears have small plastic gears in their centers which have been press fitted by flipping the gears to take advantage of the laser cutter related taper. (Below) CG rendering of the HSR-5995TG servo





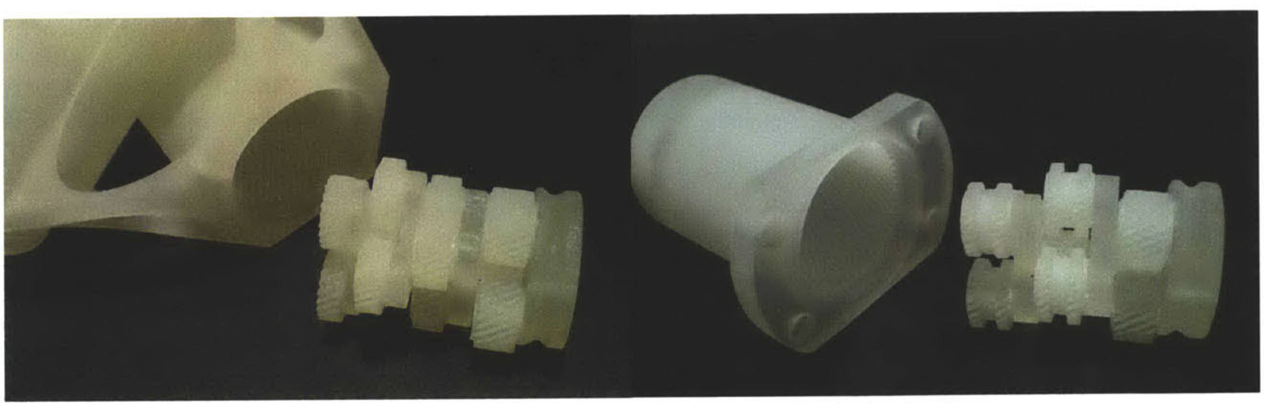
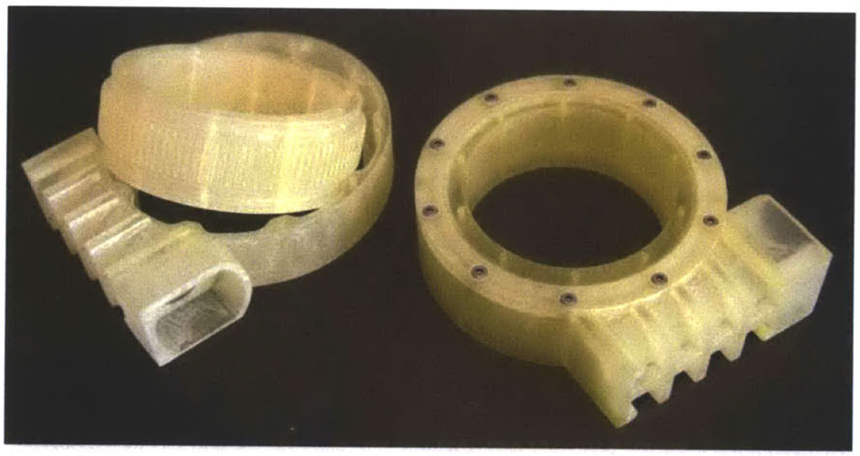


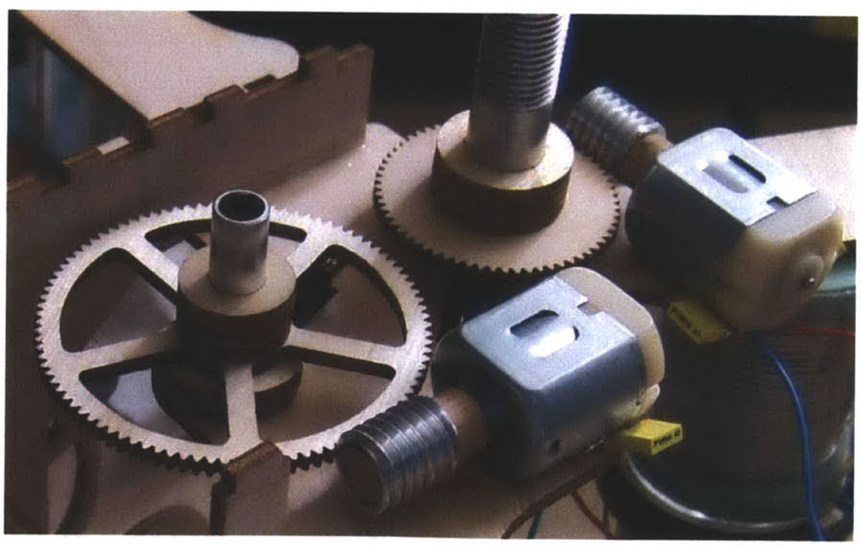
Figure 3.49 (Above) CG rendering of planetary gear box intended for 3D printing and demonstrating a dynamic and form oriented integration of the planetary concept. Photos of two staggered planetary gear boxes with 3 and 4 stages utilizing helical gears.

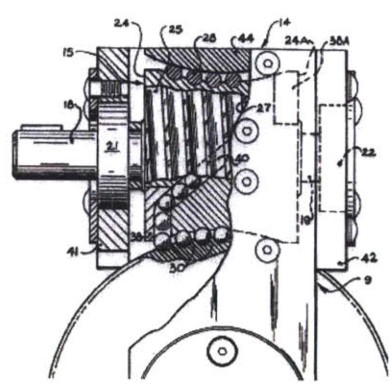
In a worm drive the output gear is driven by a screw like helix (the worm) oriented tangentially to the gear resulting in one revolution of the helix pushing the gear over one tooth. This operating principle results in the gear ratio being the number of teeth on the output gear. Worm drives have a high reduction ratio and can create and withstand huge torques, they are not back drivable and qualify for handling heavy loads. Unfortunately due to the perpendicular sliding motion huge friction is created between the worm and the output gear which effects the overall efficiency of the drive.

The idea of a recirculating ball worm drive has been developed in the 1960s and several patents exist on these designs though the oldest ones have already expired. The ball worm drive introduces balls in between the contact surfaces of the worm and the output gear, exactly where the most friction is created. This friction is translated into a rolling motion which would make the ball worm drive hugely attractive for all types of applications. Difficulties manufacturing the ball worm drive have prevented it from being used widely. 3D printing can overcome exactly these problems potentially revitalizing this almost dead evolutionary branch of the worm drive species.

Figure 3.50 (Below) CG rendering and 3D print of a double enveloping worm drive. (Lower Left) Patent drawing of a recirculating ball worm drive with improved efficiency. Patent # 3468179, 1969 by R. K. Sedgwick







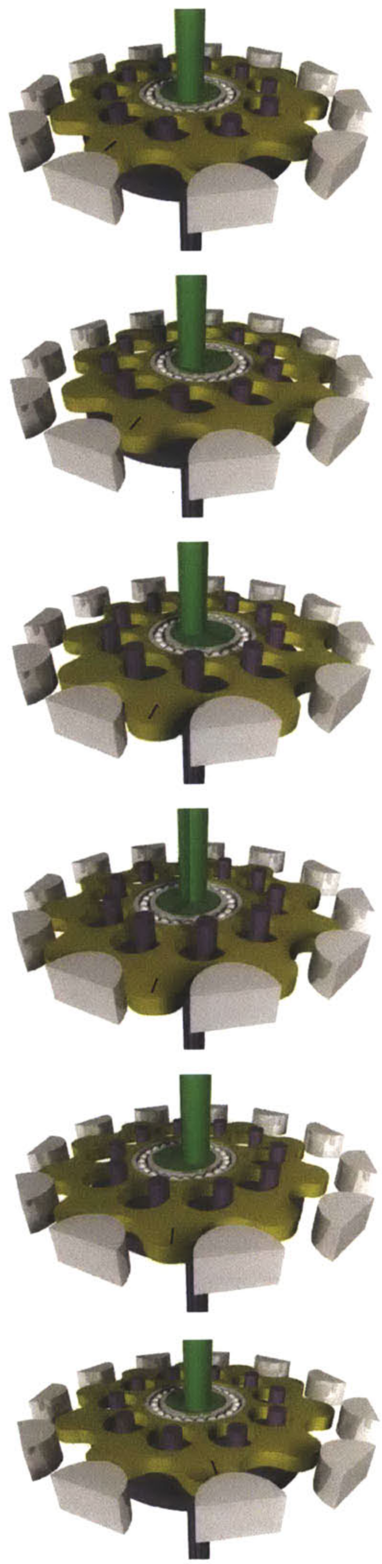


Figure 3.51 (Above) Cycloidal reducer working principle tracing the movement of one tooth for a full input shaft rotation. Source: wikipedia.com

A cycloidal gear reducer has at its center an eccentric rotating input shaft. Through a bearing this eccentric motion drives a cam disc whose outside perimeter is made from a certain number of arches all in contact with cylinders surrounding the cam disc. There is one more cylinder surrounding the cam disc than there are arches on the disc. The arches are the hypocycloid curve resulting from the number of surrounding cylinders, their diameter, the eccentricity and the pressure angle between the cam disc and the cylinders. The hypocycloidal outside of the cam disc will roll on the surrounding cylinder and also engage in a spinning motion causing it to turn around its center as the result from having one hypocycloid less than surrounding cylinders. Every time the input shaft has rotated as often as there are surrounding cylinders the cam disc will have gone through one full rotation (Fig. 51). In a single stage setup the cam disc is the output of the reducer requiring an output shaft connecting to it while accounting for the eccentric motion. A second disc serves as output solving the eccentric motion compensation through pins connecting into the cam disc while the hole's diameter is enlarged by the eccentricity. A dual stage cycloidal reducer is possible where the cam disc connects to a second cam disc. This second cam disc reverses the relation of driver and driven and different from the first stage drives the surrounding cylinders of the second stage which now becomes the output. This type of 2 stage cycloidal reducer can achieve a wide range of reduction ratio options with only 3 movable parts involved which is an unmatched characteristic among all gear reducer principles. The eccentric motion in every cycloidal reducer causes vibration which presents difficulties for some applications. In order to cancel the vibration a second cam disc is used within a one stage cycloidal reducer. This second cam disc and its eccentric center shaft are positioned counter phase wise to balance the overall gear reducer setup. Cycloidal reducers are characterized by their high reduction ratio, high torque and impact strength, precise operation, non back drivable and low maintenance properties. They can be found in various applications where heavy and long lasting actuation is required like industrial automation, robotics and power generation.

The adjacent images Fig. 3.52 show various cycloidal reducers built in plywood and 3D printed plastic. The 3 part design and its resulting simplicity makes this type of gear box especially attractive for digital fabrication. The CNC router is the preferred tool as it creates straight cutting surfaces which support tight tolerances and precise operation. 3D printing combines precision parts with the additional advantage of ease of making as fewer steps also regarding manual labor are involved.

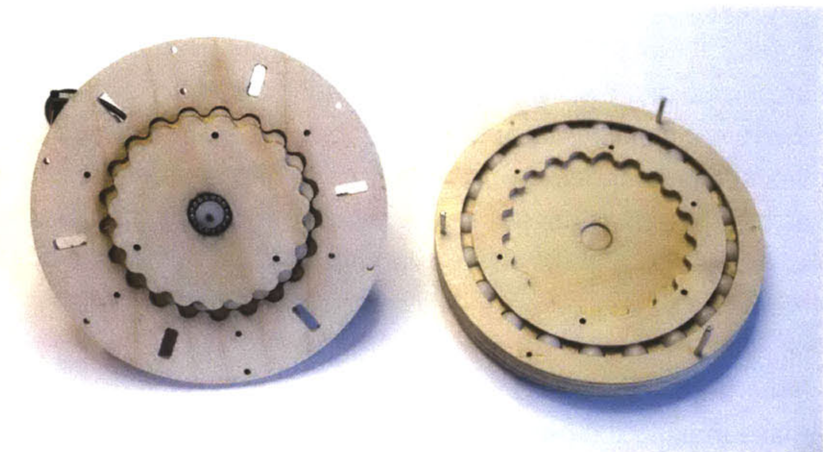
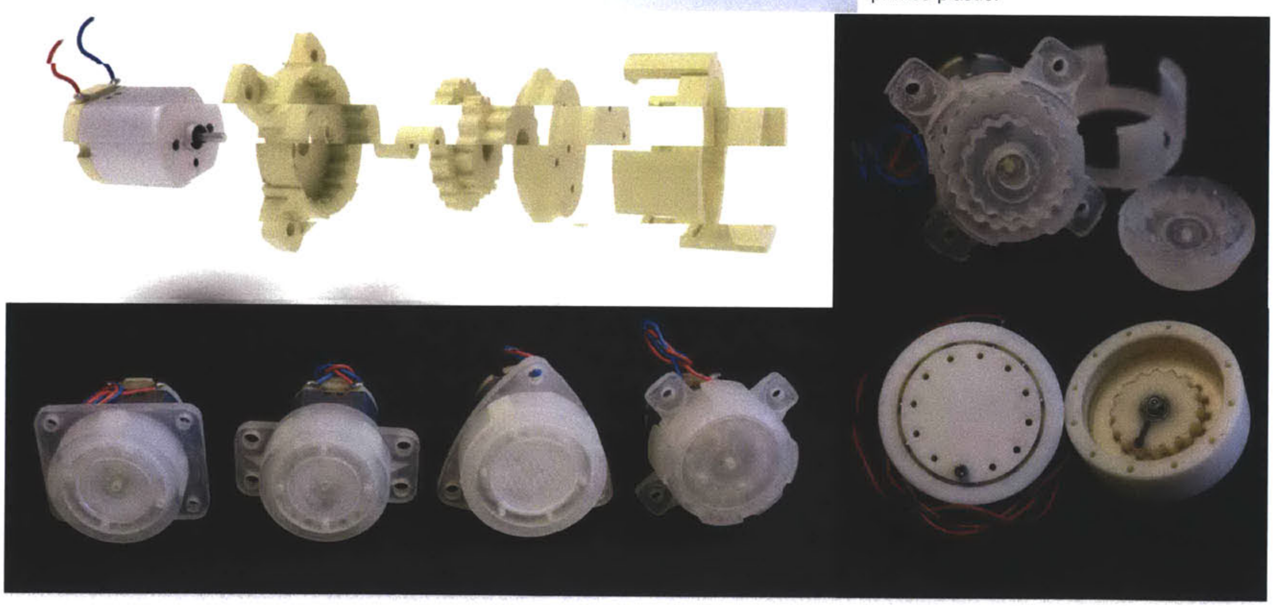


Figure 3.52 (Below) Various cycloidal reducer rendered in plywood and 3D printed plastic.



Harmonic drives or strain wave gears have three to four components. The static spline which is similar to the internal ring gear of the planetary gear box. The flex spline which is a flexible ring gear rolling on the inside of the static spline while having two or sometimes three teeth less than the static spline. The wave generator which deforms the flex spline into an ellipse touching the static spline in two points and allowing for rotation which will cause every flex spline tooth to roll on every static spline tooth. As the flex spline is short two teeth the rotation will cause the flex spline to move relative to the static spline (Fig. 3.53). The ratio of this gear reduction is calculated by flex spline teeth minus static spline teeth divided by flex spline teeth. The output of the harmonic drive is the flex spline as it moves relative to the static spline which is the stationary part. In the long version of the harmonic drive the flex spline is extruded into a cup allowing for sufficient flexibility. The flat bottom of the cup can be connected as output. In the pancake version an additional part is introduced: the dynamic spline. It is located above the static spline and the flex spline connects over both of them. The dynamic spline has as many teeth as the flex spline and is grabbed by the flex spline to create the output. This design allows for a flat version of the harmonic drive. The wave generator is often made from a special elliptical ball bearing preventing the flex spline to buckle under heavy load. However, this elliptical ball bearing is very difficult to make making the harmonic drive less common.

Other wave generator designs include a solution where a planetary gearing principle is used involving two planets to deform the flex spline. In this case, the gear ratio of the harmonic drive can be multiplied by the planetary gear ratio which can enlarge the overall ratio tenfold. In addition the planets can be pre-loaded and given a slight flexibility. The preload will result in canceling any mechanical backlash as the planets will press against the sun in the center and the flex spline (static spline) on the outside. A slight flexibility will result in dampening any unevenness involved with machining the harmonic gearing parts. Thus the precision of the harmonic drive with planetary wave generator does not result from the precision of machining or printing the components like with all other gear boxes mentioned above but adjusts and corrects itself by design. This property gives the harmonic drive with planetary wave generator a significant advantage especially for digital fabrication processes. Though the cycloidal reducer has fewer movable parts, the general operational principle and output precision relies on precise machined parts which increases the risk of not functioning properly in the context of digital fabrication.

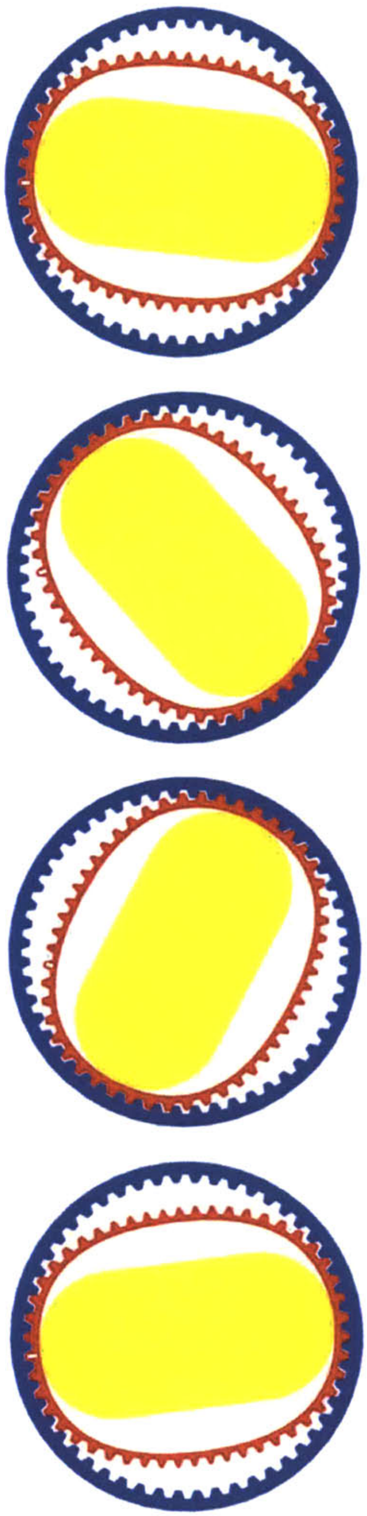


Figure 3.53 (Above) harmonic drive working principle tracing the movement of one tooth for one full rotation of the wave generator. Source: wikipedia.com

Fig. 3.54 shows various pancake harmonic drives made using 3D printing. The PolyJet machine creates the highest quality results in terms of smoothness of operation and gear noise. Also the small feature detail of the PolyJet made it possible to create the teeth and bearing raceways. The earlier version using metal pins also functioned but due to the round pins the output spline would slip against the flex spline at higher torque.

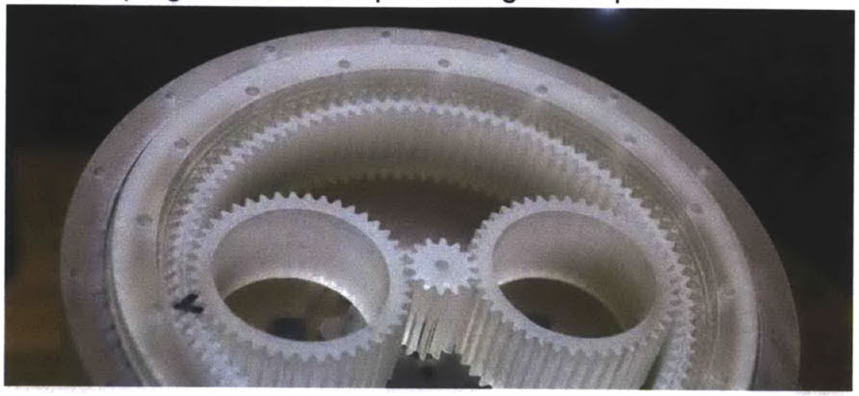
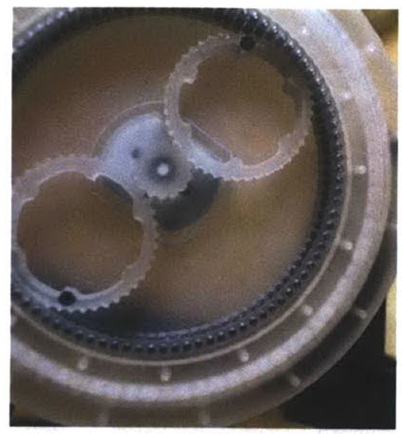
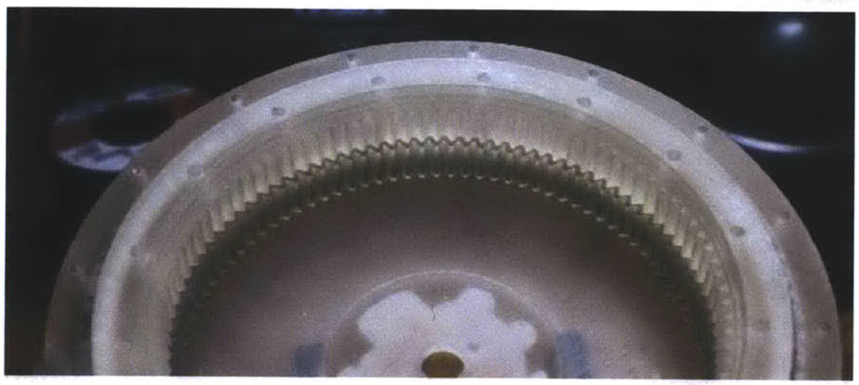
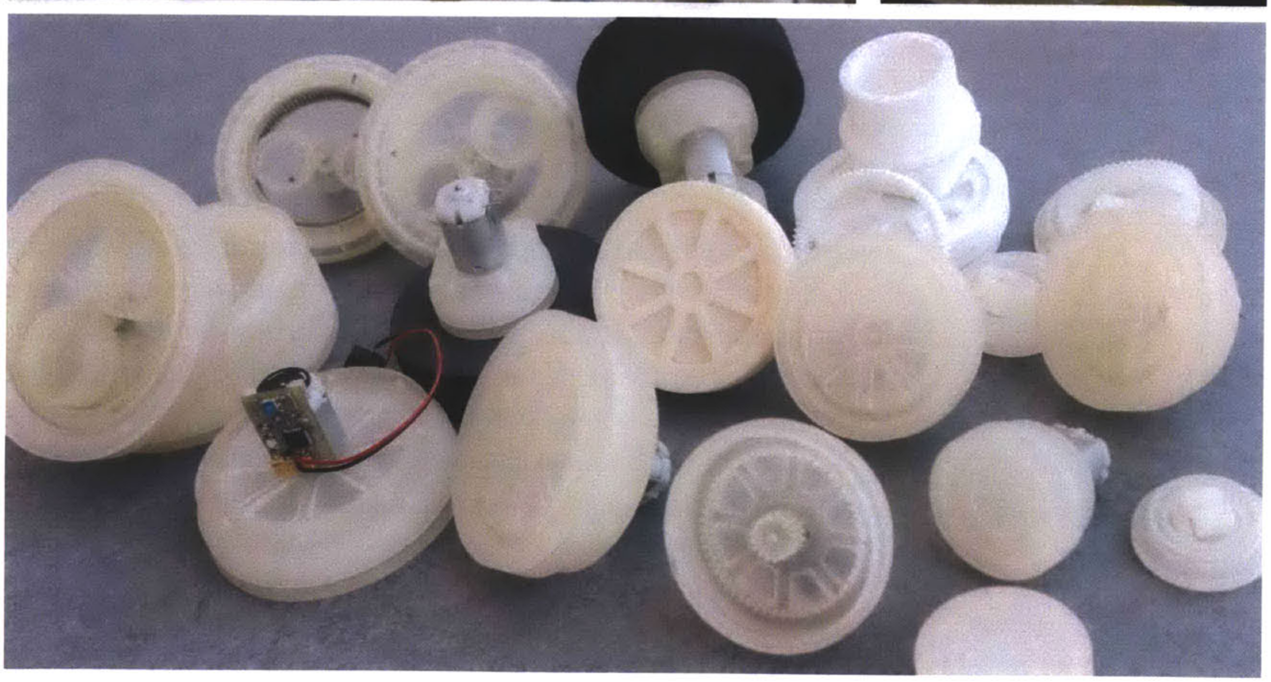


Figure 3.54 (Below) Various pancake harmonic drives 3D printed in plastic using the planetary wave generator design.



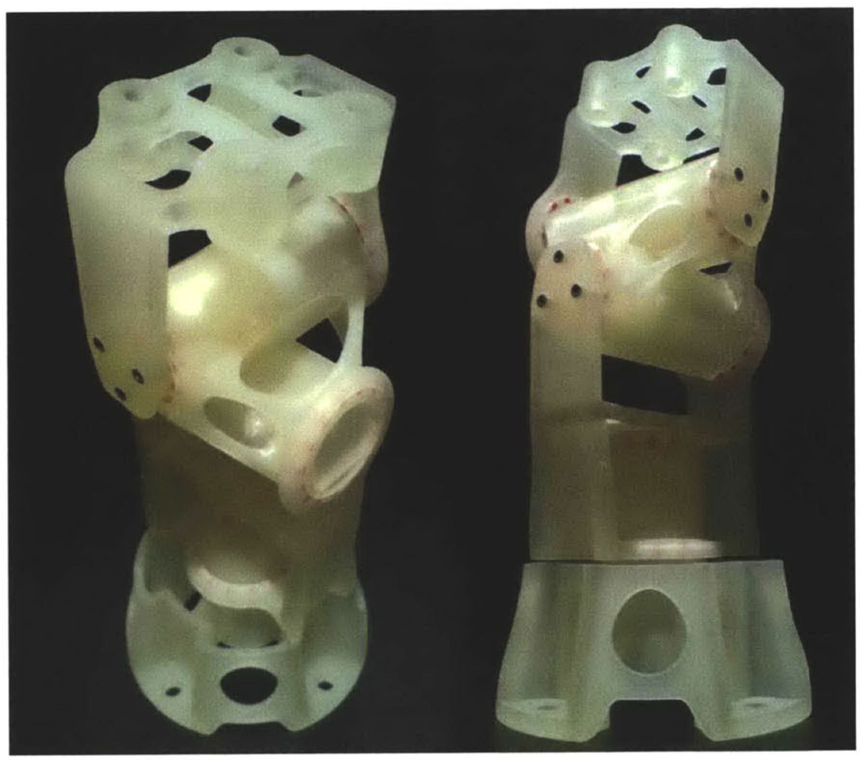




Integrated Assemblies

3DoF head

A three-degree of freedom (DoF) head is shown in Figure 3.55. This design exploration focused on creating some kind of mechanism to be connected to a base plate while actuating an end-effector in three directions like a tripod head to pan, tilt and rotate a camera. The goal was to integrate power transmission elements like the gear boxes and bearings in a fluid way. Using 3D-printing enabled helical gear teeth, in which each tooth is not



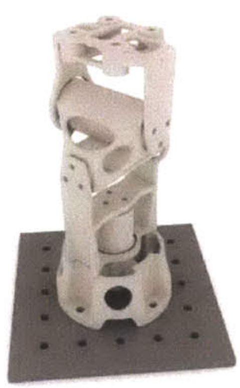
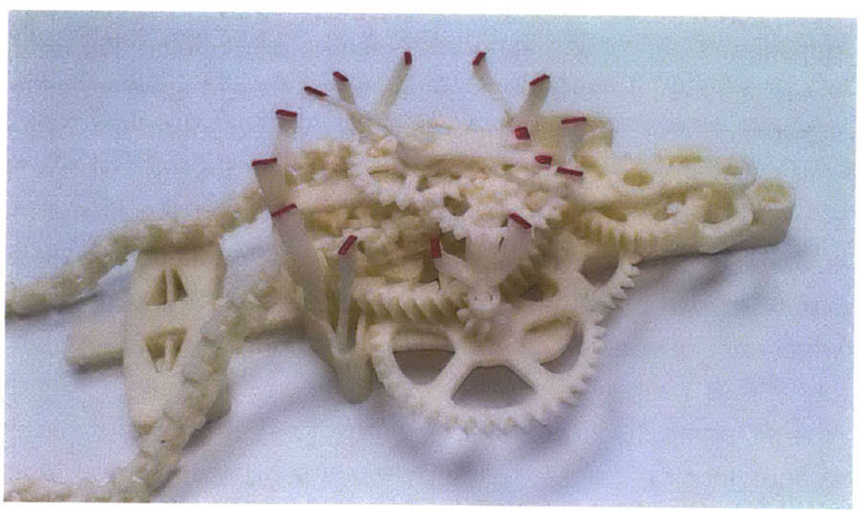
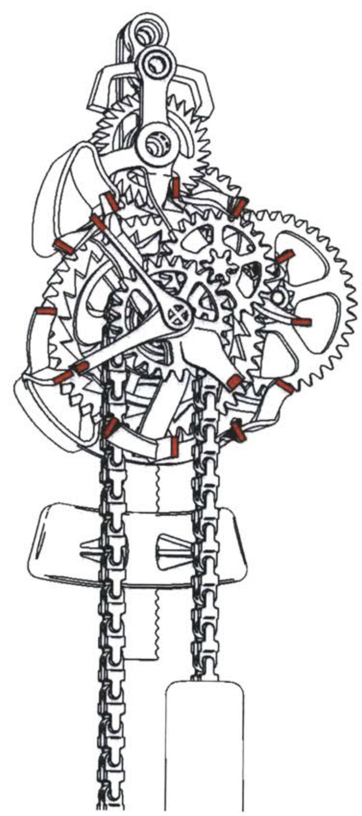


Figure 3.55 (Above and Right) The 3DoF head and its iterative development steps tracing various formal options.

extruded straight in order to form the gear but follows a helical path (upper right). The advantages are smoothness, efficient and quiet operation. The raceways for the bearings were printed into the parts and components. The assembly of the bearings followed the normal procedure in which both parts are positioned towards each other while touching in one point thus making it possible to feed in a certain number of balls. When the cavity has been filled with balls, both parts of the bearing are moved into their final position while simultaneously moving and distributing the balls around the circumference. Once everything is in place a ball retainer is inserted to fix the positions of the balls thus holding the bearing together. In this case assembling the bearings used the same procedure as assembling the mechanism. Only one connection per DoF required to be manually screwed together. The overall experience with this exploration was good. The helical gears worked well in their planetary configuration with a 625:1 reduction. The bearings also functioned as expected. Unfortunately the 3D printing material shrank and did not allow the motors to slide into the position as required restricting the machine to manual operation. Another 3D printing related issue was the feature detail. The bearing retainer ring became very fragile as its size was too small for the minimum feature detail. Overall this exploration mainly supports the feasibility of near-complete assemblies introduced above and proves the integartion of overall shape with subcomponents like bearinsg and gear reducer.

3DprintedClock

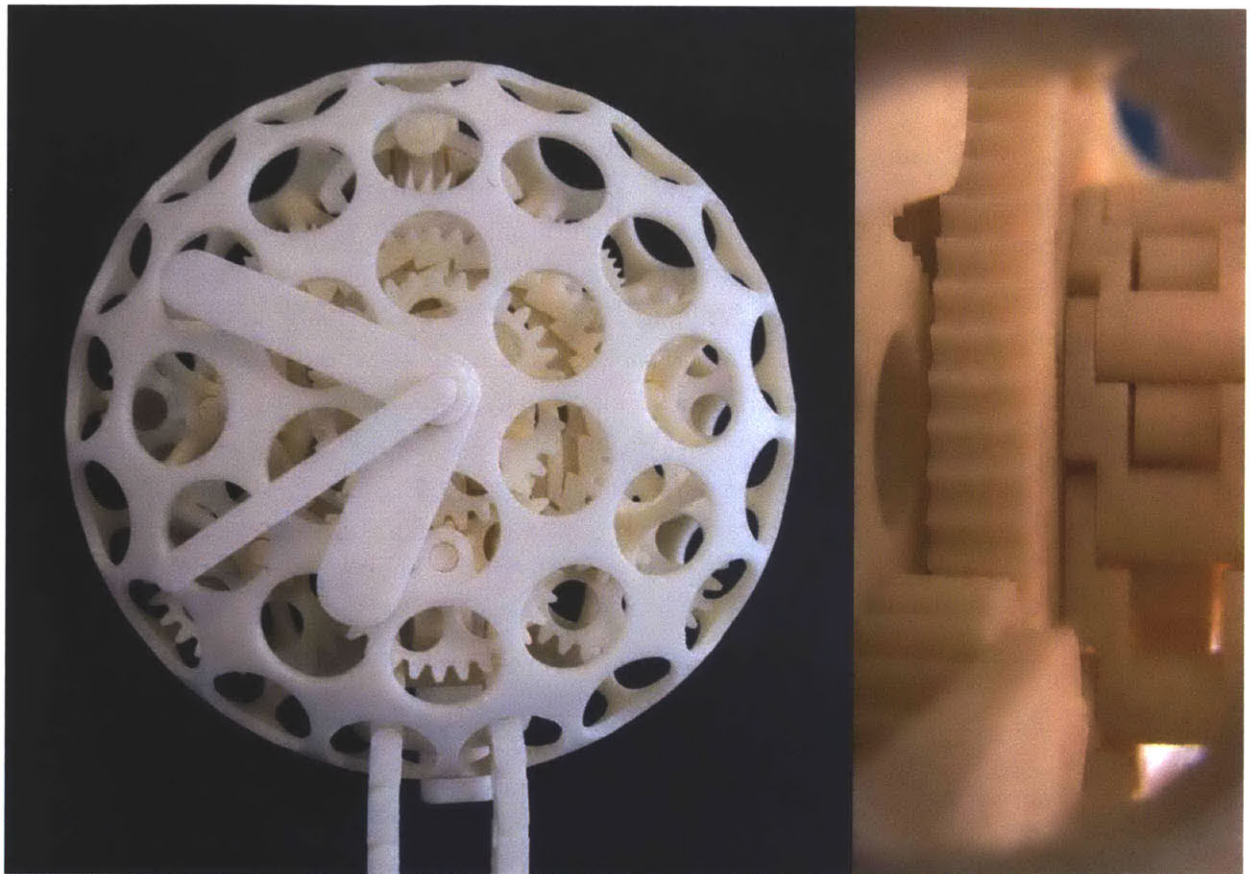




The 3DprintedClock (in collaboration with Bob Swartz, MIT Media Lab) presents a fully functional clock using a pendulum and descending weight to keep track of time. It was modeled in CAD software after an existing clock while ensuring gaps and clearances in between its components matched 3D printer specifications. The entire clock was 3D printed in one piece with the exception of the metal weight. A 3D printed container meant

Figure 3.56 (Above and Left) The first iteration of the 3DprintedClock. The gear support frame is driven by the spacial gear arrangement and develops the clock face by branching out. Red color is used to identify the clock face and the hands to be filled with sand or water could replace the metal weight but for build size requirements the metal weight solution was chosen. The CAD model also included drainage holes and channels to fully remove the support material after the print. After the printing process the support material has to be removed before the clock can be mounted on a wall. Once the metal weight has been added it will start ticking. The design underwent several iterations staring with the first version having an open gear train and frame while the following iterations enclosed the gear train within the casing to demonstrate the gears being created within the case to emphasize the lack of any post-printing assembly. The biggest challenge throughout the design process presented itself in the form of friction between the 3D printed parts. Ball bearings were added to the main axle to ensure all force created by the weight is be transmitted through the gear box to the escapement mechanism.

The 3DprintedClock impressively demonstrates the state and potential of current 3D printing and rapid manufacturing technology as well as the benefits of integrating form and object qualities in complex mechanical assemblies. 3D printing holds the promise of superior capabilities compared to state of the art fabrication processes like injection molding or stamping sheet metal. No up front tooling costs are required and the minimum quantity resulting from today's processes can be overcome towards a profitable small series or custom one-off approach. No more warehouses full of alike products and no more landfill from unsold obsolete products. In addition, some 3D printing processes are fully recyclable contributing to a "greener" future. The 3DprintedClock also contributes to the aspect of ready assembled or "fab-sembled" products where former labor intensive assembly are folded into the design and fabrication process. This can lead to revitalizing local small manufacturing of mechanical products as opposed to large-scale production overseas. As discussed in the 3D printing section, the "built to last" paradigm for which mechanical clocks have been an important driver can be revised through examples like the 3Dprinted clock and evolve towards new relations between time, material, form and function.



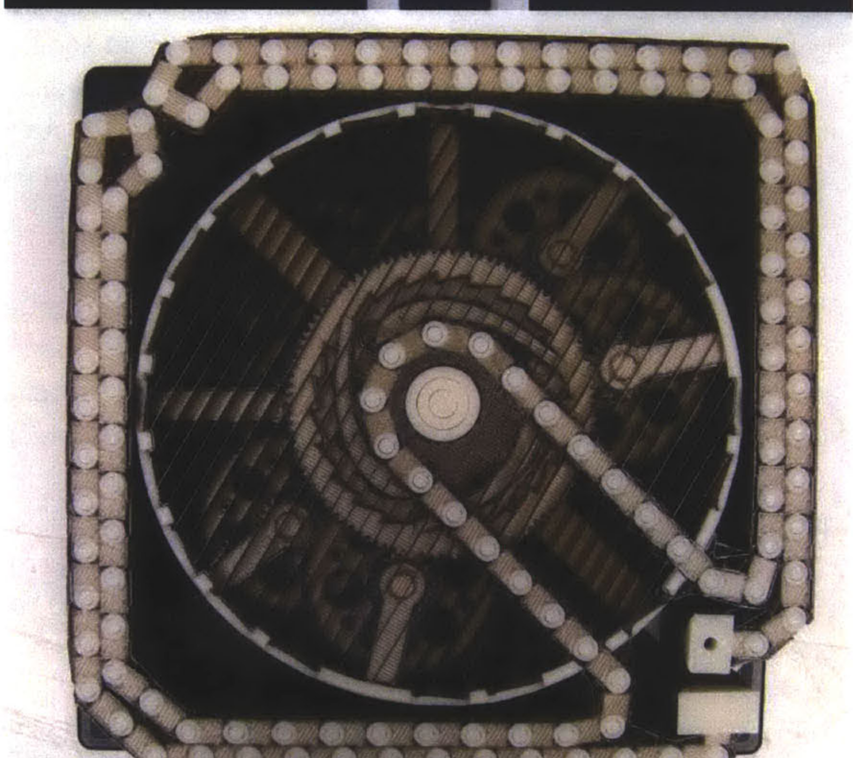


Figure 3.57 (Above and Left)) 3DprintedClock and detail. Image of an unexpectedly aborted 3D print of the clock revealing its inner structure and the support material added by the 3D printer.

Electronics

All the assemblies and mechanical components described above require customized and customizable electronic components that can be fluidly integrated with the specific objects. I have been focusing on integrating and miniaturizing as a strategy for electronics after trying plywood PCBs which proved themselves highly labor intensive and impossible to debug. With the exception of mechanical devices such as the clock, kinetic artifacts in general require motor drive, control and communication electronics to create motion. My approach is to combine a micrcontoller, H-bridge and sensor circuit to create a basic board for operating servo motors. The result aims to be a mid-range servo implementation balancing all tasks involved and leaving some extra computational resources on each motor control board. The program stored on the board allows a wide variety of operational modes for the servo. It can be operated by communicating numerical values which can either be interpreted as angle values or speed values. Incremental values can also be used to rotate the servo in either direction for a certain number of degrees quite like a stepper motor. In addition, values from the servo can be read out on a computer through the micro controller. The program can also be used to run user code in addition to the servo control code to enable embedded and stand alone applications. This strategy will hopefully support a basic stand alone machine or robotics setup not requiring bigger computers to be involved but allowing for it when necessary.

Throughout the iterative development process the laser cutter was used for wooden PCBs and the modella mini mill for early prototypes. After these initial attempts, the success of Arduino and Open Servo communities as widely used and stable development environments with algorithms for motor control and servomechanisms prevailed. The circuit design is guided by Arduino compatible components with a generic H-Bridge circuit. The H-Bridge is made form two channels each having one N-Channel MOSFET and one P-channel MOSFET for which and additional N-channel MOSFET is used as driver. Common packages like SOP8 or TO252 are used for the power MOSFETs to be replaced with project specific ones. Optical interrupters are intended as the motion and position sensors supporting continuous motion and sensing rather than a potentiometer limiting all applications to mostly less than 300 degrees of motion. The entire circuit is implemented on a small footprint while keeping motor power and ground separate from the logic circuit section to reduce motor noise interference. A voltage regulator guarantees the whole circuit to be operational at a wide input voltage range. I/O pins not used for the H-Bridge or optical interrupter are made available through solder pads including TWI, serial and USART communication pins. The latest iteration of the board has the optical interrupter located on the bottom side of the board in order to align with the encoder disc mounted to the rear end of the motor axle.

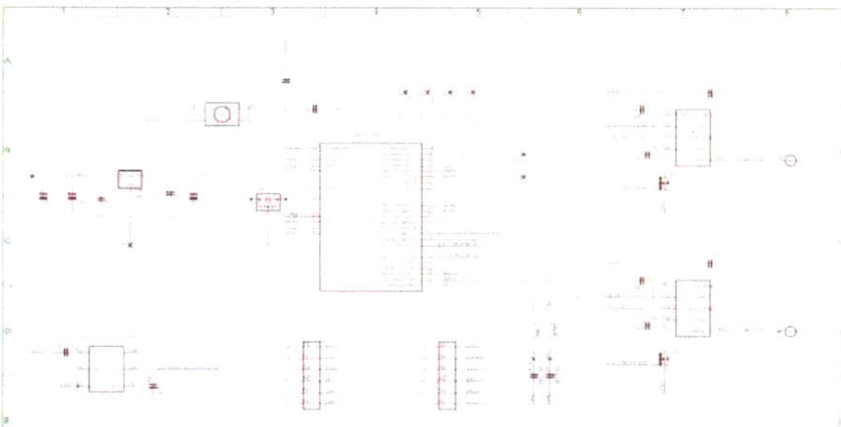
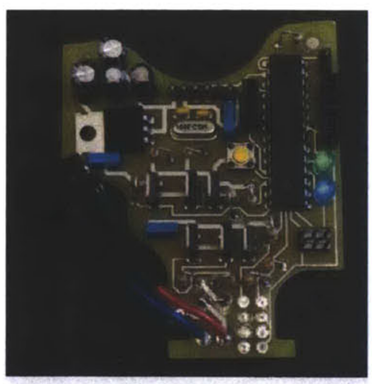
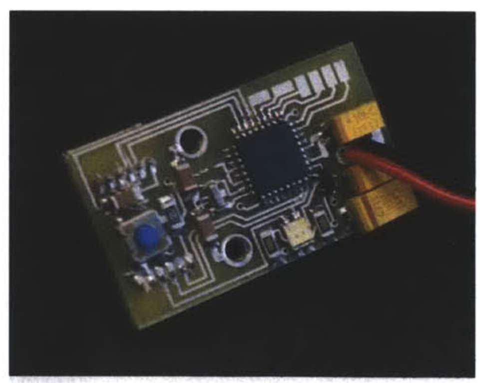
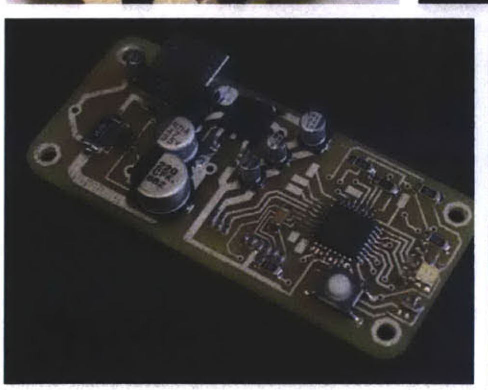
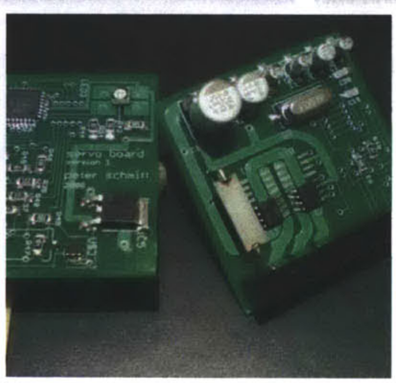


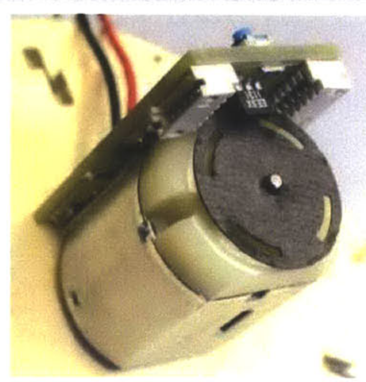
Figure 3.58 (Left and Below) Circuit diagram of the latest iteration of the pick back board combining an Arduino compatible circuit with a











PlywoodServo

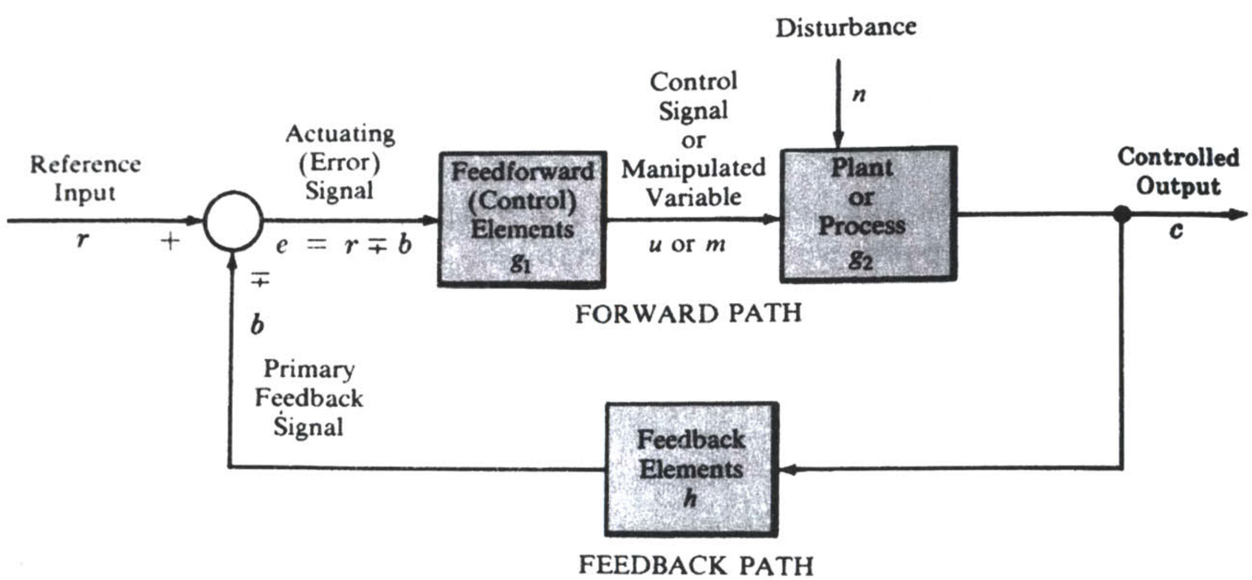
The Servomechanism

The term "servo" refers to a mechanism or actuator including an error-sensing feedback loop which continuously and directly contributes to an automatic self-correction. A servo can be described as continuously self-sensing and control entity which transfers an input signal into a – mostly physical- output such as rotational motion. Common fields of application include position control, force control, temperature control and many more.

The basic functionality of a servo motor and servomechanism is formulated in the field of Control Theory. Fig. 3.59 shows the basic control theory underlying a servomechanism, its elements and their interconnections.

The desired commanded input value, (Reference Input) at time of command often differs from the actual accomplished value (Controlled Output). This difference is measured by comparing the feedback value with the commanded input value. The difference determines the error value which is used to actuate/ drive the system. This actuation will bring the system closer to the actual accomplished value which will feed back into the error and regulate the actuation until the commanded input value equals the output value. The actuation is not only determined

Figure 3.59 (Below) Control Theory functional diagram of a Servomechanism, DiStefano, J. J., Stubberud, A. R., & Williams, I. J. (1995). Schaum's outline of theory and problems of feedback and control systems. New York: McGraw-Hill. Page 16



by the error but also by outside factors acting on the system (referred to as "Plant" in the diagram). These disturbances also need to be incorporated into the actuation in order to accomplish the commanded value.

One of the most commonly used software-based algorithmic implementations for this closed loop control system using a motor, gearbox and position sensor is realized with a PID (proportional, integral and derivative) algorithm.

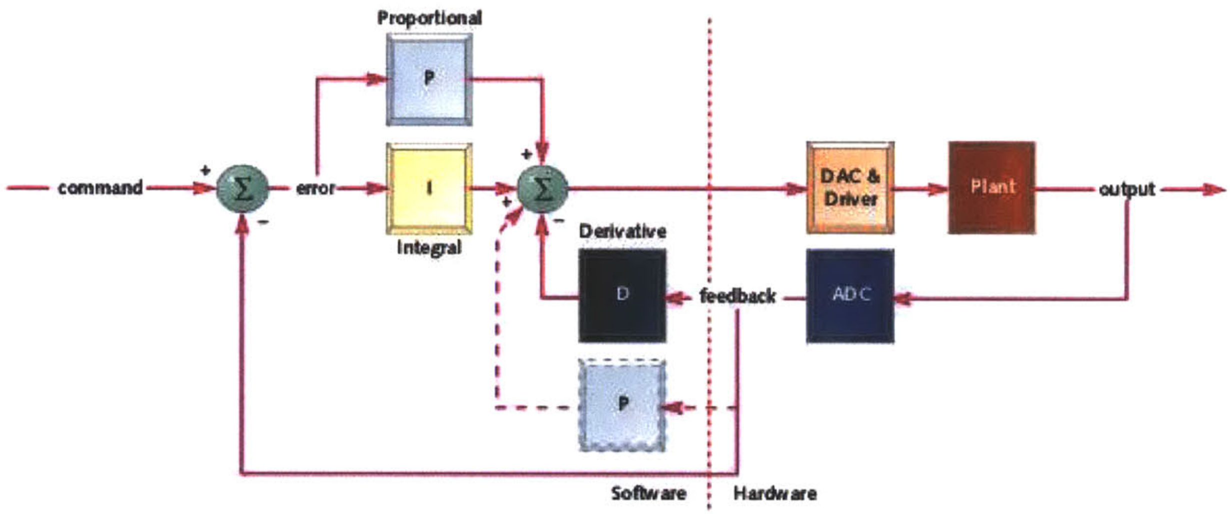


Figure 3.60 (Above) functional Diagram of the PID algorithm, Wescott, T. PID without a PhD. In Embedded Systems Programming. http://www.embedded.com/2000/0010/0010feat3.htm

This algorithm defines three components, the proportional, the integral and the derivative gain and combines them in order to create an accurate output. Every motor and mechanical system provides friction and inertia which will cause a delay between actuation and actual accomplishment. Like the delay between pressing down the accelerator pedal in a car at a certain speed until one reaches the desired speed. The proportional component of the algorithm does exactly what a car driver would do, it presses down the accelerator pedal a bit harder in order to get up to speed faster. This is accomplished by multiplying the error with a fixed value, the proportional gain. This gain results from the "delay" of the system. In the example of the car, a light-weight small size vehicle would get a smaller proportional gain value while a heavy duty truck would get a bigger proportional gain value. However, multiplying the error with a proportional gain can lead to several problems like accelerating too fast and overshooting the desired position or under-accomplishing the controlled task. While the algorithm senses the system several thousand times per second and calculates the according actuation output several hundred times per second, the integral component is introduced to track a certain number of past calculations in order to eventually correct the proportional gain to ensure accurate accomplishment. Similarly, the derivative component of the algorithm compares last position with current position which can be used to predict the future position (as opposed to tracking past calculations as in the case of the integral component). This will also allow adjusting the correct actuation. The combination of all three components ensures a fast, accurate and efficient way of driving and controlling a servomechanism.

The RC Servo

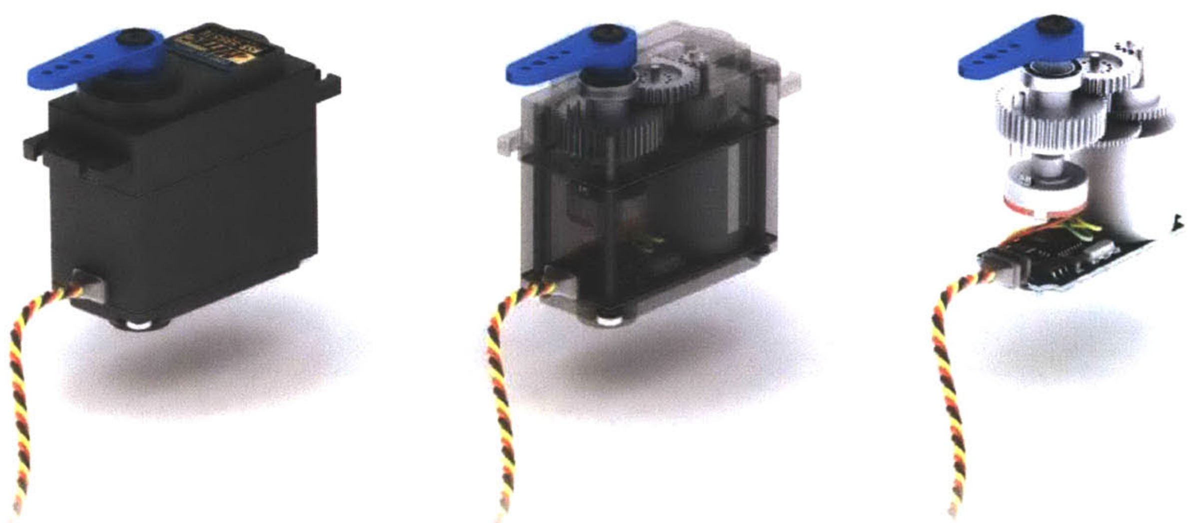


Figure 3.61 (Above) RC Servo, HSR-5995TG (CAD model and CG rendering by Peter Schmitt)

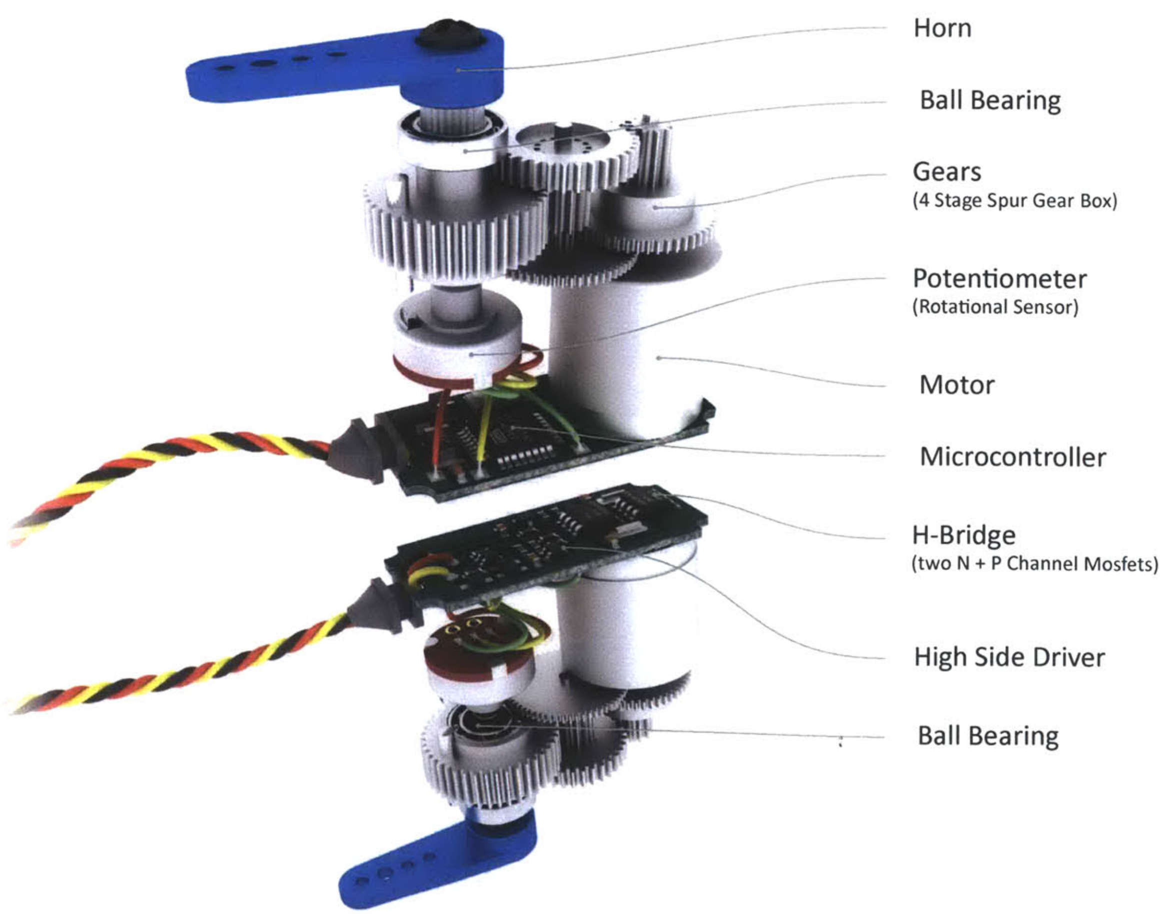
The RC Servo dates back to the beginnings of radio control. Its popularity surged with the emergence of RC model-making scenes following WWII. The basic functionality and principles of the RC servo were established at this time and remain the dominant mode.

A standard radio-controlled (RC) servo unit (see Fig. 3.72) contains a DC motor, gearing, and control circuitry. Based on coded input signals the motor positions an output shaft to a particular angular position under varying conditions. Exterior forces such as impacts or the system's own weight also impact the servo's position. In order to maintain a desired position, the coded signal must be continuously adjusted. Applications for RC servo units include radio-controlled airplanes, cars and puppets as well as robotics and other actuated devices.

Typically, a PWM (Pulse Width Modulation) signal is used to set the angular position of the servo's output shaft. A 1ms pulse causes the shaft to turn to its zero position while a 2ms pulse results in the maximum position. Motion ranges from 0deg to 90deg or 180deg. The PWM signal is refreshed 50 times per second. When the pulse length changes very little the servo turns slowly. When the pulse length changes very quickly the servo turns at maximum velocity (approximately 200ms per 60deg).

Internally, a feedback loop assures position accuracy. A potentiometer connected to the output shaft is continuously monitored. Its resistance provides feedback on the actual position of the shaft. A PID control algorithm compares actual position to signaled position. It adjusts speed and force of the

Figure 3.62 (Below) RC Servo exploded (CAD model and CG rendering by Peter Schmitt)



motor to reach a desired position. With a constant input signal, the PID control algorithm will also drive the motor to maintain the signaled position.

To summarize, the RC servo typically consists of a box-shaped body closed with four screws. The top provides an output shaft equipped with a horn like lever arm which can also be connected with three to four screws. The range of motion provided by the horn in relation to the body is typically between 90-180 degrees. A continuous rotation modification can be made which unfortunately makes the servo unusable for position control because the input signal will be interpreted as a speed/velocity/RPM signal. A three conductor wire of a predetermined length and with a specific connector serves as connection to power and input signals. The signal wire only provides one-way communication and does not support reading out servo data. The signal to drive the servo typically needs to be created by an outside instance like a microcontroller.

Figure 3.63 (Above) Dynamixel RX-64 Small Scale Robotic Actuator

An Example: Dynamixel

In response to a growing interest in small-scale robotics, most RC servo manufacturers offer specific "robotic servos" intended for robotic applications rather than the RC hobby field. Dynamixel is a manufacturer whose only focus is to make small-scale robotic actuators very similar to RC servos. The hardware and electronics consist of a DC motor, a gear train, an output shaft and a circuit board with motor drive electronics as well as control and communication capabilities. The casing varies by RC servo to support more attachment options within roughly the same scale. Instead of a power and signal wire, the Dynamixel actuators provide a plug allowing for user-defined wiring. In terms of functionality, they achieve a bigger range of motion (up to 360 degrees), a significantly higher torque and provide a RS 485 or TTL or CAN bus interface allowing to power and control up to 254 devices on one network. This high-level communication also enables reading out data such as position, force and power consumption as well as commanding the position by transferring values like numerical or angular positions. In order to drive the Dynamixel servos, a microcontroller or computer capable of communicating via RS 484, TTL or CAN is needed.

The Servo Motor

The term "Servo Motor" is often used to describe electrical motors that are part of a servomechanism not being self-contained like a RC or Dynamixel Servo. The term is misleading as it implies the servo only consists of the motor while in fact the servo is a system requiring a motor, some kind of gearing, at least one sensor, a motor drive electronics and a control instance assuring closed loop operation. Due to the fact that most mid-size and large-scale (industrial) servomechanisms are typically modularized into motor, gearbox and drive/control- electronics, the term "Servo Motor" stuck as an acronym. In some cases when a rotary sensor is attached to a motor this unit is likely to be intended for use in a servomechanism and as a result it is referred to as a servo motor.

The diagram shows a typical servomechanism loop with all the necessary components. The motor is sandwiched between the rotary sensor in the back and the planetary gearbox in front. The planetary gearbox provides the output shaft as well as threaded inserts for mounting. The motor is connected to the H-Bridge which is an array of four MosFets acting as switches. The H-Bridge owes its name to the shape of the MosFet array which -when drawn as schematic- resembles the capital letter "H". The H-Bridge as well as the rotary sensor of the motor is connected

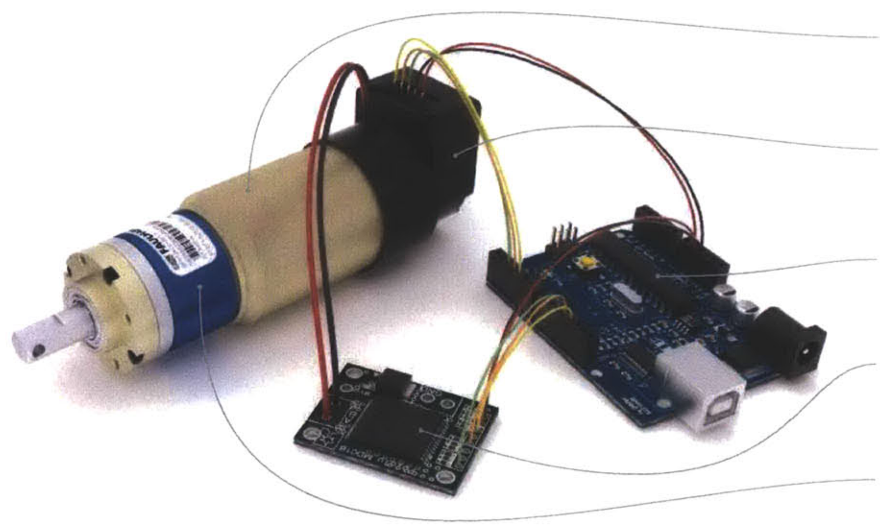


Figure 3.64 (Above) Faulhaber DC Motor with Planetary Gearbox, HEDS Rotary Sensor and Arduino control PCB with additional H-Bridge.

to the Arduino PCB which contains a microcontroller capable of running a PID algorithm. The Arduino PCB also provides a USB bus to interface with a PC and has additional computational resources allowing the user to place the program running the servo on the PCB itself. This is a great advantage over RC and Dynamixel servos as they contain microcontrollers which are not accessible for user code and programs. In addition Arduino is an open source community based project which offers lots of support and access to sample projects. This enormously supports conceptualizing, designing and iterating actuated objects by the designer and user.

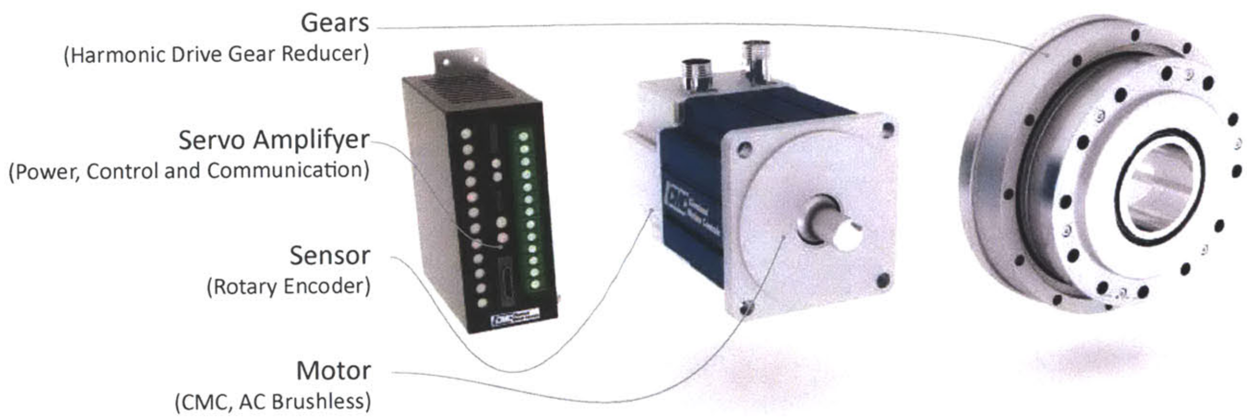
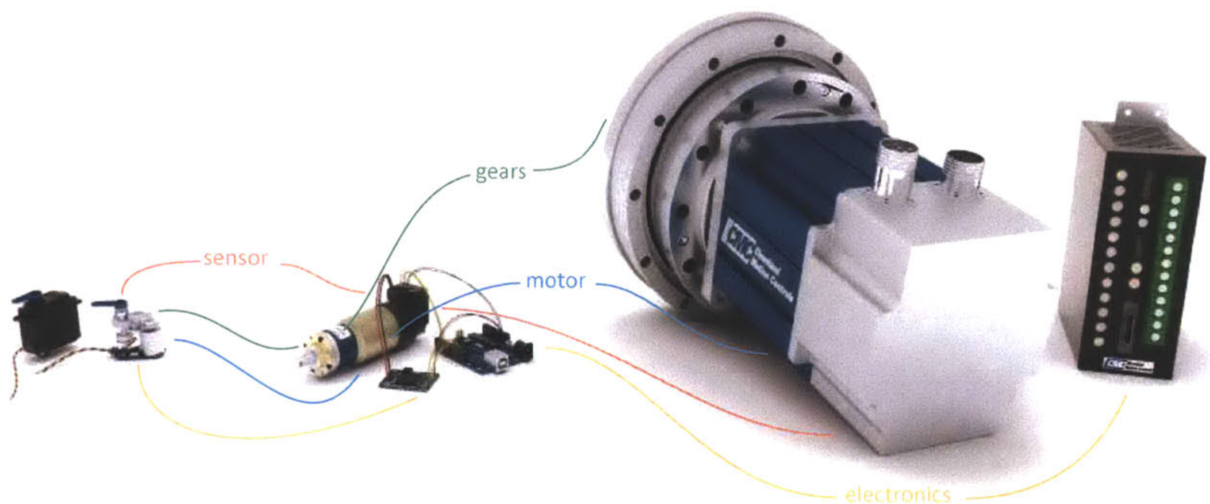


Figure 3.65 (Above) Cleveland Motion Controls MDM-5000 Brushless Servomotor, Harmonic Drive Gearbox and ... Drive and Control Electronics (CAD model 3dcontentcentral.com, CG rendering by Peter Schmitt)

Similar to the mid-size servomechanism Fig. 3.65 shows all necessary components on a large-scale. The motor and rotary sensor is one unit to which the control electronics box connects. The gear head is a harmonic drive which has a big reduction ratio as well as huge torque while still operating efficiently. Harmonic drives are mostly used to directly drive a rotary joint like industrial robot arms. They also function as load bearing for the same joint which is not the case with most planetary gear heads for example. In terms of motor control, algorithms for position- speed- and force- control and communication the control electronics represent a black box. As these large-scale components are high end and high cost they are also purposely highly specialized by the manufacture mostly for business reasons. Interfacing to a computer while not using the manufactures software is almost impossible which all together makes these components hard to use for an iterative development process.

Problem Space Diagram

The diagram in Fig. 3.66 shows a comparison of the RC servo, the mid-size servomechanism and the large-scale (industrial) servomechanism and their typical appearances. The diagram traces the recurring principles in each application revealing the common DNA of each servomechanism containing sensor, motor, gears and electronics. Despite this common anatomy the implementations are vastly different and scale-bound. The question arises of how to overcome the specific boundaries towards a more universal approach enabling the transfer of a



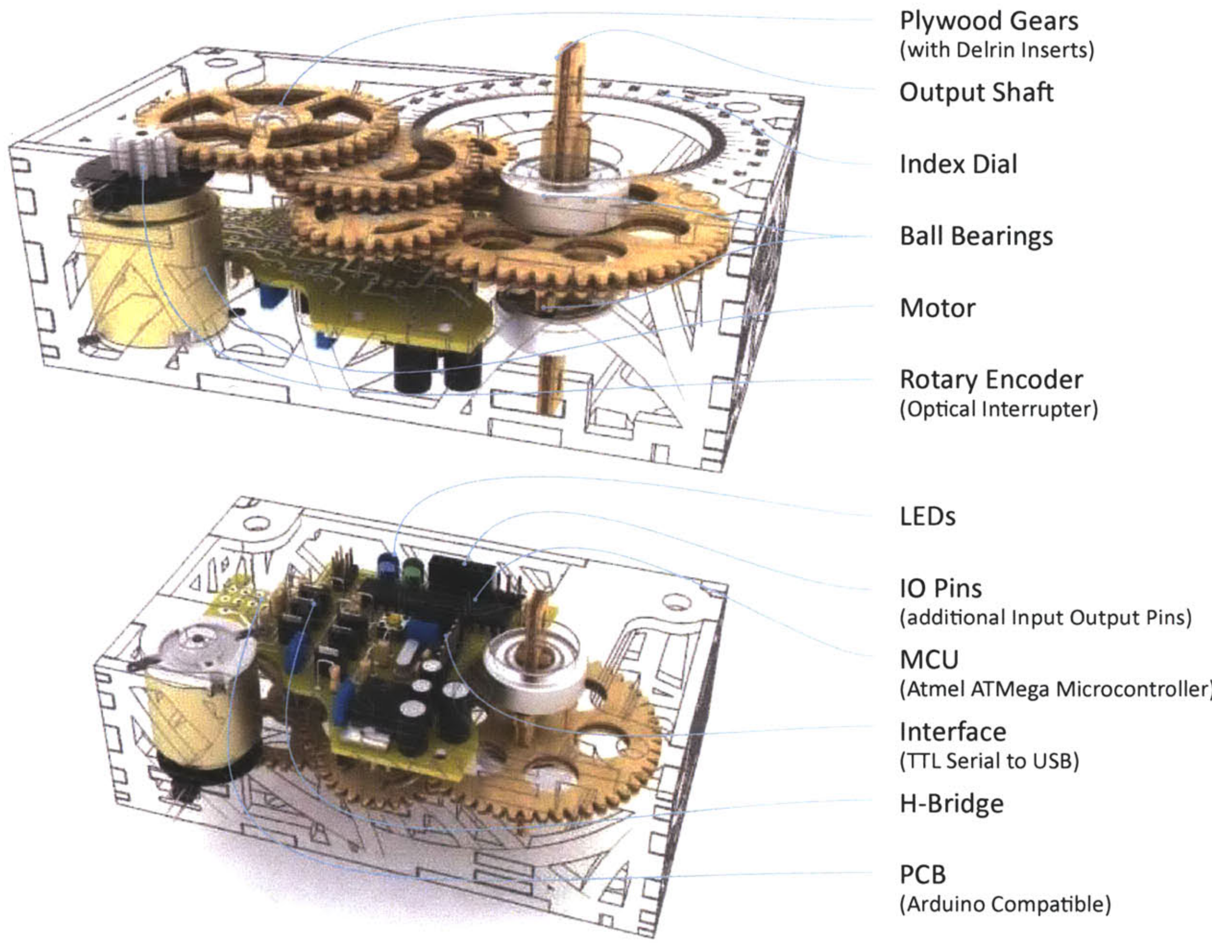
servomechanism from one application to another without the constraints of scale and operation environment. Component boundaries and their resulting interfaces seem to play a big role. Electric motors exist in various sizes and require an H-Bridge circuit to be driven. The electronic power components (MOSFETs) needed for an H-Bridge circuit are standardized in their packages and pin-outs thus making them interchangeable for different power requirements. In regards to sensors optical interrupters in a one or two channel configuration can be used to sense almost any kind of motion feedback possible. Taking all these subsets into consideration the ideal of a universal, scale-independent and adaptable (for different components) servo mechanism implementation should be within feasibility of today's technology.

Figure 3.66 Above) Comparison Diagram of Different "Servos" and their Applications; Bioloid, Domo, Industrial Robot Arm (CAD model and CG rendering by Peter Schmitt)

Developing PlywoodServo

Figure 3.67 (Above) Photo of the Plywood Servo (Photo and Plywood Servo by Peter Schmitt)

In robotic systems, the most basic component parts such as servos present a designer with a set of constraints such as form that she cannot control. The underlying logic for these factors derives from standards and established practices that have evolved over time to fulfill the most typical design requirements



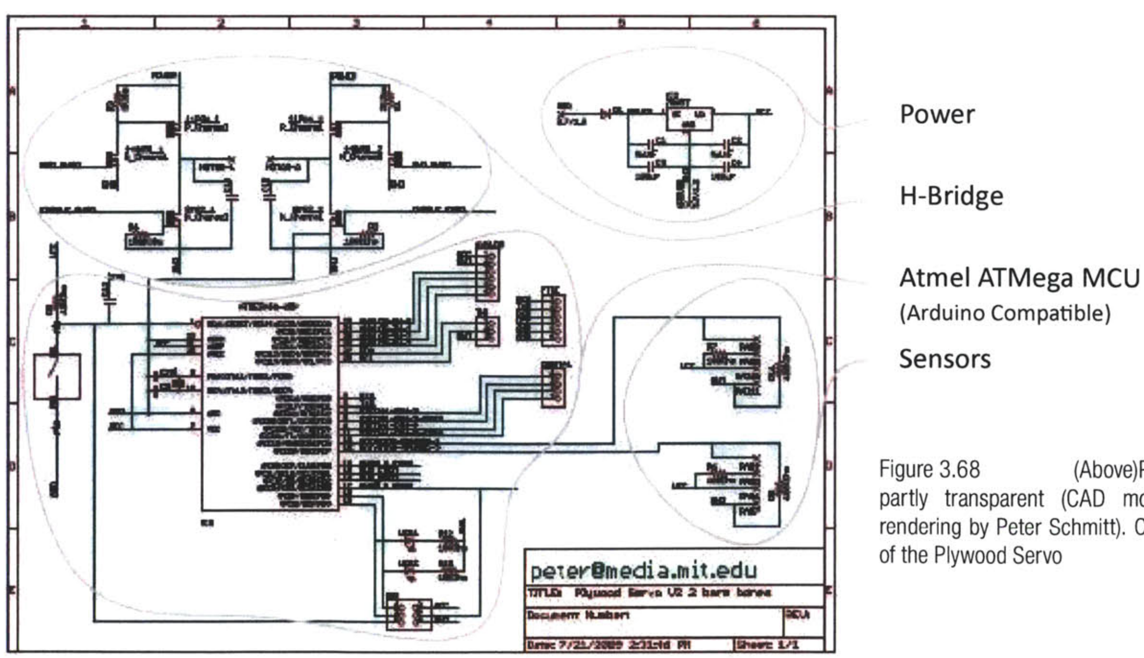


Figure 3.68 (Above)Plywood Servo partly transparent (CAD model and CG rendering by Peter Schmitt). Circuit diagram of the Plywood Servo

in various problem spaces. The resulting components could be considered black boxes that become given factors around which design is created rather than an integral part of the completed artifact.

The PlywoodServo revisits the basic servo and opens its inner workings. The study targets designers, tinkerers and students who may not be satisfied with off-the-shelf products and who may envision more diverse animated artifacts than are currently available. In the long run this exploration is intended to demonstrates how more holistic approaches to motion-enabled objects may lead to new types of artifacts that celebrate mechanical and electrical components and recapture the magic of engaging with them.

Description of the Plywood Servo

For this analysis, the components of a servo described above were distilled into the following elemental steps:

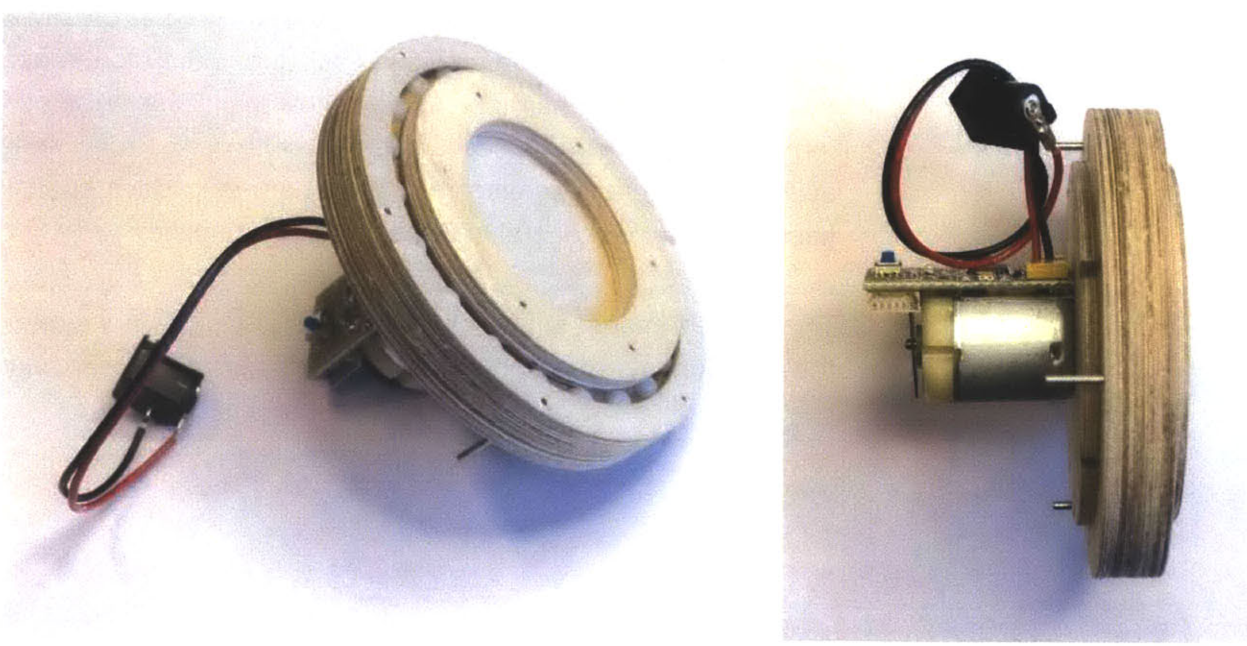
- Connect motor and gears to an output shaft
- Sense motor rotations as a simple feedback loop
- Attach a clock hand to the output shaft for visual feedback on servo position and performance
- Provide standard communication interfaces like serial, USB, or TWI to drive the unit
- Provide open source, community supported coding environment like Arduino and Processing

The early laser cut plywood gear explorations shown in Fig. 3.78 informed the iterative development of the PlywoodServo. And most of the structure and gears remain constructed from plywood. In sum, there are three main iterations starting with a simple three-stage plywood spur gear box, over a layered version using refined gears and axles featuring less friction and the third iteration in which the plywood case also houses a plywood PCB. The final PlywoodServo contains a four-stage spur gear box made from press fit small delrin and larger plywood gears mounted on aluminum tube axles which are fixed by a screw through their center. Conventional ball bearings are used for the output shaft. Some of the smaller gears are in Delrin, a plastic designed for mechanical applications. The plywood and Delrin components were designed in CAD software (Rhinoceros) and cut out on a laser cutter. Screws and standoffs serve as axles for the gears.

A small electric motor, electronic components (microcontroller, MOSFETs, resistors, capacitors and optical interrupter) were embedded in a circuit board first made from plywood before outsourced as PCB for later iterations. Fig. 3.68 also shows the circuit diagram used for the plywood servo. The plywood cage keeps all components together and in position. The main axle equipped with an indicator hand and a dial is held on two (skateboard) ball bearings. All components are held in place by the plywood case. The output shaft is surrounded by a scale which is laser etched into the plywood and indicates 360 degrees. The overall gear ratio is 360:1 which relates every motor rotation to one degree of rotation on the output shaft.

In a second approach a plywood cycloidal gearbox was used to implement a servomechanism. The miniaturized electronics with a sensor on the bottom of the PCB serves as a piggy back solution sitting alongside the motor and sensing the encoder disc glued to the end of the motor shaft. The plywood used is 1/8", 5 ply,

Figure 3.69 (Below) Second version of the Plywood Servo using a 2 stage cycloidal gear reducer and the piggy back circuit board.



aircraft plywood machined on the Shopbot CNC router using a 1/16" end mill. Metal pins align the layers of plywood. The bearing was implemented using the V-groove end mill. The precision of both the cycloidal cam disc and the bearing raceways need to be improved which is mainly related to the Shopbot's capability of repetitive precise operation.

The overall performance of this early design is already very promising. The simple functional design only requires three movable parts with easy-to-machine and simple-to-assemble ball bearings in a material as widely available as plywood leads to a successful combination to be shared with other users across different backgrounds. The cycloidal plywood servo holds the promise for a successful solution in regards to opening mechanical black boxes for a higher level of integration.

3D printed Harmonic Servo

Similar to the cycloidal plywood servo, the 3D printed harmonic servo uses the piggy back electronics driving and sensing the motor with a pancake style gearbox. The harmonic drive presents the difficulty of machining many small gear teeth in a material with enough flexibility to serve as a flex spline. 3D printed plastic comes very close to these requirements and the Invision PolyJet printer provides the necessary resolution for the small gear teeth. As an added benefit, the Invision PolyJet uses wax as support material which provides basic lubrication to the gear box. Designing the harmonic drive is the main challenge as it requires multiple special cases of the involute gear. Professional applications actually create their own, harmonic drive specific gear tooth shapes for static, flex and dynamic spline.

The adjacent images show a cut open CG rendering demonstrating the components and configuration of the harmonic servo and two images of teh finished 3D printed version. The harmonic drive shown has a 560:1 gear ratio in a 4" diameter housing also including the four-point contact bearing using 5/32" delrin balls assembled through a ball feeding channel sealed with a cork after assenbly. A bigger motor was attached and the drive circuit upgraded with more powerful MOSFETs.

Operating the harmonic servo revealed unexpectedly fast fatigue problems with the flex spline. Flex splines with a diameter smaller than 3" tended to break within a couple of minutes of operation. Unfortunately the minimum tooth size could not be reduced in

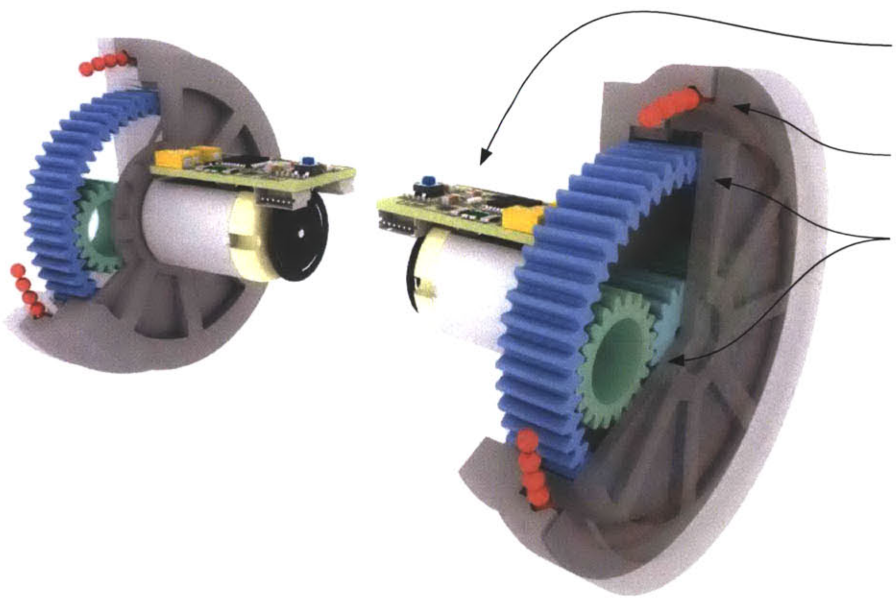
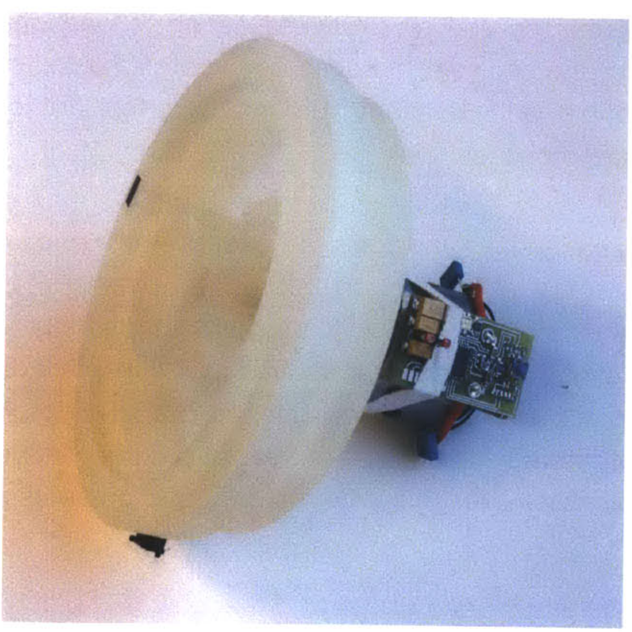


Figure 3.70 CG rendering of the paraServo created by the Grasshopper definition in combination with a motor and the motor/servo electronics and the sensor disc.



order to improve the diameter to flex ratio which led to exploring different 3D printing methods. The FDM machine was able to create flex splines with sufficient flex and fatigue resistance with the downside of lower resolution causing the teeth to be less accurate. Laser sintering though the most likely candidate for durable and flexibly nylon parts was extremely difficult in terms of adjusting CAD models to the machine-related offsets. Solido, a tape laminating 3D printing method printed the most reliable flex spline parts with smaller than 3” diameter. Still, improving the accessibility of laser sintering for this type of gear box and servo remains a key enabler for widespread use and should be revisited. Overall the 3D printed harmonic drive servos performed better than expected in terms of torque, speed, ease of assembly and precision.

Conclusion

Current robotic and kinetic systems, animated objects and machines are frequently defined by existing systems of modules that predetermine many crucial design elements of a final object such as material, scale, formal and kinetic appearance. In order to break open the design of actuated objects and animated artifacts, the preceding analysis studied digital design and fabrication tools and methods, conventional mechanism and their translation into digital fabrication environments and the servo mechanism. All these aspects are central elements in most system architectures. The analysis exposes existing conventions for modularization and begins to propose a new organization of building blocks that enable iterations across scales that do not require entirely new architectures.

The costs of gaining the additional freedoms from digital fabrication also come at a cost compared to off-the-shelf components. Manual labor is required to assemble parts that are normally readily available. Assembly requires skill and practice that can only be acquired through experience. Depending on the complexity of the project, significant specialized knowledge about mechanical assemblies might be required and presents a barrier to entry.

Though well-documented above, the cook book is only the first step towards translating conventional and new mechanical assemblies into something that can be digitally fabricated. Many of the details listed above can be framed as new constraints linked with a particular fabrication method or adapted hybrid design and assembly process. These constraints could be codified into parameters that are built into existing software pacakges so that users without special skills in the area can use them as they create new artifacts.

Chapter 4

Towards a Parametric Design Tool for Object-Oriented Mechatronics

CAD software as a computational tool has the capabilities to integrate all aspects of form, shape, mechanics, kinetics, assembly, manufacturing, control and animation. However, individual users may not be specialists in all these domains. A mechanical engineer using CAD software in order to design a movable joint knows the tolerances and offsets required for a particular manufacturing process. However, he may know little about the concept and design qualities of the desired end-product. Conversely, a designer may be using the same CAD software to shape a concept with specific design qualities. However, he may not be capable of implementing it in a mechanically functional way. "Object-oriented mechatronics" is a parametric design approach that connects knowledge about mechanical assemblies and electronics with the requirements of digital manufacturing processes. Parametric instances like gears, bearing and servos are made available as objects within a CAD environment which can then be implemented into specific projects. The approach addresses the missing link between accessible rapid-manufacturing services and currently available design tools thereby creating new opportunities for self-expression through mechatronic objects and machines.

Object-Oriented mechatronics depends on a set of tools enabling a design and fabrication workflow that integrates actuation, animation and fabrication with final object qualities. Rigid mechanical and robotic assemblies can be treated as malleable, parametric and digitally manufacturable entities. Previously separate components like ball bearings, gear box components, motors and electronics can seamlessly be integrated along with the overall structure and shape at any scale. Thus overcoming the need to have pre-made, discrete and off the shelf components drive the design process and final outcomes. The results qualify for additive manufacturing processes like 3D-printing using plastic or metal materials. The assembly requires a minimal number of steps involving only a small number of additional components like ball bearing balls, electronic and motors.

The parametric design tool includes a knowledge base for all aspects of an original machine project including concept, design, fabrication, control and animation. Like in object-oriented programming, the system will allow users to call an object with specific parameters. A new instance of the object then builds itself within the system with specific parameters that can be adapted. Users will then be able to choose at which level they would like to engage with different aspects of their design.

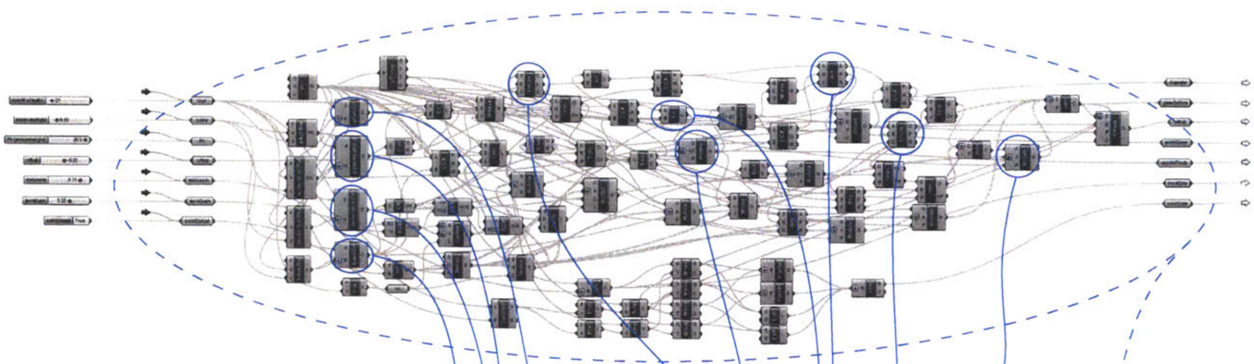
Chapter 4 describes the tools involved in the object-oriented mechatronic approach, the workflow enabled through the tools using a simple example and the assembly process for originalMachines. To get started, originalMachines are created from a set of components defined in Grasshopper (http://www.grasshopper3d.com/) for Rhinoceross 3D-CAD (http://www.rhino3d.com/). These software packages should be installed before beginning an originalMachine project. Rhino is a commercial 3D computer-aided design (CAD)software developed by Robert McNeel & Associates with a free 30-save-trial. It is currently in version 4 SR9 supporting 32-bit operating systems with version 5 released as work in progress (WIP) supporting 32- and 64-bit operating systems. Grasshopper is developed by David Rutten, Robert McNeel & Associates. It is a free plug-in for Rhino 4 and 5 and currently in version 0.8007. For further assistance on installing Rhino and Grasshopper and for tutorials please refer to http://www.rhino3d.com/tutorials.htm and http://www.grasshopper3d.com/page/tutorials-1

Tools

Grasshopper is a node-based visual programing plugin for Rhino. Nodes representing Rhino commands and components as well as nodes representing mathematical functions, scripted algorithms and files created by other programs can be dragged onto the canvas and interconnected visually in order to compute data or create geometry within Rhino. The grasshopper definition can be "baked" (a Grasshopper command) in order for the geometry to become editable in Rhino. More complex definitions can be collapsed into cluster components within Grasshopper. The tools used for the object-oriented mechatronic approach are such cluster components. The following will explain the parametric gear (paraGear), parametric servo (paraServo) and parametric bearing (paraBearing) cluster components.

paraGear

Fig. 4.71 shows my grasshopper definition of an involute gear. The input number-sliders and toggle-switches located on the left side allow control over number-of-teeth, module, pressure-angle, offset, thickness, bore diameter and solid output creation. The middle part of the definition calculates and creates the various construction lines needed to draw the gear outline. Starting with the base circle, dedendum circle, pitch circle and addendum circle followed by creating points to interpolate the involute curve for one half of a tooth which is then mirrored around the tooth centerline. Together with a top arch and bottom circle component contingent upon the base circle being bigger or smaller than the dedendum circle all tooth forming components are transformed in a polar array around the center of the gear according to the number of teeth. This gear outline is connected to one of the outputs and also to the offset and solid creation parts of the definition which feed into the other outputs. In addition the pitch circle of the gear and the diameter is outputted. The entire definition is collapsed into a cluster component called paraGear also shown in Figure 4.71. This cluster takes the same input slider and toggle-switches and outputs the same values and geometry while summarizing the entire definition within a cluster.



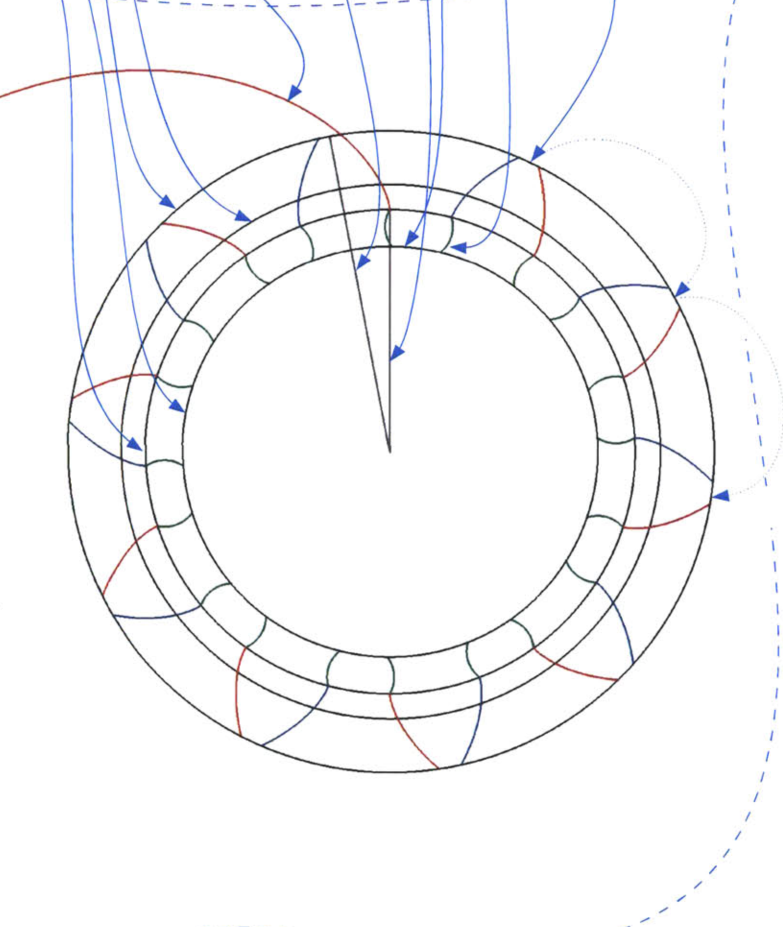
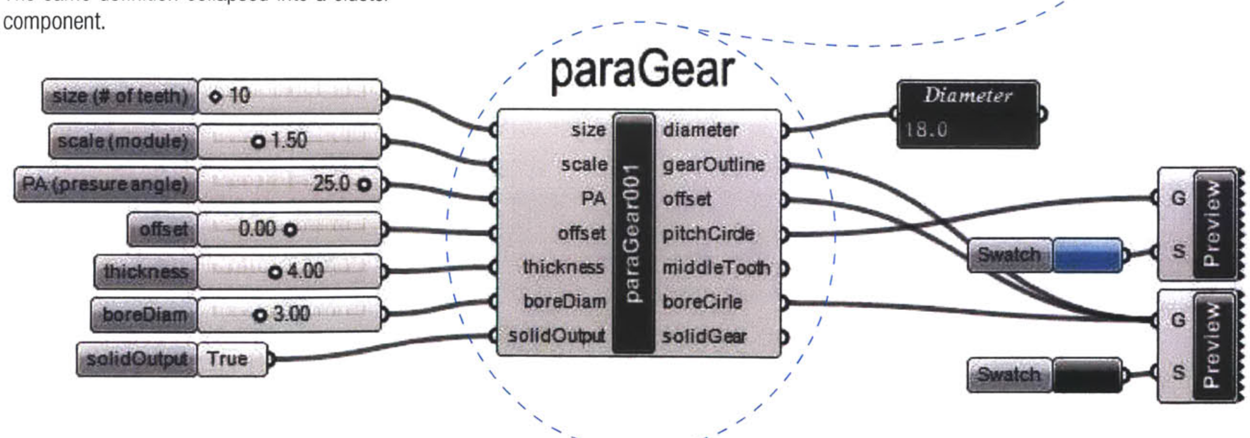
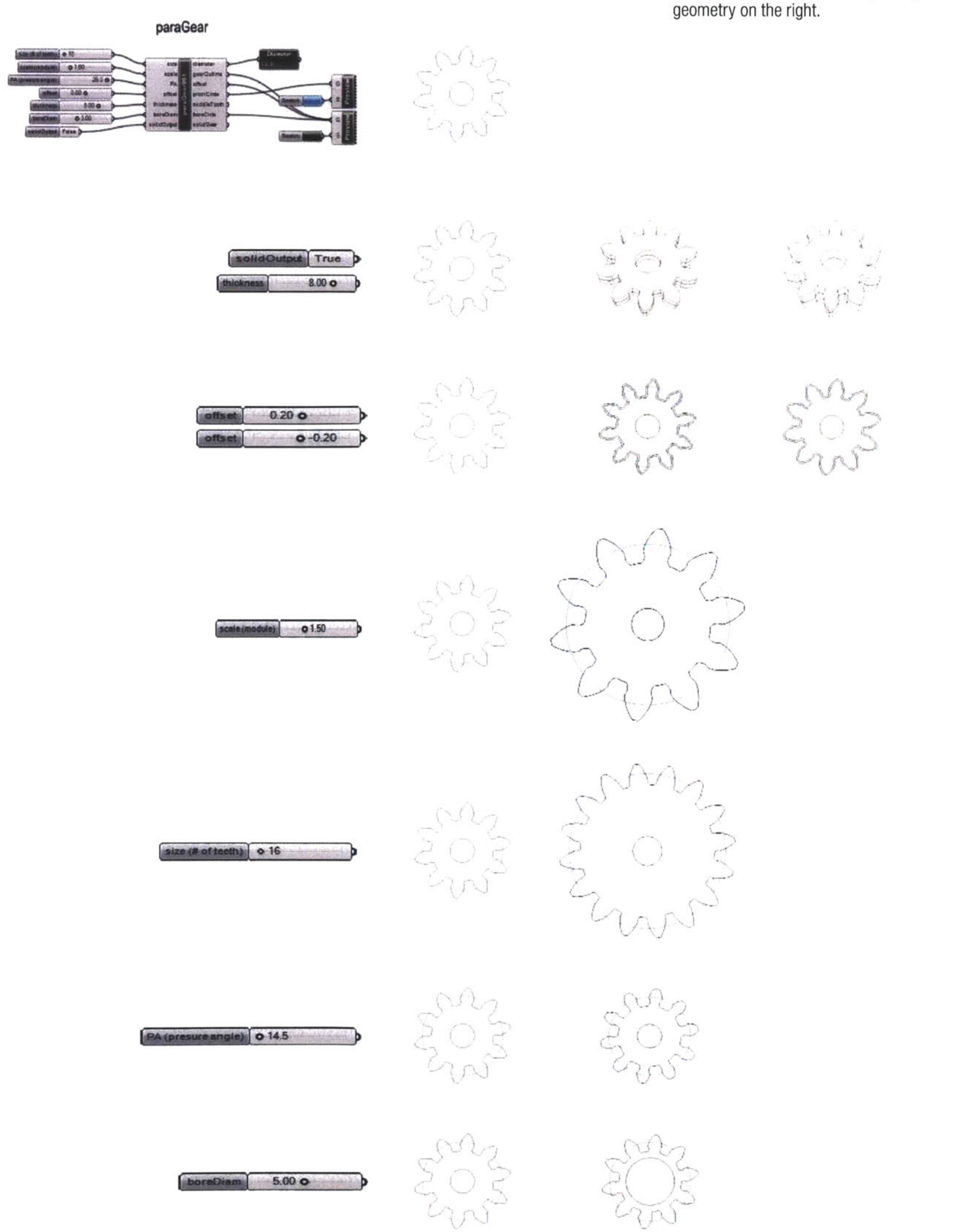


Figure 4.71 (Above) Definition of involute spur gear within Grasshopper. (Right) Curves created by the Grasshopper definition and how they relate with the definition above. The basic curves and circles important for the gear are created on the right side. Followed by the involute of the pitch circle to form one side of the tooth. This one side profile is mirrored around the tooth center line and in combination with the tooth base circle and arc rotated around the gear center as often as the gear has teeth. (Below) The same definition collapsed into a cluster component.



grasshopper cluster components | rhino geometry

Figure 4.72 paraGear definition and it's specific grasshopper input parameter on the left driving corresponding Rhino geometry on the right.



The diagram of Fig. 4.72 gives an overview of how the input parameter based in Grasshopper influences the gear properties in Rhino. The size input represents the number of teeth while the scale changes the gear module (based on a metric gear definition) which influences the overall scale of the gear. The pressure angle describes the angle of the force being transmitted to the gear relative to a perpendicular line through the center of the gear. The value of the pressure angle can vary between 14-25 degree. As the angle increases the gear teeth become pointier. I added an offset function for laser cutting or CNC milling/routing of gears which in some cases requires a cutting-tool related offset. The thickness describes the height with which the gear is extruded and the bore diameter is related to the center hole of the gear. The solid output toggle-switch initiates the 3-dimensional creation of the gear based on the curves and parameters entered. For usability it makes sense to decouple the 3D representation from the 2D one as the gear is more computationally intensive and the input sliders no longer change the 3D output in a real-time and smooth way. This issue becomes even more important with the paraServo definition. The output side provides: diameter, gear outline, offset, pitch circle, middle tooth line, bore circle and the solid gear. Each output requires to be connected to a container, panel or additional grasshopper component in order to be visualized or further computation. The middleTooth output delivers a line from the center of the gear through the middle of one tooth. This line can be used to rotate or orient the gear in a larger assembly.

Gears are used for power transmission purposes in which they mesh with each other. Fig. 4.73 shows the gear-Meshing-Rotation example based on the paraGear definition which allows for gears of any size to be rotated using the rotation slider. There are two important aspects in regards to meshing gears. First, only gears of the same module, or scale as I call it, can mesh with each other. If a change in module (scale) is required for power transmission purposes a second gear with a different module/ scale has to be attached to the same axle. This second gear can than mesh with another gear of the same module/scale. The paraDoubleGear definition below offers a relatively easy solution for this particular problem. Second, every gear-profile

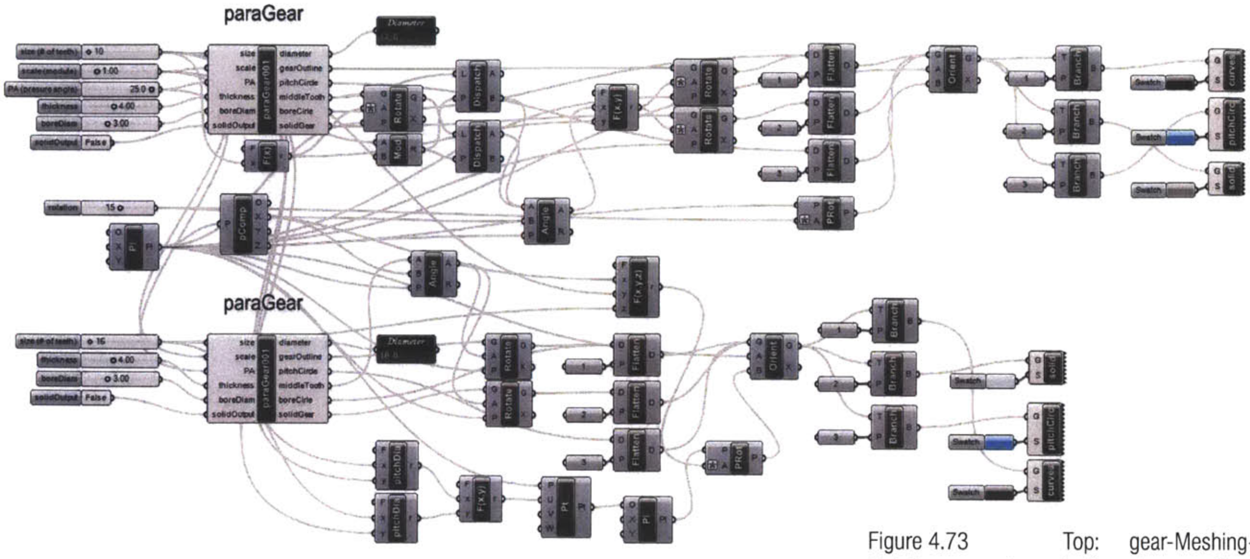
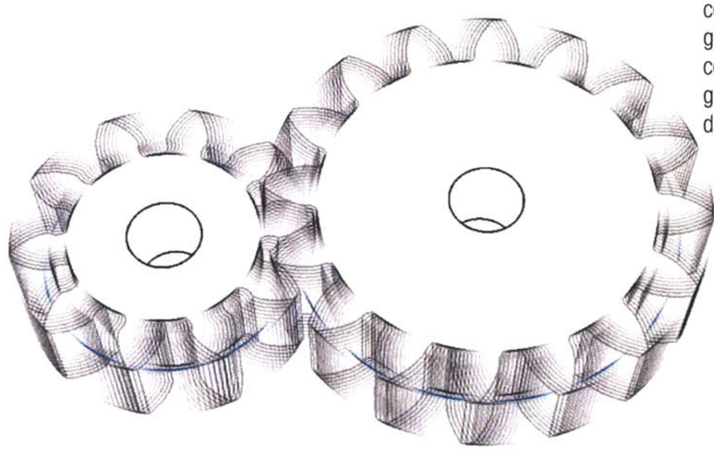
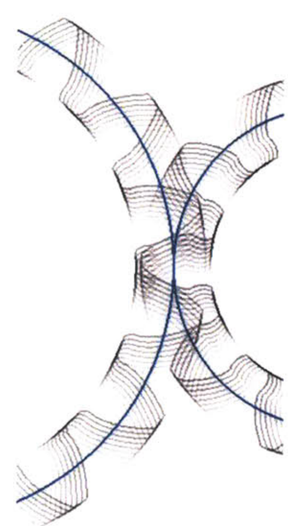


Figure 4.73 Top: gear-Meshing-Rotation example made from two paraGear components which position and rotate two gears according to their size. Left and Below: corresponding Rhino geometry visualizing gear rotation by overlaying several images of different rotation angles.





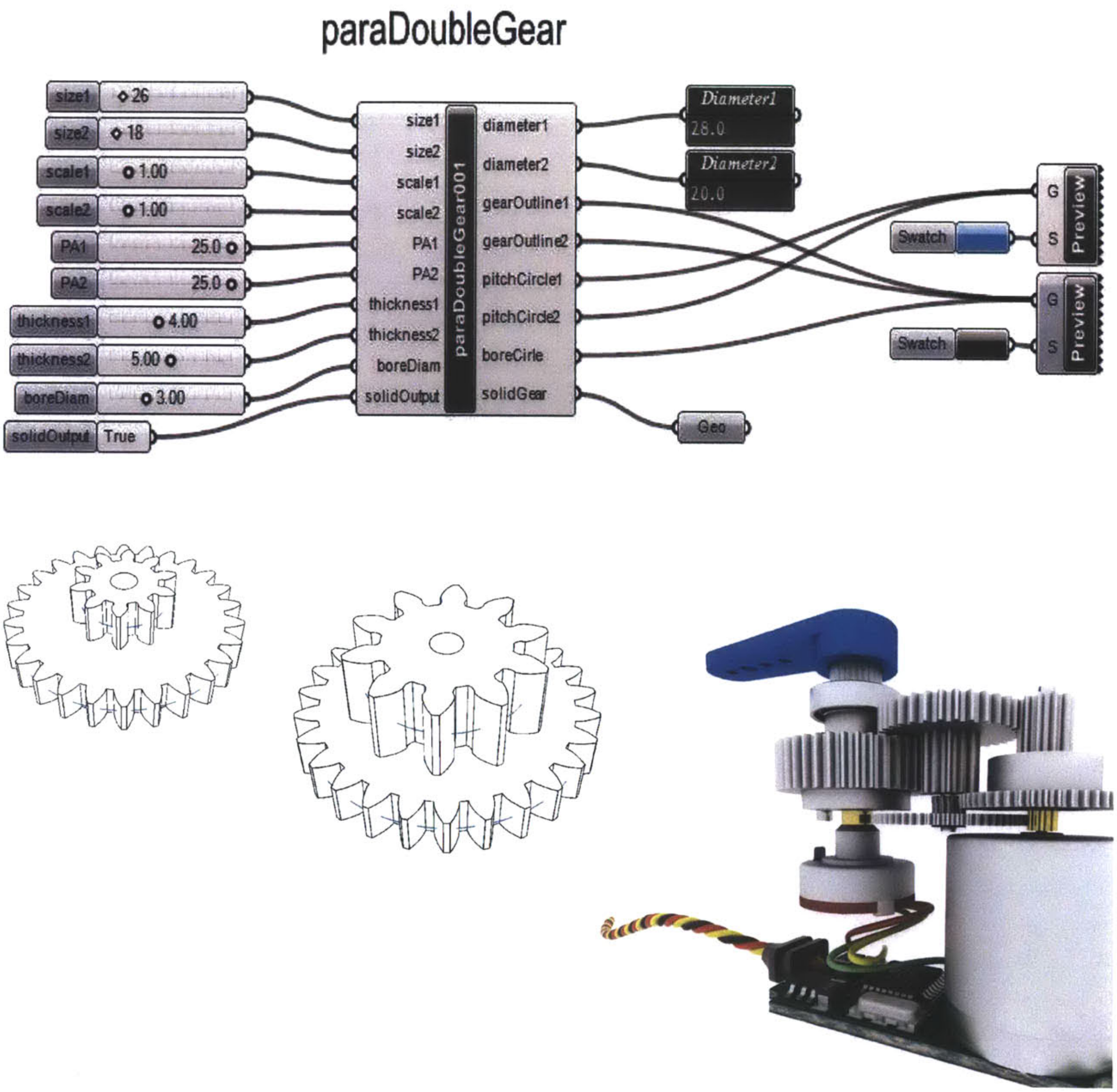
is constraint by a pitch circle. When meshing two gears both of their pitch circles have to intersect in one point. The pitch circles (drawn in blue here) define the distances and center points of tow gears.

The following paragraph describes the gear-Meshing-Rotation example to demonstrate how the cluster components created in this section can be further augmented within Grasshopper. Starting with two paraGear definitions the goal is to create an input slider determining the angle of rotation of one of the gears which will cause the other gear to follow. In order to achieve this behavior the two gears must be from the same scale and pressure angle and oriented towards each other such that they always mesh. In the next step the gear ratio must be determined and used as a factor to multiply the input angle slider which in the third step rotates both gears.

Staring on the left side with two paraGear definitions of which the second one is feed the same scale and pressure angle as the first one. The pitch diameters are computed by size and scale and used to move the gears apart for their pitch circles to meet in one point. A plane is introduced to serve as system of reference for the move, rotation and orientation changes. Both gears are positioned towards each other such that the tooth of one gear is aligned with the space between two teeth of the other gear. This is achieved by measuring the angle between the gear middle tooth lines and the reference plane axis. Additionally the cases of meshing between gears with even numbers of teeth versus gears with odd numbers of teeth or a mix of both cases are separated using the modulo of the division of the number of teeth of both gears. This division also determines the gear ratio which is finally used to re-orient the gears according to the rotation input slider. A significant part of the example definition is made up by flattening the paraGear outputs, feeding the flattened data through the orient component and branching the data afterwards before previewing each data set with different colors. This procedure serves the purpose of applying one orientation change to multiple sets of data (like gear outline, pitch circle, bore circle and solid gear) while maintaining individual identification of each data set. For example the gear outline and bore circle are flattened using identification number 1 while the solid gear is flattened with identification number 2. After feeding both through the same orientation command they can be separated again by branching the orientation-output-data calling the previously assigned identification numbers. Once separated again they can be assigned different colors to be displayed in their rotated position. The example runs fluid in the curve mode and still moderately fluid with solid output set to true.

To complete the definitions regarding gear generation here one more species of gears very useful for gear box applications. Generally, power transmission happens in a gear box in which gears mesh with each other to reduce or multiply their revolutions per minutes (RPM) and torque. A typical gear box application is to take an electric motor as an input and "gear" its RPM down and its torque up as electrical motors usually rotate very quickly (5000-17000 RPM) at low torque (1-15 mNm). Such a gear box would develop through multiple stages from the motor input to the output shaft. Each stage is in contact with the stage below and above which requires two of the same or different gears to be connected in order to form one stage. The paraDoubleGear definition shown in Figure 4 outputs one solid geometry of two gears of same or different specifications making staggered gears for gear box applications very accessible.

Figure 4.74 :paraDoubelGear grasshopper definition allowing to create staggered gears. Bottom Left: corresponding Rhino geometries of a staggered gear with same sizes (module) and staggered gear with different sizes. Bottom Right: CG rendering of the HSR-5995TG servo and its four stage gear box using single and double staggered gears.

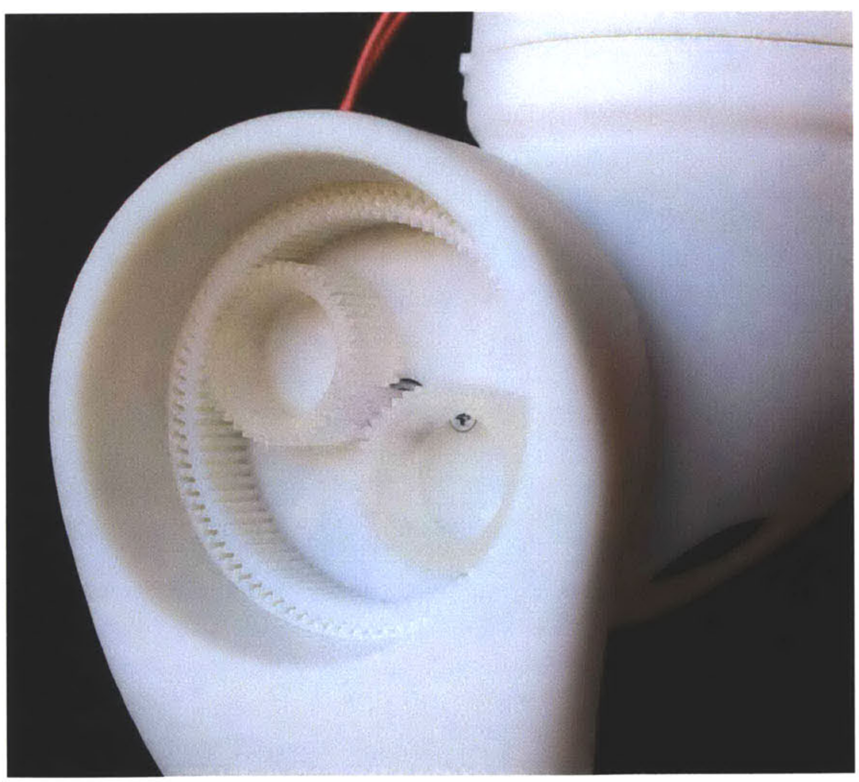


paraServo

The paraServos shown in Fig. 4.74 and 4.76 illustrate parametric harmonic drive gear reducers in combination with large diameter four-point contact-bearings for load-bearing outputs. The paraServo is meant to be directly integrated in larger assemblies thus overcoming conventional component boundaries. The adjacent images show mechatronical artifacts created with the paraServo tool. they show how different reduction ratios, torque and angular velocity settings as well as motor options can be chosen. The motor and servo electronics consist of an Arduino-compatible circuit with an integrated H-bridge and optical interrupter as sensor. The Grasshopper paraServo definition creates all solids required for additive manufacturing including a planetary wave generator, all gears and the housing consisting of a motor and output side in between which the load bearing raceway is located.

The Grasshopper paraServo definition is comprised of multiple paraGear definitions each responsible for different aspects of either the harmonic gear reducer, the planetary gear arrangement of the housing parts containing internal gear rings. Fig. 4.77 shows the un-clustered Grasshopper definition. The inputs

Figure 4.75 A paraServo used in the "Idling" original machine.



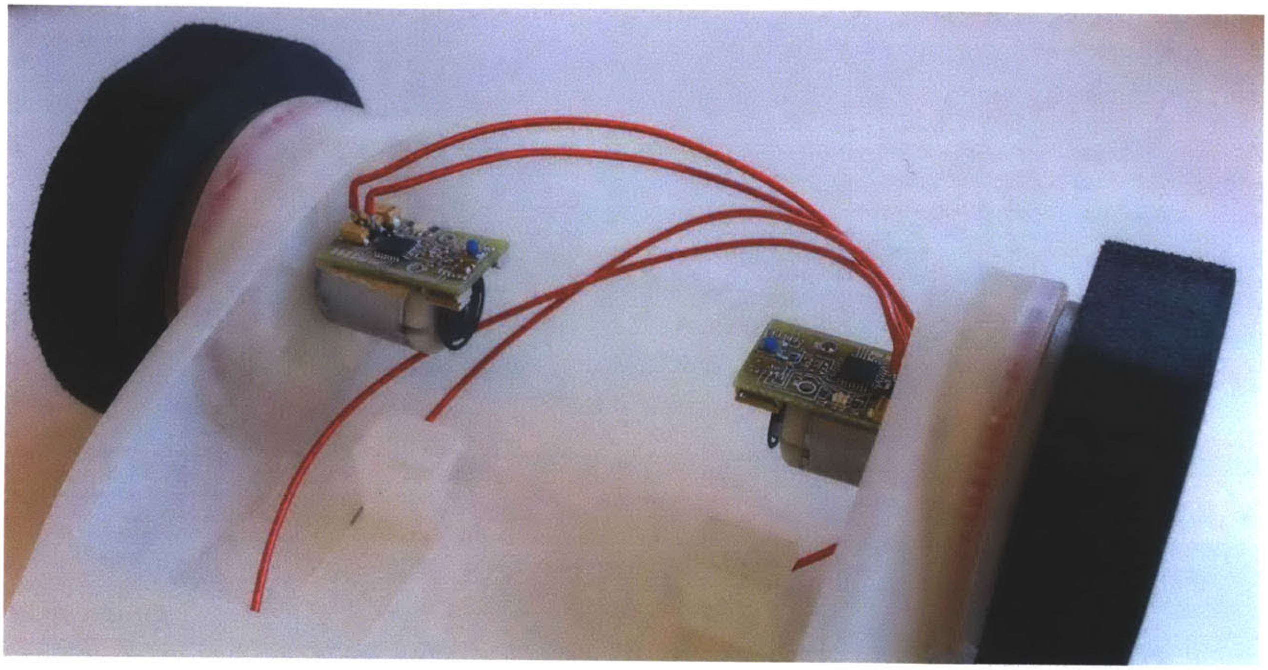


Figure 4.76 (Above) Two paraServos used in the differential drive robot example for original machines.

are located on the left side and feed into the steering section in which various values needed for the overall arrangement are computed. This data is then used to create the curved outlines of the gears and components in the middle part of the definition. The motor geometry is defined as a type of library item which can be addressed by identification numbers. The curve outlines and motor geometry is handed to the solid creation sections where first the spatial arrangement of all items takes place followed by the surface and solid generation. The steering, curve and solid sections are all connected to various outputs of the definition on the right side allowing access to either only the solids or the curves and options needed for tweaking the paraServo.

The clustered version of the definition together with input sliders, toggle switches, specification plus option panels and position and orientation control components is shown in Fig. 4.78. The specification and option panels show data like diameter of the paraServo, overall gear reduction, possible options regarding reduction ratios and more detailed information. The option panels present all possible internal gear options resulting from the selected size. The panel is meant as a look-up table in which horizontal lines have identification numbers which can be used to set the option input. An example of different internal gearing options is given in the bottom part of Fig. 4.78. The

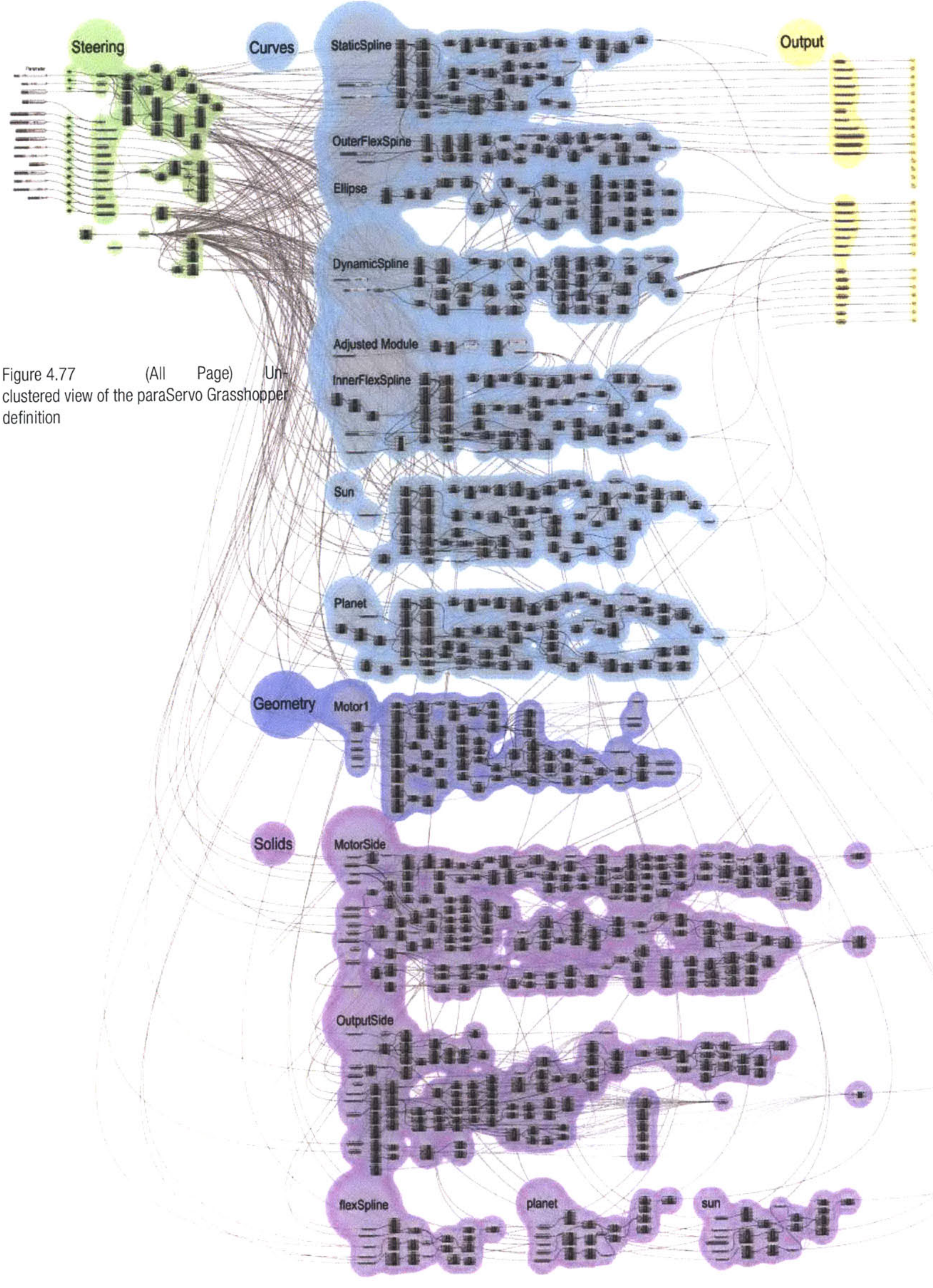
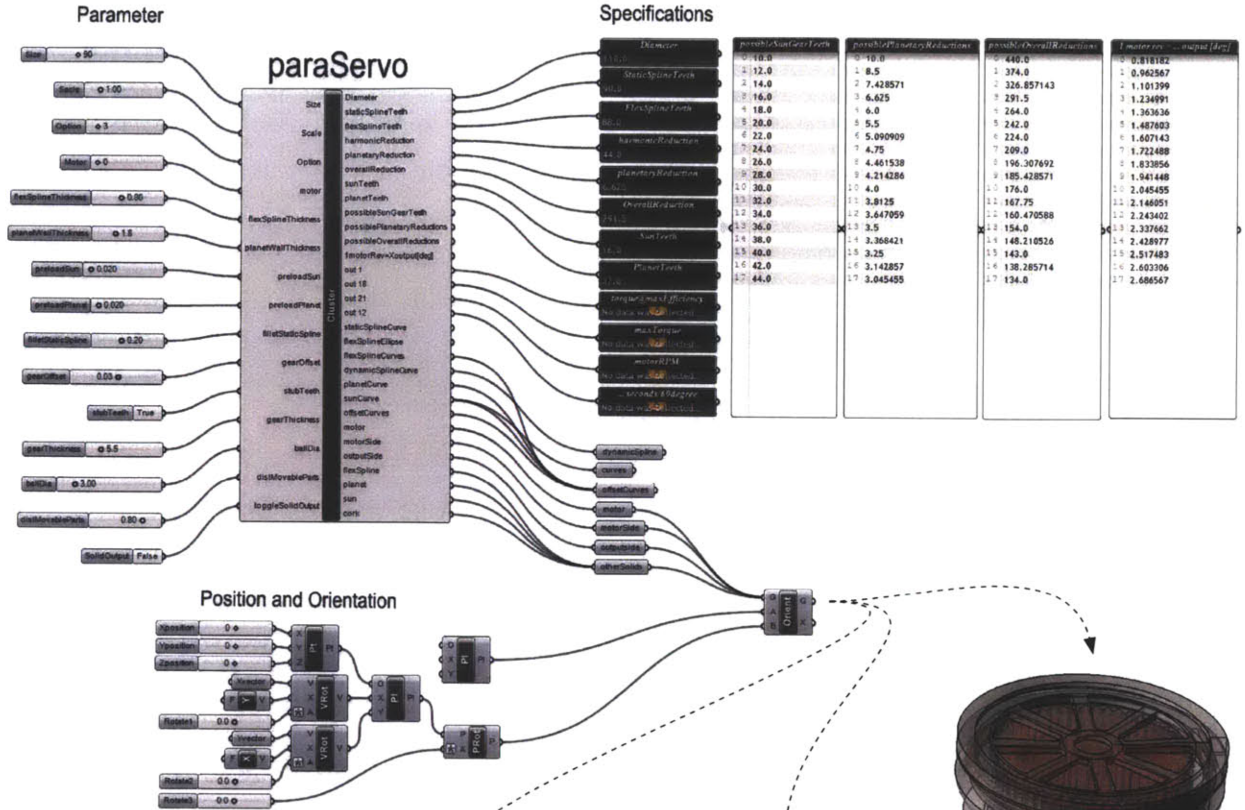


Figure 4.77 (All Page) Unclustered view of the paraServo Grasshopper definition



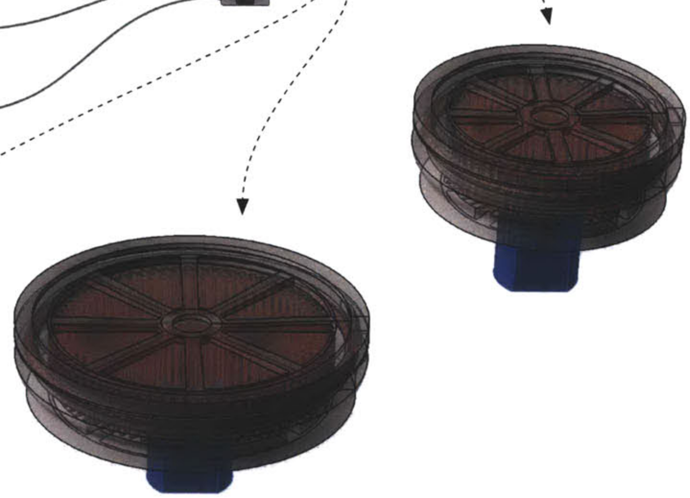


Figure 4.78 (All Page) clustered paraServo grasshopepr definition. Middle: 3 different version of paraServo outputs created by the Grasshopper definition. Bottom: 5 different options of gear-reductions within one paraServo.

black specification panels reflect the actual paraServo values. The diameter is driven by the size and the ball diameter of the four-point contact bearing. The height is driven by the gear thickness and ball diameter. As in the paraGear definition, size represents the number of teeth in this case of the static spline of the harmonic reducer. Scale is the module of the gear teeth. The motor input calls different motors and displays their data in the specification panels. The flex spline thickness determines its wall thickness. This value also drives the factor by which the gear module of the inner flex spline and planetary gears is reduced as compared to the input scale. The planet wall thickness describes the opening in the planet gears by setting the thickness of the remaining material. The preload for sun and planet serve the purpose of canceling any mechanical backlash of the output. The preload is represented by an offset of the involute curve creating one half of a tooth. This offset only affects the involute curve while maintaining tooth spacing to guarantee proper gear meshing. The fillet flex spline input changes the fillet radius of the outer flex spline. The gear offset function affects only the gear curves and allows the user to offset the curves in both directions. This offset is important in connection with certain 3D-printing methods where movable parts need to be separated by a slightly larger gap in order to function. The stub teeth input toggle is also included to account for laser-sintering. In order to maintain a minimum wall thickness, the tip of a gear tooth cannot be pointy but must have a flat face as wide as the minimum wall thickness. While regular gears tend to be pointy stub gears pull back the outside face which helps to make the gears suitable for SLS. The distance of movable parts sets the gap bewteen both housing parts. As in the case of the paraGear, the solid output toggle initiates the instantiation of the paraServo after which updating parameters slows down significantly.

On the output side the various specification and option values can be found and also the curves and solids created by the definition. All the outputs need to be connected to the appropriate panels or container like curve container or solid container within Grasshopper in order to make them visible within Rhino. One container can be connected to multiple instances like all curves can be connected to one curve container which will then show all curves created by the paraServo definition. Also for the position and orientation definition to affect any components they must either all be connected to it or fed into one container which will then feed into the position and orientation change.

paraBearing

The paraBearing component is extracted from the paraServo and only builds the geometry of the four-point contact bearing shown in Figure 4.79. It takes the same input parameter as the paraBearing and can be either connected in parallel to the paraServo to create a bearing with the same specifications or used independently. Shown in Figure 4.79 below is the paraBearing component in combination with the paraServo to create a second bearing supporting the paraServo.

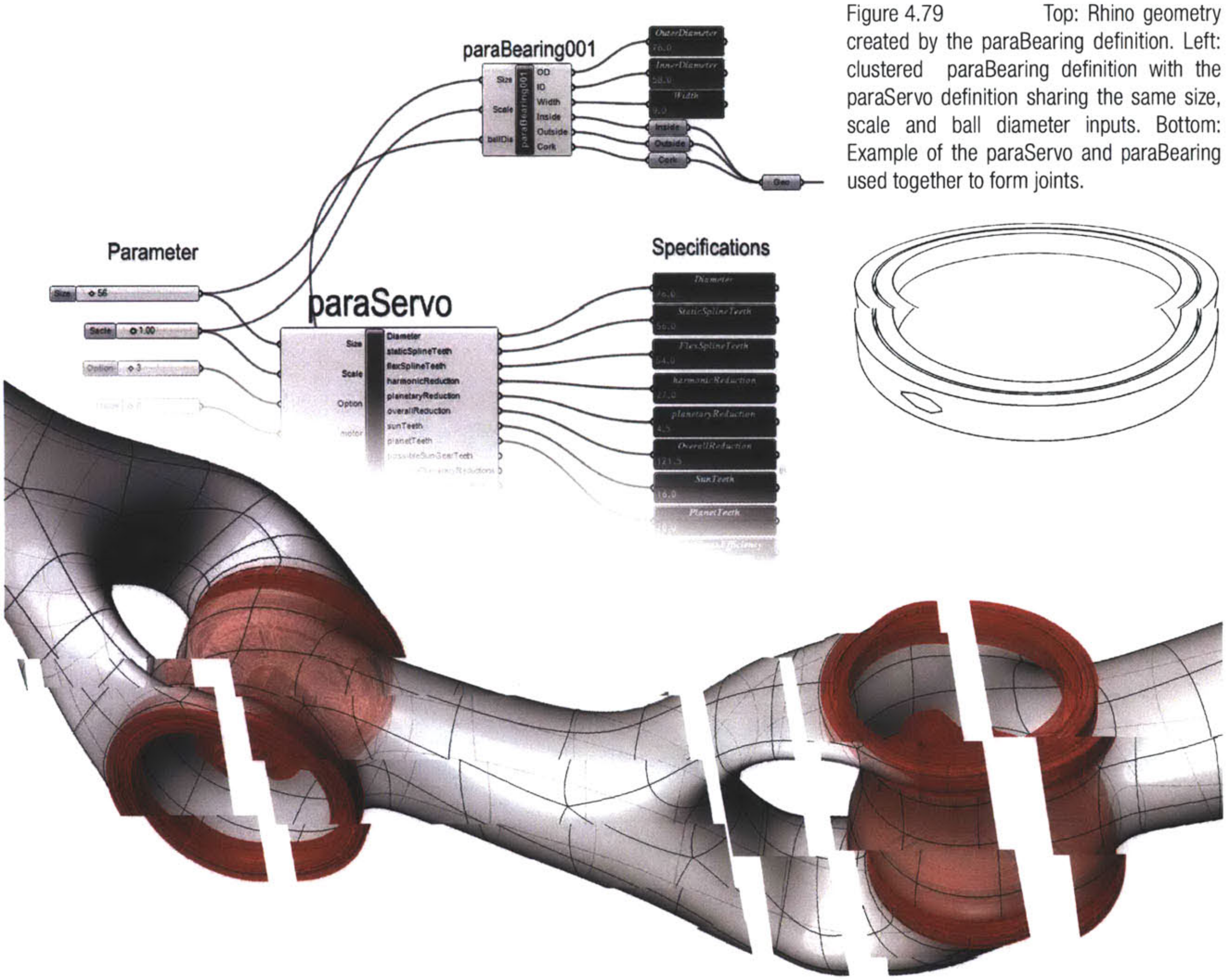


Figure 4.79 Top: Rhino geometry created by the paraBearing definition. Left: clustered paraBearing definition with the paraServo definition sharing the same size, scale and ball diameter inputs. Bottom: Example of the paraServo and paraBearing used together to form joints.

Workflow

The following section demonstrates an originalMachine workflow using the simple example of a differential drive robot. Manual inputs can be tweaked to parametrically adjust the robot's specifications. This straight forward example can easily be expanded to more complex scenarios that link the inputs to constraints originating from other entities. By linking an animation rig to the inputs of the paraServo, for example, the resulting assembly would fulfill the requirements for that animation including such specifications as maximum angle velocity values. Another example might be the model of a robot arm which would drive the paraServo overall gear ration and torque.

I selected the example of the differential drive robot for its simplicity. The DIY robotics design website http://letsmakerobots.com features many examples by beginner robot builders who might also be potential users for the originalMachine tool suite. The first step towards descirbing the originalMachine workflow is a conceptional representation of robot's anatomy. Figure 10 shows some examples of differential drive robots people showcase on "Lets Make Robots" and my conceptual representation. I use words to describe the shape and volume of each component.

The conceptual representation serves as a guide while using and specifying the object-oriented mechatronic tools within Grashopper illustrated in the workflow chart in Figure 11. The vehicle's wheels and drive servos are mirrored symetrically so one paraServo definition is linked to two different location outputs. The first position and orientation component moves the paraServo to X:75, Y:0, Z:48 and tilts it 90 degrees around the Y axis. The second instance moves it to X:-75, Y:0, Z:48 while also tilting it 90 degrees around the Y axis. The paraServo's size, scale and ball diameter also drives a separate paraBearing to ensure that all three wheels are identical.

This bearing has its own positioning component at X:-22, Y:160, Z:48 and is also tilted. The swivel bearing allowing the rear wheel to be a caster following the robot's movements is its own entity with the position X:0, Y:160, Z:70.

Once these grasshopper components are in place they can be baked to become editable in Rhino. The output side of the paraServo housing can be extruded to form the rim for the rubber wheels. Using the T-spline plug-in for Rhino the robot's body can easily be modeled while connecting and integrating the paraServos and paraBearings. Details like sensor cut out and battery fixture hooks can be added before sending all parts to the 3D printer.

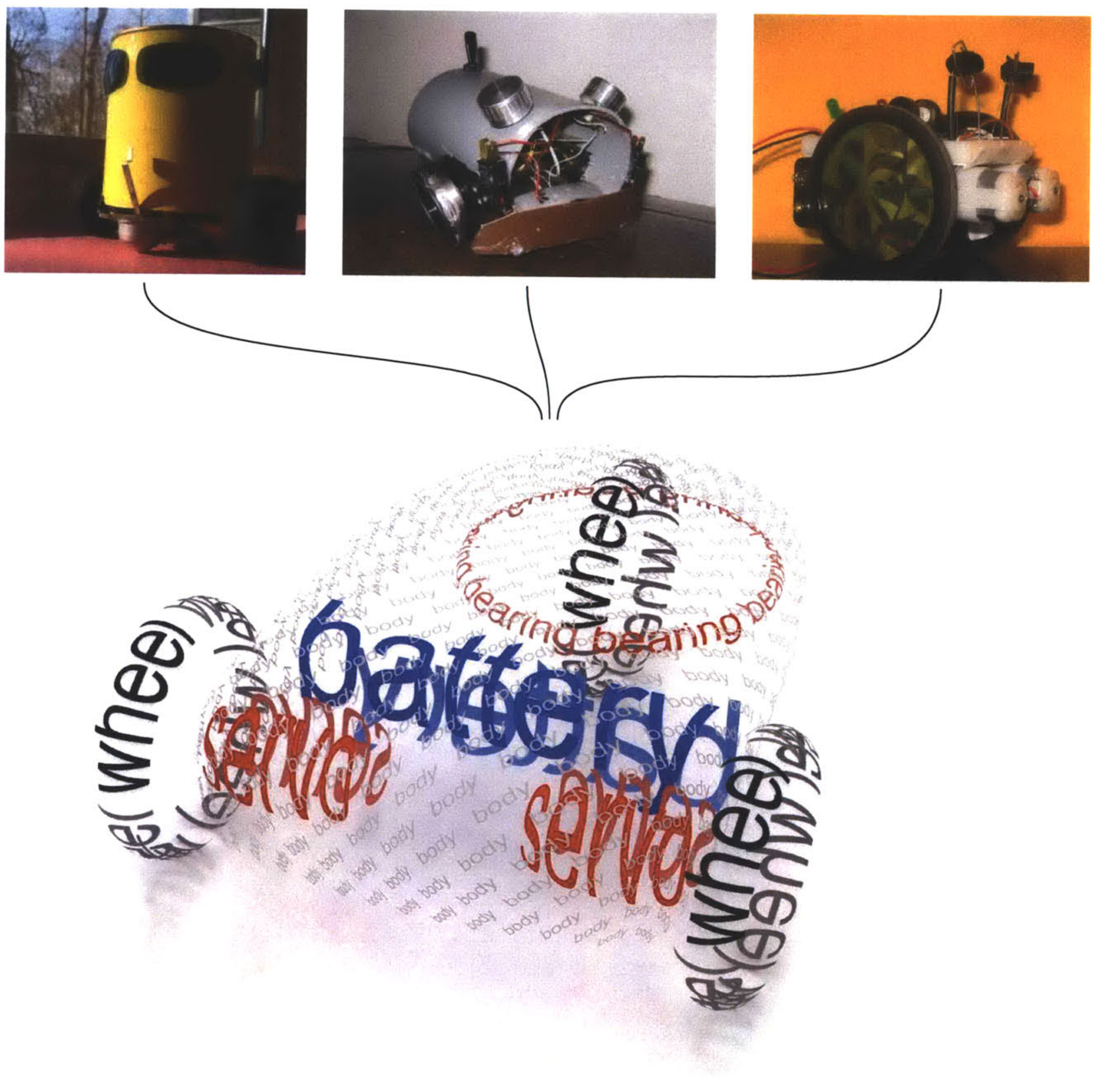
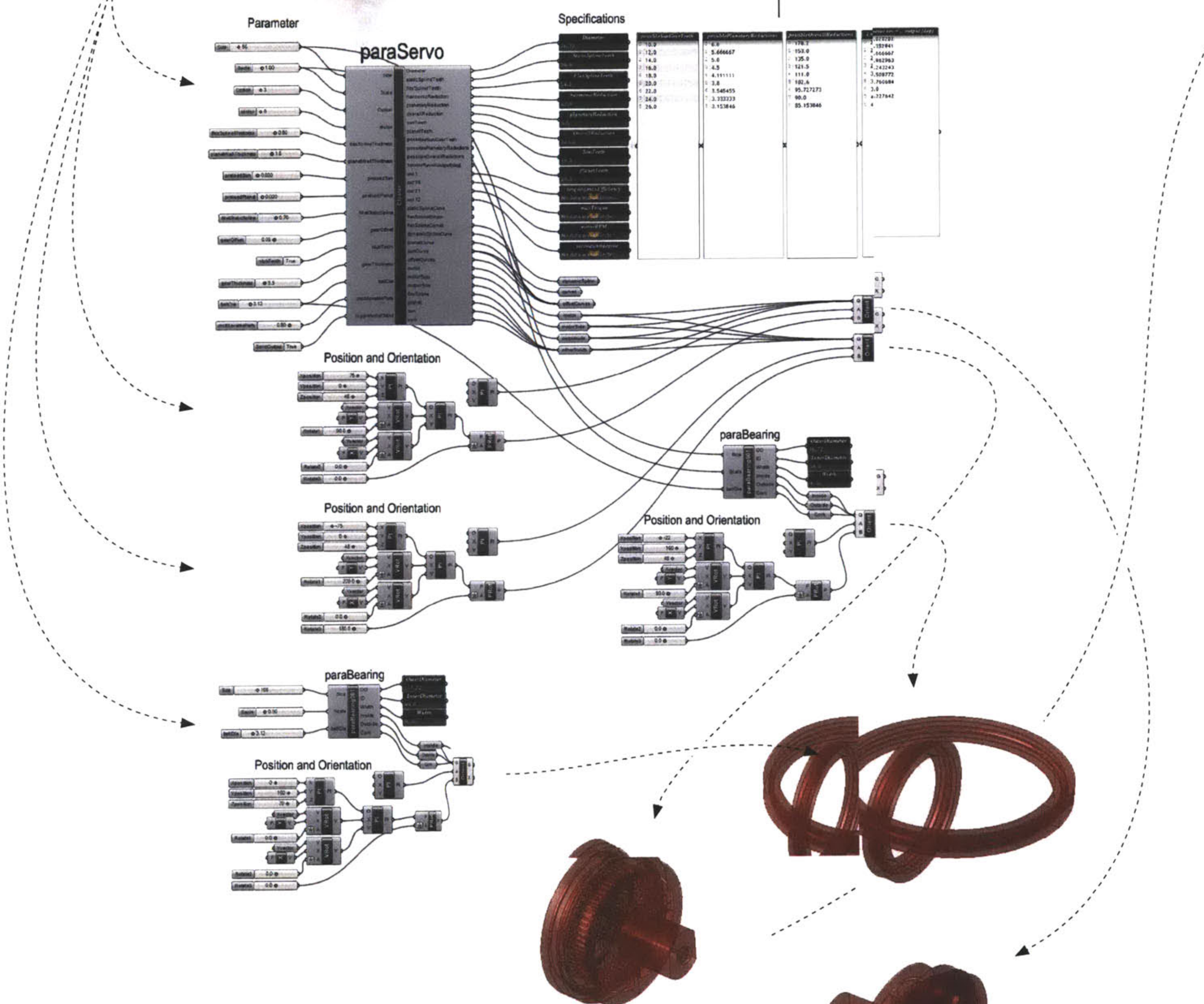


Figure 4.80 Extracting a conceptual representation for a differential drive robot to serve as starting point for the originalMashine workflow example. Pictures on the left origin from http://letsmakerobots.com user names of the authors, top: ingoblegnome, middle: GG, bottom: Rik

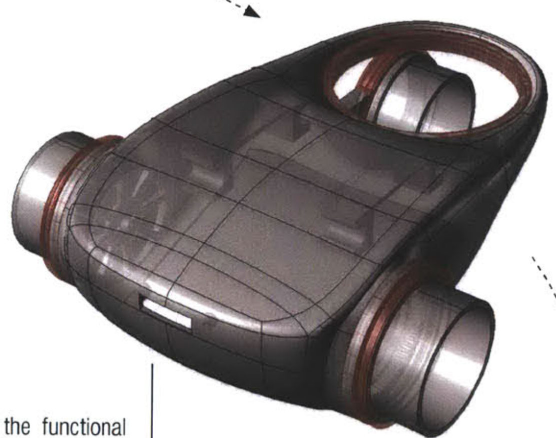
The conceptual representation of the differential drive robot serves as starting point for the originalMachine workflow.

One paraServo component is specified for the project's requirements. The definition generates two identical wheels. In addition one paraBearing with its own position is created to serve as rear caster wheel. An independent paraBearing forms the swivel bearing for the rear catser.



The finished Grasshopper geometry resembling the conceptual representation can be baked in order to become ditable in Rhino.

Adjusting the output side of the paraServo housing to form rims for the robot's tiers.



Modeling the body shape around the functional paraServo and ParaBearing components in Rhino using the T-Splines plug-in.

The Rhino model of the finished robot including sensor cut out in the front, battery fixtures and rear swivel caster.

The 3D printed robot shell with integarted para Servos and rear swivel caster bearings.

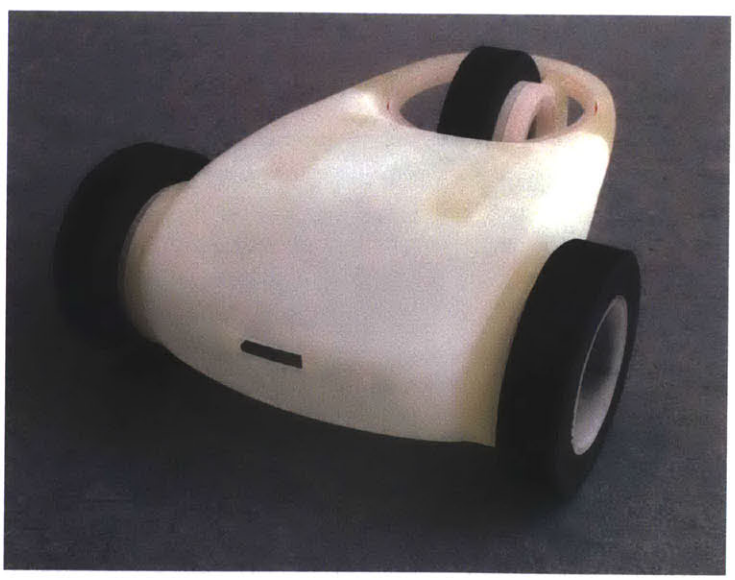


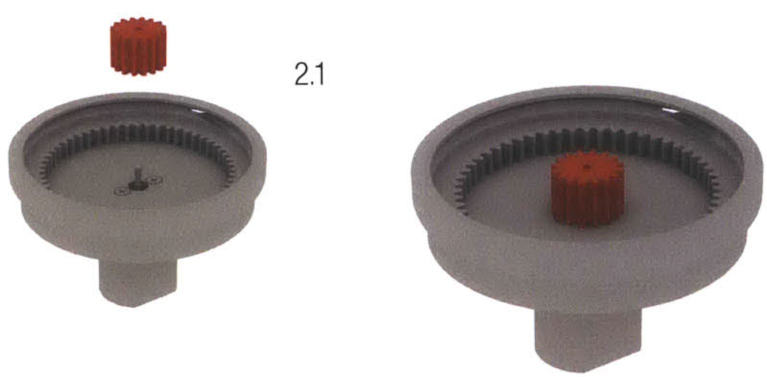
Figure 4.81 Visualisation of the originalMachine object-oriented mechatronic workflow using the example of a differential drive robot.

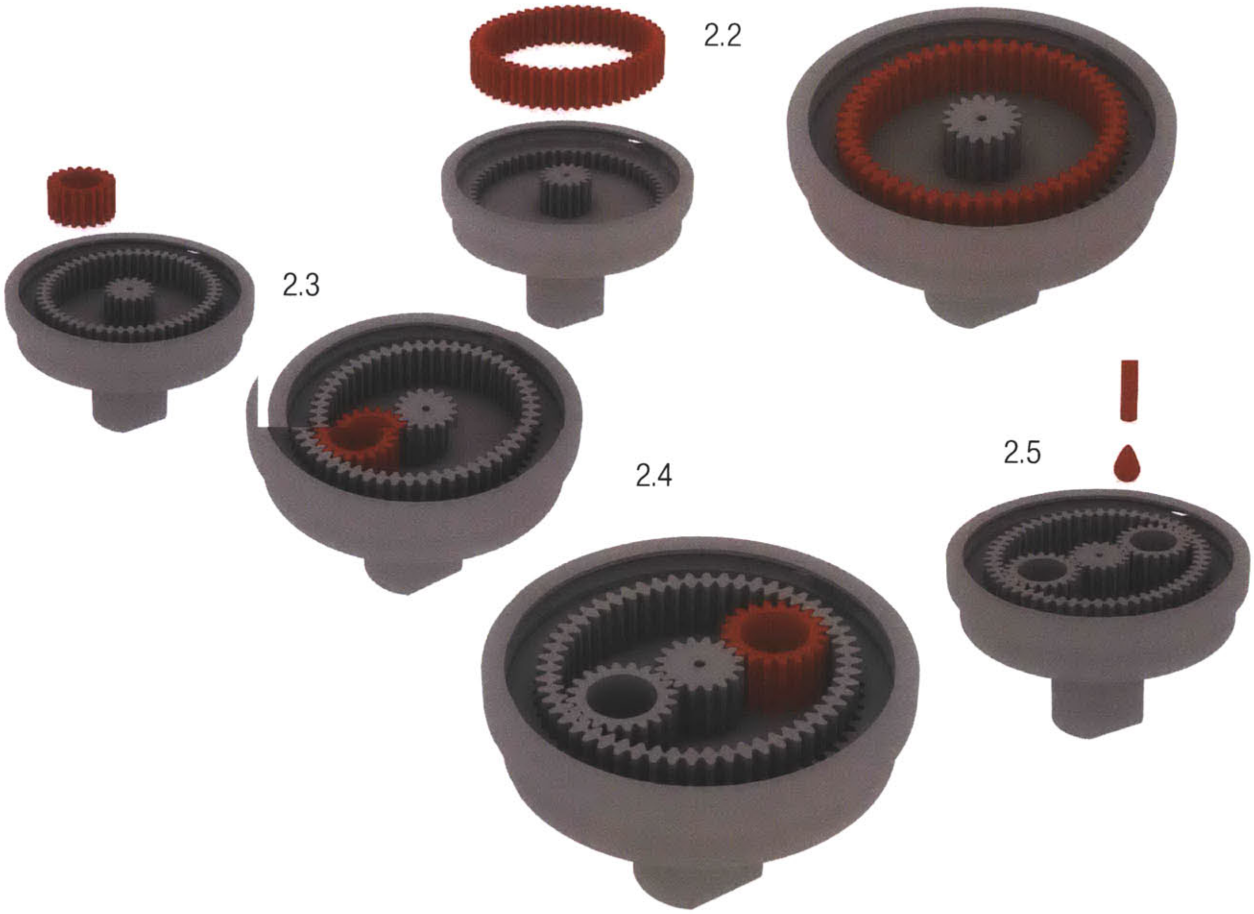
Assembly

Assembling an originalMachine requires four steps. The following illustration shows the assembly process using the bare paraServo component, but it can be applied to any originalMachine derived from the paraServo. The following additional components are required: (1) the electric motor, (2) the motor screws, (3) ball bearing balls, (4) the motor/servo electronics and (5) the sensor disc. The four steps include: (Step 1) mounting the motor using the motor screws; (Step 2) inserting the flex spline and planetary gears together with some lubricant; (Step 3) attaching the output side by feeding in the ball bearing balls and sealing the raceway with the cork; (Step 4) attaching the electronics and the sensor disc. The following CG rendering illustrates the process.

Step 1: Starting with the motor side of the housing push the motor in from the back side and attach it with the motor screws from the front side. The screws should end up flush with the housing bottom.

Step 2: Inserting the gears starts with the sun (center) gear which press fits on the motor shaft. Sliding it on requires force and is best done while supporting the motor shaft on the other side onto a hard surface. Attention is required not to slide the sun gear down to0 far guarantying smooth rotation of the motor shaft. The flex spline and planet gears are installed next by sliding them in from the top. The flex spline is shorter than the internal gear of the housing and needs to mesh at two opposing points. The planet gears are inserted at exactly these opposing positions. At last, depending on the material some lubricant should be added.





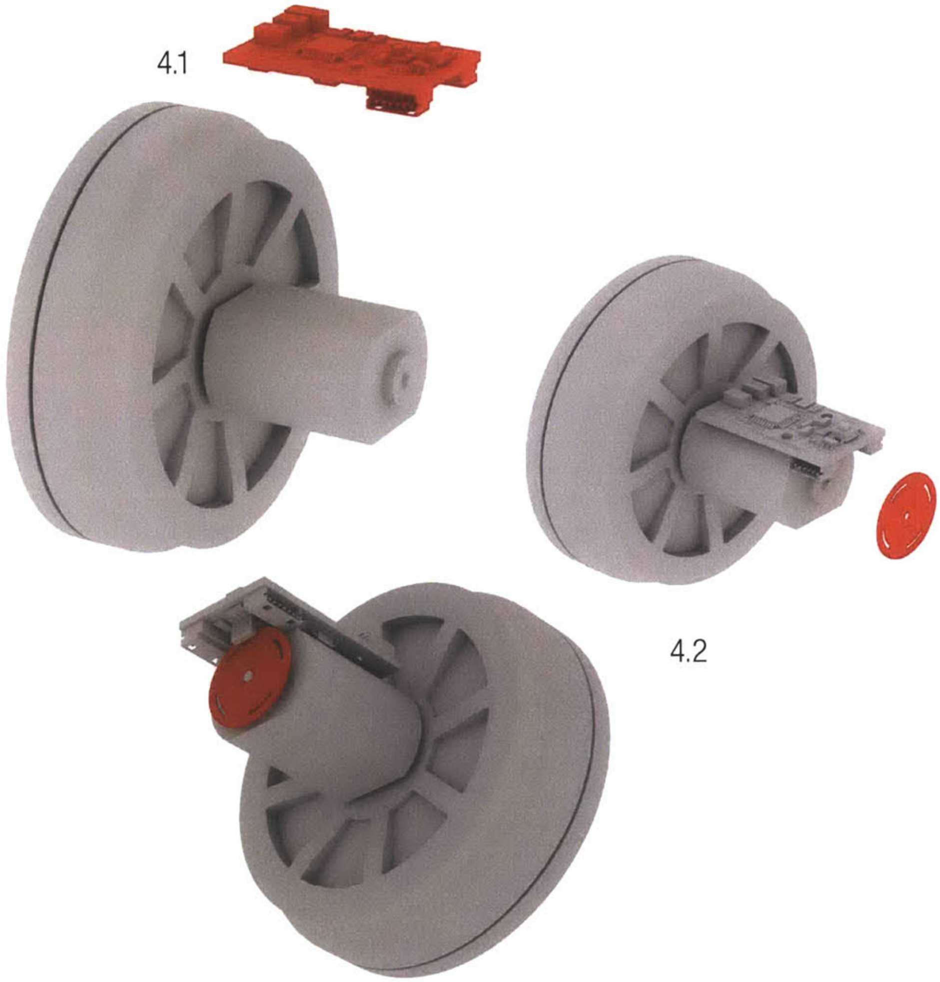
3.1

Step 3: After checking smooth operation of the gears assembled in the prior step and eventually operating the gearbox for a while, both sides of the housing can be combined by positioning the output-side towards the motor side leaving a gap in between. The motor side has a hole forming a channel leading into the bearing raceway. Ball bearing balls can be inserted this way and gently pushed into the raceway. (A ball-insertion-device is in process!) A tweezer and some kind of wooden push rod helps. If the raceway clogs operate the gearbox briefly to loosen the components and try again. Once the raceway is filled with balls the cork can be inserted to properly seal the bearing .

3.3

Step 4: The motor/servo electronics has two through holes in its PCB. These holes need to be aligned with and soldered to the motor terminals. Additionally double-sided tape can be used to hold the PCB in place. This position will also ensure the optical interrupter on the bottom side of the PCB aligns with the sensor disc mounted to the part of the motor shaft extending from the rear side of the motor. A drop of super-glue helps to fix the sensor disc.

4.1



4.2

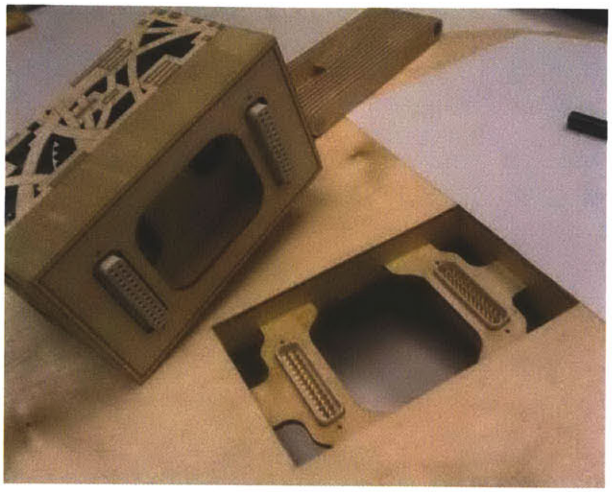
Chapter 5

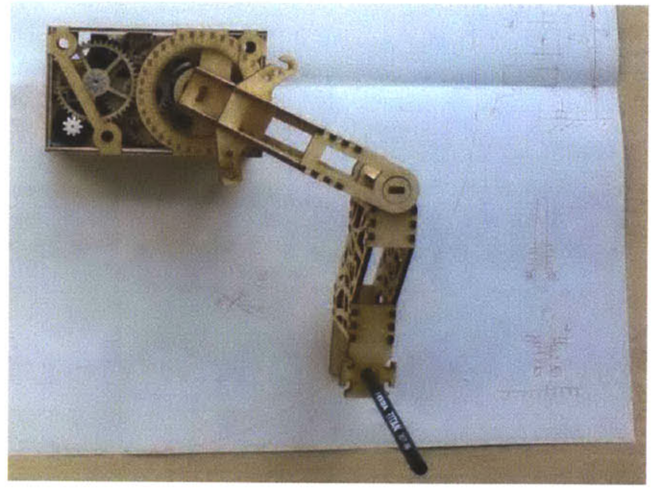
Making originalMachines

The following chapter discusses the outcomes of engaging with the tool discussed in the previous chapter. Some early work is included to trace how certain qualities became important for originalMachines such as blurring boundaries between components. In the first half of the chapter, I describe my own work using the tools, especially two projects entitled differential drive robot and Idling. In the second half of the chapter, I review four case studies of design proposals by a diverse group of makers. I conclude with some reflections on the design process for originalMachines.

Towards originalMachines: The Audiograph

In collaboration with Alex Taylor at Microsoft Research Cambridge UK, I built the "Audiograph" based upon the PlywoodServo mechanism described in Chapter 3. Audiograph is part of a series of artifacts called "rudiments" (Helmes, Taylor, Cao, Hook, Schmitt, P. & Villar 2011). The interdisciplinary research team from design, ethnography and embedded hardware aimed to create surprising and novel objects that have no precedent in order to explore people's reactions to them in a home setting. This requirement seeded an integral quality of originalMachines, namely the integration of various components to create an appearance different from customary home electronics or appliances.





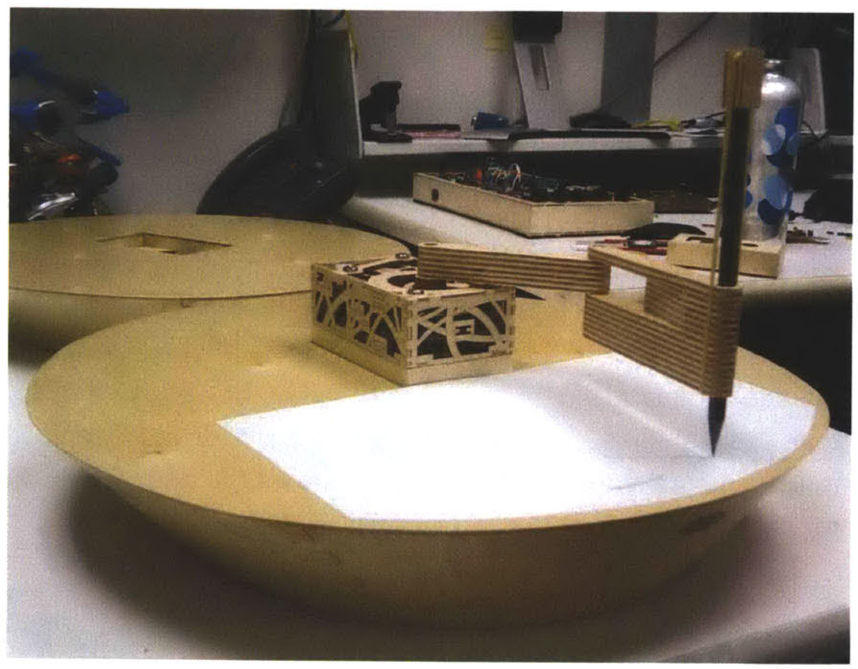


Figure 5.82 Audiograph, an originalMachine prototype based on the plywood servo in collaboration with Alex Taylor (MSR)

To achieve the desired aesthetic and behavioral qualities, Taylor and I used the PlywoodServo as a building block integrated into an assembly of the same material (Fig. 5.82) to create an unusual drawing machine. Sensors in the base of the Audiograph localize the direction from which sounds in the environment are coming. The PlywoodServo at the center of the object then moves an arm towards the sound. As the arm moves it draws a line with a pencil and distorts it as a result of a free hinge in the middle of the arm. The images above show CG renderings of the design process, an early prototype, a detail of how the plywood servo connects with the base and the final Audiograph as it was placed in two households homes.

The PlywoodServo proved to be a successful integration of mechanical motion enabling components with overall object design using rapid prototyping technology. Rather than interconnecting black-boxes it achieved the celebration of its constituent parts and sub-components. As such the Audiograph can be seen as an early example of an originalMachine with the specific intention of creating an unprecedented (in terms of affordances and functionality) and unique piece of machinery (in terms of aesthetics, motion, and responsiveness). It explores machine and apparatus expression in different ways like using plywood as material and creating "in-accurate" or seemingly unpredictable, distorted drawing as intended function. Even though the level of component integration is high some traditional boundaries still exist between the servo as a unit integrated into a base and the drawing arm. These results and the object itself inspired the initial move towards creating new types of machines and mechatronic artifacts.

Design Explorations

The Audiograph presages some of the key goals for originalMachines such as component integration discussed above. However, there are still some missing elements such as the dependencies on subcomponents like the ball bearing on the output axel between the PlywoodServo and the drawing arm. In addition to the components, the overall shape and kinetic qualities of the object cannot be easily changed. The physical appearance and form factor are drawn by hand in a CAD program. The electronics are completely separated from the mechanical components. Any changes throughout the design require manual adjustments throughout the object. Though I had a rich ongoing discussion with Alex Taylor during the design process, there was no shared object definition that both of us could modify directly. Thus any changes were filtered through my own interpretation of the discussions we had. The object-oriented tool set provides this missing flexibility. The following two examples - differential drive robot and Idling - demonstrate the possibilities for creating originalMachines with the tool described in Chapter 4 and drawing on the lessons from Chapter 3.

The two examples represent two extremes on the spectrum of what might be termed an originalMachine. On the one hand, differential drive robot exemplifies how a relatively simple and familiar robot might be translated into the language of originalMachines. It provides some continuity with the existing domain. On the other hand, Idling is a completely unprecedented object with no particular functional requirements much like the Audiograph.

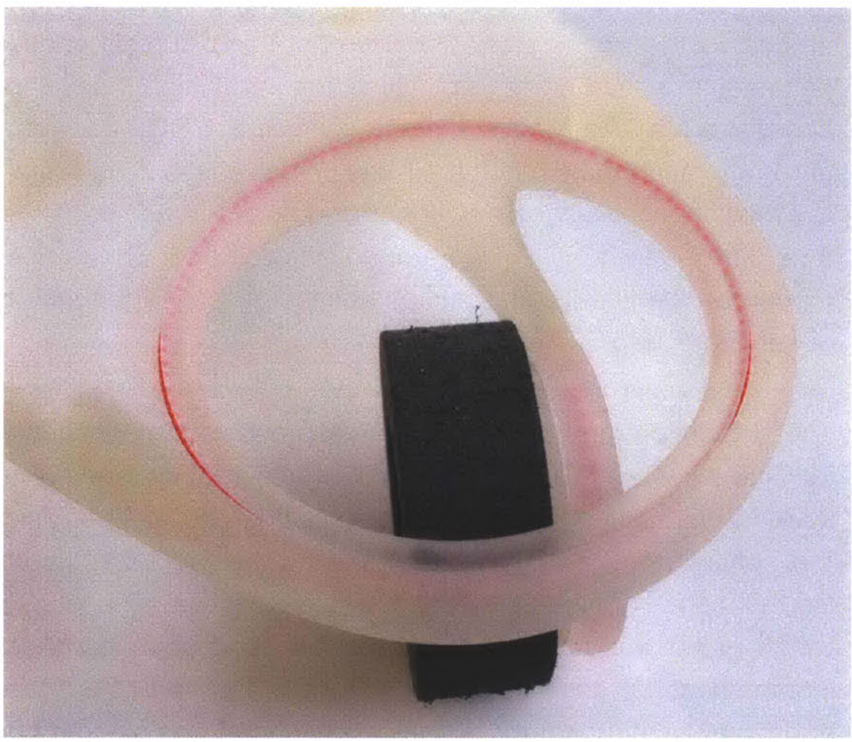
Differential Drive Robot

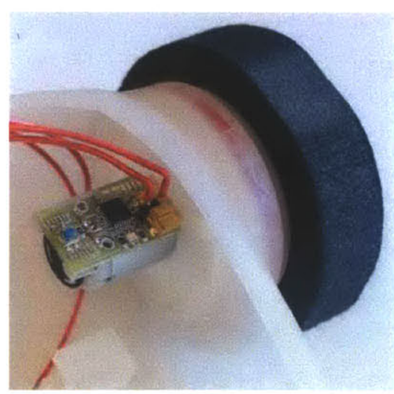
The idea for the differential drive robot as an early example for originalMachines and the corresponding workflow originated form observing the web page letsbuildrobots.com where many users build this type of robot at some point though out their carriers and development process. It is also frequently used teaching robotics. As a type of machine the differential drive robot has a relatively simple typology which is good for an initial example. The conceptual representation shown in the adjacent images (Fig. 5.84) derives the main components and shows they in a spacial context. The robot has three wheels two of which are the co-axial drive wheels and the third one is a free spinning caster which can be in the front or back. Some types use two casters on in the front and one in the back resulting in a four wheeled (diamond configuration) robot. The drive wheels are identical in terms of motor, gearing and wheels diameter. The control of the robot results from the identical co-axial drive wheels which can move the robot forward in a straight manner while spinning at the same speed or steer the robot while spinning at different speeds. The body of the robot needs to house the power source which in most cases is a battery, the drive, control and communication electronics and sensors needed for navigation. In this case the paraServo electronics can be augmented into the control electronics by adding the control code into the servo code of one of the wheels.

Figure 5.83 (Above) melting and extracting the Invisin PolyJet 3D print in corn oil.

From the conceptual representation the Grasshopper based object-oriented mechatronic components are chosen and given spacial positions forming the functional base of the robot. Chapter four describes this process in more detail. The body of the vehicle is then modeled in Rhinoceros connecting all components like bearings and servos. I have chosen a fairly large ball bearing in the back to serve as swivel bearing for the rear caster as large diameter ball bearings are unconventional, rare and not used very often in mechanical engineering and design due to cost and affordance reasons. The body is modelt around the battery volume including hooks to attach the battery using rubber straps.

3D printing the entire robot with all parts happened in one go using the Invision PolyJet machine and took about 21 hours.





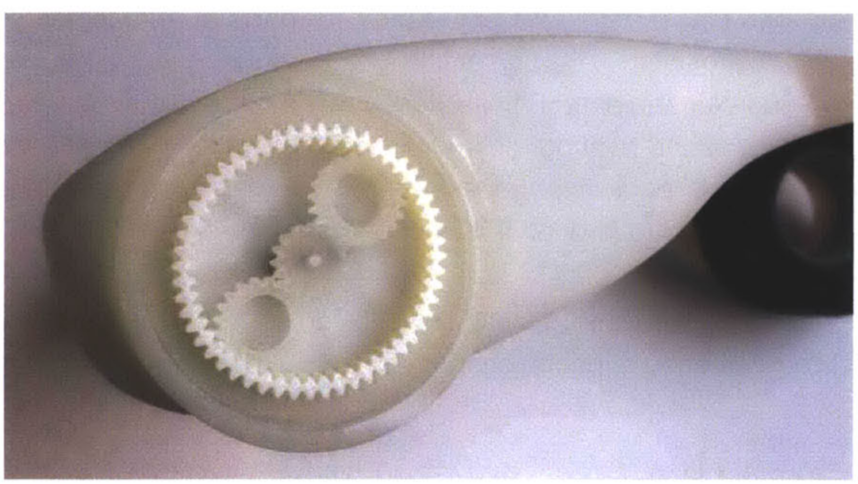


Figure 5.84 (Above) Differential drive robot, completely 3D printed, made using the originalMashine tools and workflow integrating parametric harmonic drive servo, balls bearings and overall body.

Post processing happened in a bath of corn oil as shown in the adjacent images (Fig. 5.83). The assembly followed the steps layed out at the end of chapter four and basically involved feeding the balls into the bearings to connect the movable parts and attaching motor, gears and electronics to the servos. Battery and sensor mounting required some filing as the PollJet tends to create rather tight dimensions.

Idling

Idling is an example of an originalMachine involving multiple degrees of freedom with the intention to show the organic and fluid integration of actuators, bearings and a robot body. To emphasize the organic properties the robot links a cubical wooden pedestal and head block from which and into which it develops and dissolves. The robot's anatomy mimics a 4-DOF neck joint giving the head cube a human-like expressiveness. Idling refers to a kind of behavior where no concrete goal is pursued but rather a random pattern of "looking around" is executed. This type of motion evokes life-like associates enhancing the experience of observing "Idling".

The design process followed the workflow pattern described in Chapter 4 where a conceptual representation served as starting point to engage with the object-oriented mechatronics tool sets within Grasshopper. After baking the various actuators (see Chapter 4 for details) and bearings the T-Spline plug-in within Rhino was used to model the fluid and organic shapes interconnecting the functional parts. This process took approximately four iterations until a final shape was selected for 3D printing. The first attempt to print the parts at the Media Lab Fab Lab resulted in the printer failing which subsequently required outsourcing the print. Primarily due to time constraints, SLA was chosen as the printing method. However, as already mentioned in the cook book this process is also very high resolution and relatively strong (though not as strong as laser-sintered parts).

The aesthetic and kinetic results of Idling exceeded the predetermined expectations:

- very high resolution
- enough flexibility for the flex spline inside the paraServo
- superior surface finish
- dimensions perfectly matched CAD
- ease of assembly thanks to appropriate tolerances

The adjacent images show the design process and iterations staring with one DoF and developing towards the 4 DoF neck joint configuration. Also shown is the relation between Grasshopper tools, Grasshopper-created geometry, Rhino geometry, the final Rhino model and the finished 3D printed originalMachine. The assembly process is documented and also shown together with two details of the machine.

Unfortunately, the scale of the machine (150cm tall in total on a 18cm by 29cm floor area) required more powerful motors that did not run on the already developed motor boards. In its initial form Idling was operated using on-off switches for each individual motor. The lack of controllability at this stage still makes it difficult to reduce the noise Idling makes while moving. However, this issue could be resolved with another iteration of the motor boards which would also make the object more autonomous.

Figure 5.85 (Above) Iterating Idling from 1 DOF to its final 4 DOF shape.

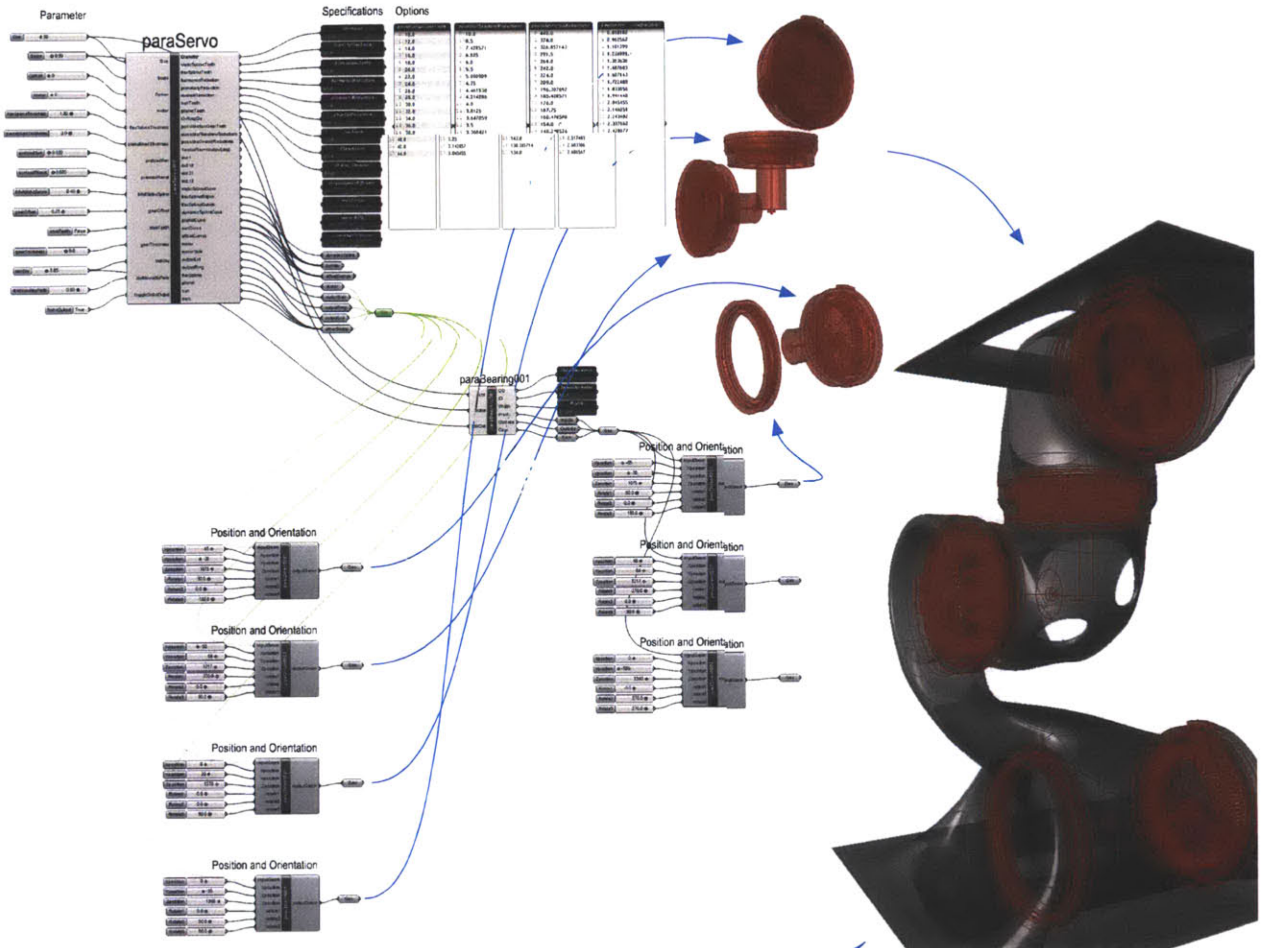
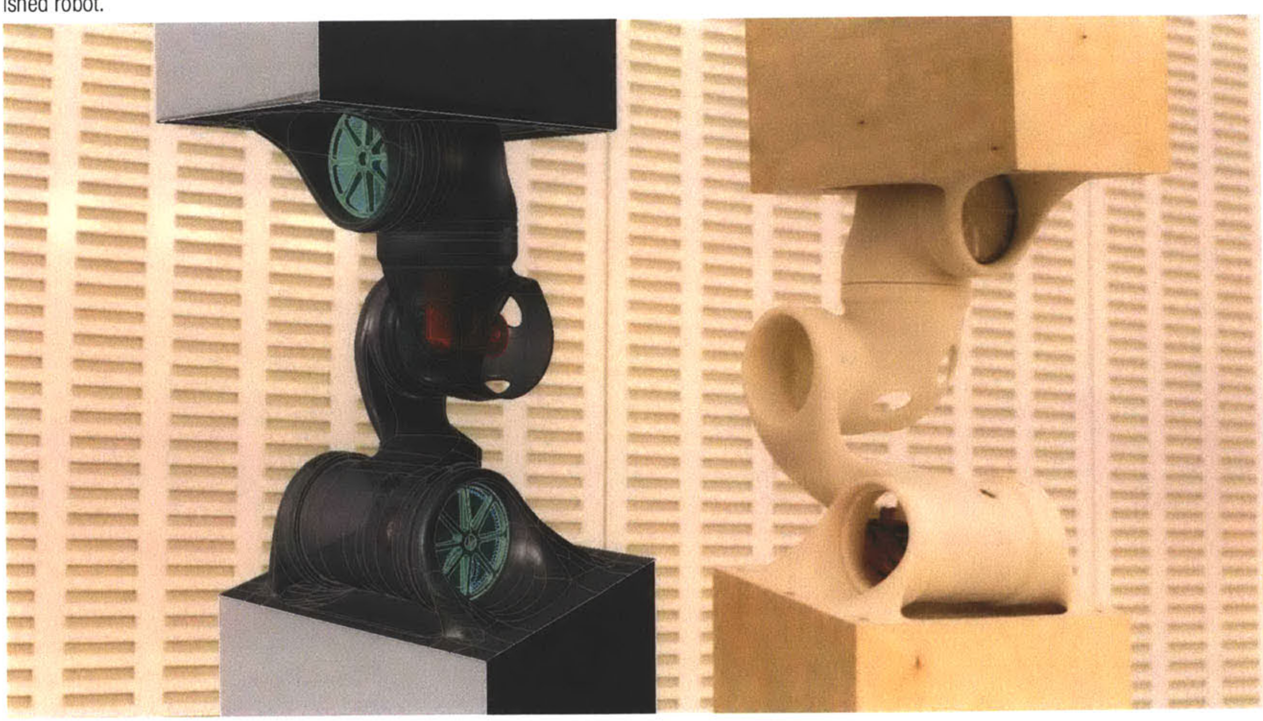
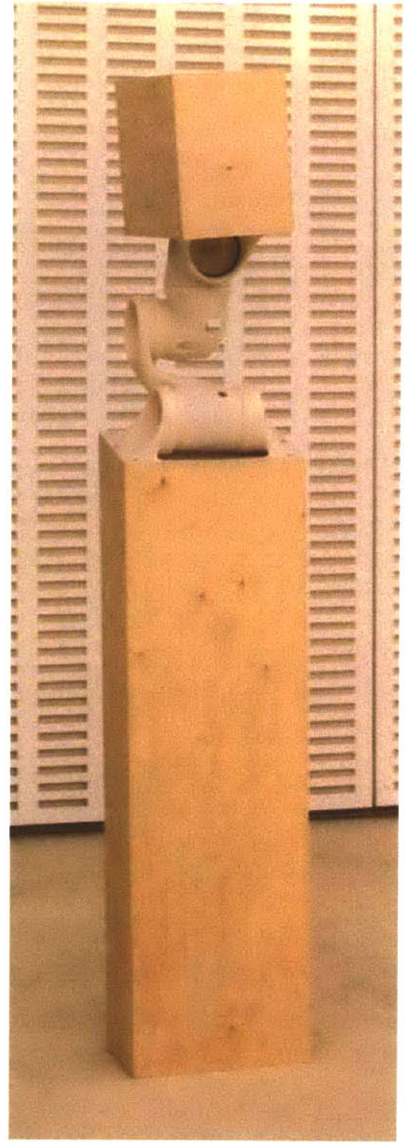
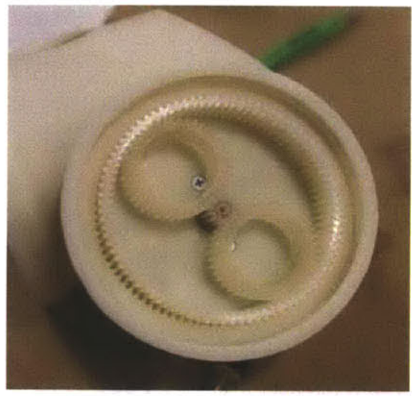
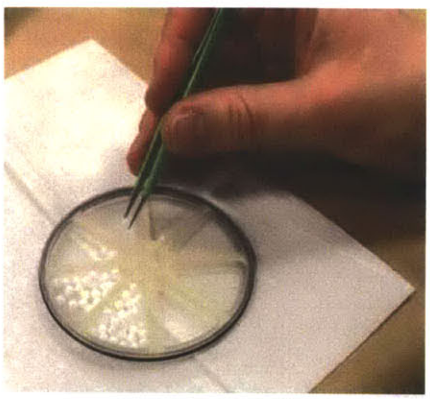
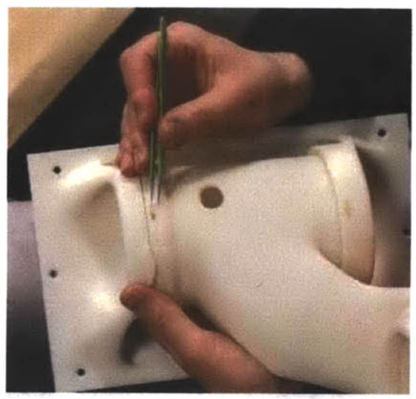
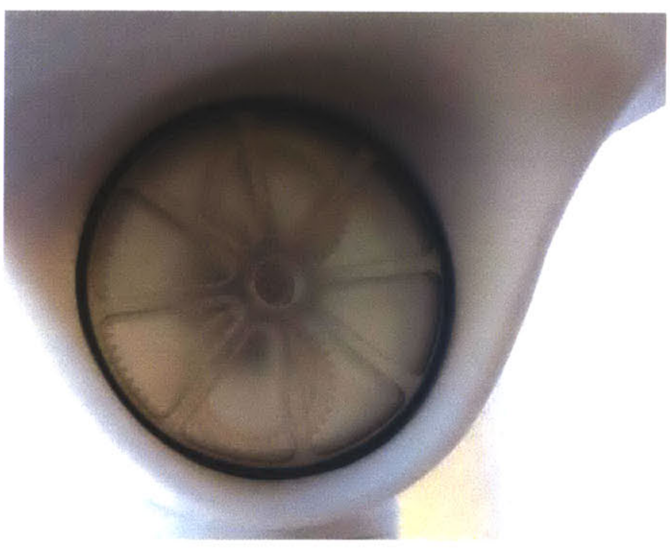
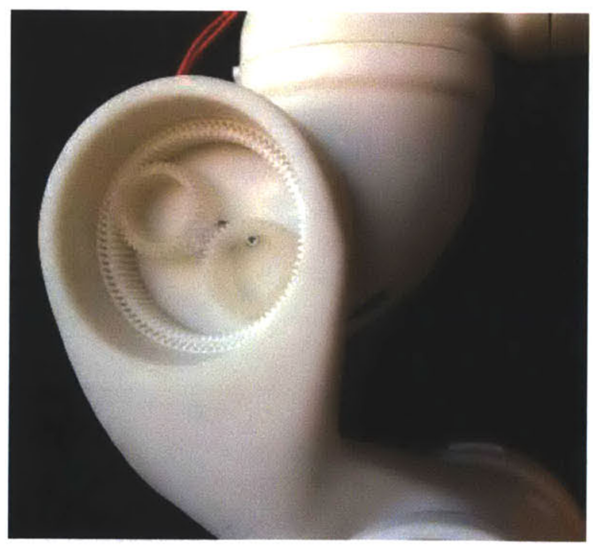


Figure 5.86 (Both Sides) Idling originalMachine design, assembly and finished robot.



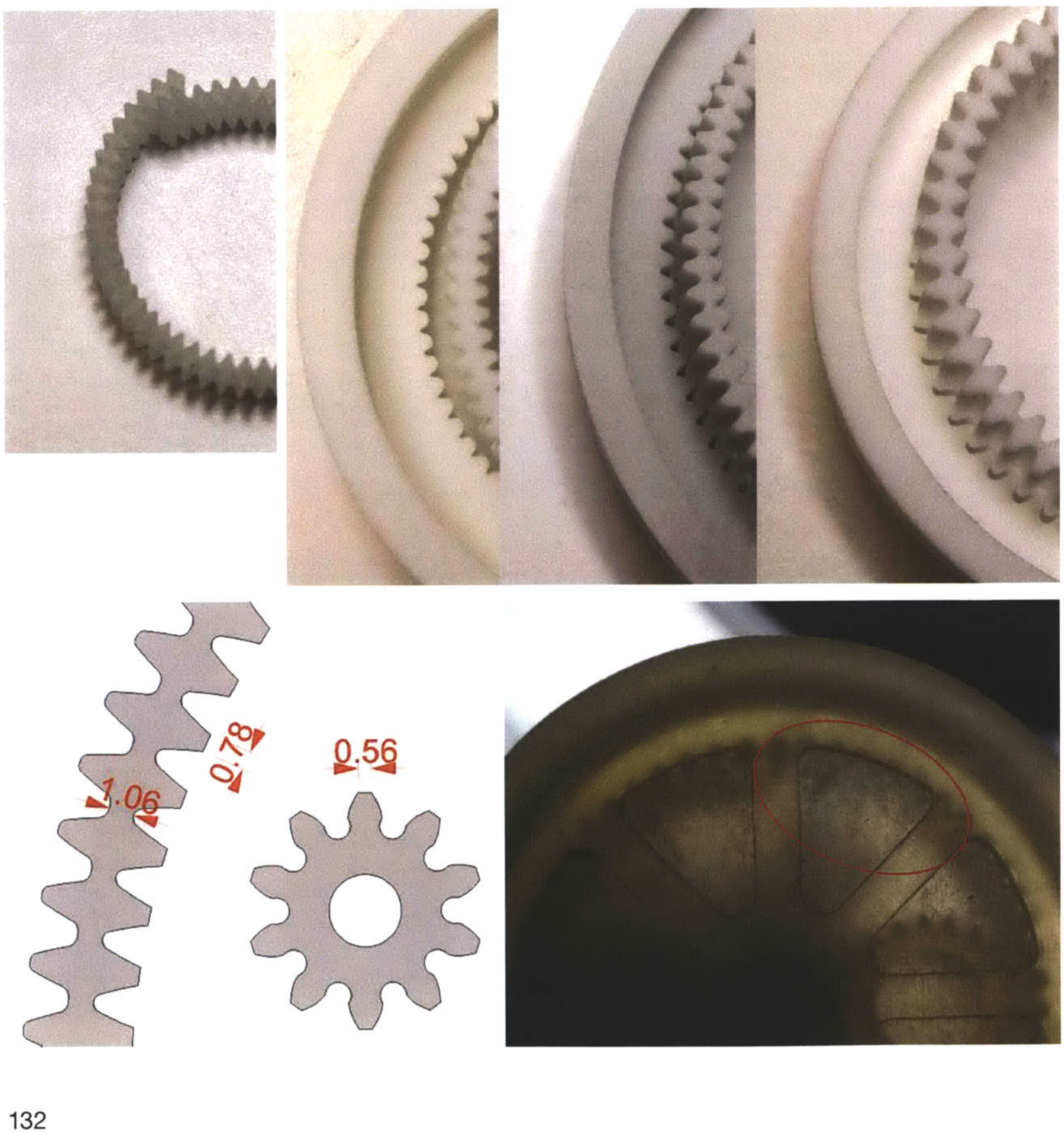



Challenges of 3D Printing

Both the differential drive robot and Idling heavily rely upon the integration of the parametric CAD definition with the particularities of the selected 3D printing method. As described in Chapter 4, the design space includes several variables such as offsets or gear tooth shape that can be adjusted based on the different methods available. Throughout the process of creating the originalMachine examples, there were many setbacks related to the particularities of the methods. SLA is the only method that always resulted in perfectly sized parts. Polyjet parts all had significant (and prohibitive) fatigue and breaking problems while

Figure 5.87 (Below) examples of fatigued flex splines printed on the Invision PolyJet and examples of various SLS 3D prints showing different offsets all rendering the harmonic drive dis-functional.



the strongest material SLS showed the most variation in terms of offsets and seemingly unpredictable gaps. In this area, each service provider (no SLS machine is available in house unlike the other two) apparently adjusts to how he or show perceives the requirements of the customer base.

In general, many 3D printing services do now seem to make it easy for users to exploit the full potential of a particular method. For example, Shapeways does not allow users to print at the highest resolution possible on the SLS printer. This limitation may have sound economic reasons, but it makes it impossible for the user-driven design domain to reach the quality and capabilities apparent in more industrial settings.

Design Explorations with Users

In addition to my own design explorations, other users were invited to use the object-oriented mechatronics tools. Again initial attempts to introduce the possibilities of customized mechatronics emerged from work with the PlywoodServo.

Initial Workshops with PlywoodServo

In 2008, I co-taught the MIT Media Lab introductory course (Media Lab 123) which was an interdisciplinary introduction to the primary skills required for creating projects that integrate many hardware and software elements. Tangible media, for example, requires knowledge of embedded hardware, interaction design, industrial design, and perhaps sociology or psychology. As a result, the course included different chapters on materials, electronics and software. I used the PlywoodServo and associated user guide (see Appedix) to teach students about mechatronics and demonstrate how they might be able to create their own unique objects. Most of the course participants

Figure 5.88 (Impression from the Original Plywood Actuation workshop at Princeton University (November 2010)

assembled the mechanical components of the PlywoodServo. However, the hand-soldered wood PCB board proved too challenging and difficult to convey. Still, most students were extremely interested in creating their own custom objects with precisely matched motions and affordances. It now seems clear that a more accessible design tool would have been beneficial. And traditional PCB boards customized to the form factor of the PlywoodServp (since then created and described in chapter 3) would not have compromised the flexibility of the system.

These small piggy-back motor boards were first introduced at a workshop a Princeton University and Prof. Axel Kilian in 2010. The "Original Plywood Actuation" workshop gave graduate students in architecture an exposure to creating their own moving artifacts. With many students having a keen interest in actuated architecture the workshop further reinforced the importance of a tool that embeds knowledge and constraints particularly related to mechatronics. As in Media Lab 123, however, the group also spent the bulk of its time assembling version two (cycloidal) of the PlywoodServo over the course of the day-long workshop. Though no specific objects were created the students suggested many creative possibilities for building and customizing their own mechatronics. For example, one student wanted to build a string robot using a laser-cut plywood chain actuated by the PlywoodServo. Another student imagined actuating her physical model in an architectural studio. Her building included a kinetic sound reflector for different modes of operation. Rather than only showing the states separately the added PlywoodServo model would have allowed her to actuate the physical model.

The workshops both demonstrated the need for a more flexible design tool that could assist users in defining their specific requirements. Though limited to users with CAD experience, the simple ability to map all the constraints driving a particular design benefit the individual or team of creators. The following design explorations emerged engagements with four users with particular ideas and requirements for objects that could be referred to as originalMachines.

Example 1: Feather Hat Gear by Amit Zoran

Amit Zoran, an industrial designer by training, was looking for ways to include actuation in his creations. Not having particular knowledge of power transmission Amit was able to include the paraServo in his Cyborg Chief's feather hat gear without learning about actuation principles and requirements beforehand. The feathers making up the hat are meant to protect from the sun and are stowed away on the back side of the chief's head as shown in the series of renderings he created. Through simple rotational actuation all feathers spread out evenly around the chief's head in case of excessive heat or certain ceremonial requirements.

Figure 5.89 Series of renderings of the Cyborg Chief's feather head gear by Amit Zoran.

Very familiar with CAD, Amit elected to represent his idea as something integrated with the chief's head rather than a separate actuate feather hat as originally planned. This integration with a sculptural element allowed him to mount the paraServo component in the ear and the gearing mechanism on the inside of the head creating something more like "head gear" than "hat gear". The resulting object qualifies for 3D printing of all components though the print has not been executed yet.

Example 2: Dumple Bot by David Robert

David Robert, a member of Prof. Cynthia Breazeal's Personal Robots Group at the MIT Media Lab, is involved with an educational research project using a simple cube-like robot using the differential drive robot anatomy described above. The robot plays a role within a mixed-reality learning environment which requires it to interact with many different children and adults in various contexts. The box-like appearance gives the robot the ability to easily adapt to any requirements because the form itself does not limit users to thinking about a particular, limited purpose or set of capabilities.

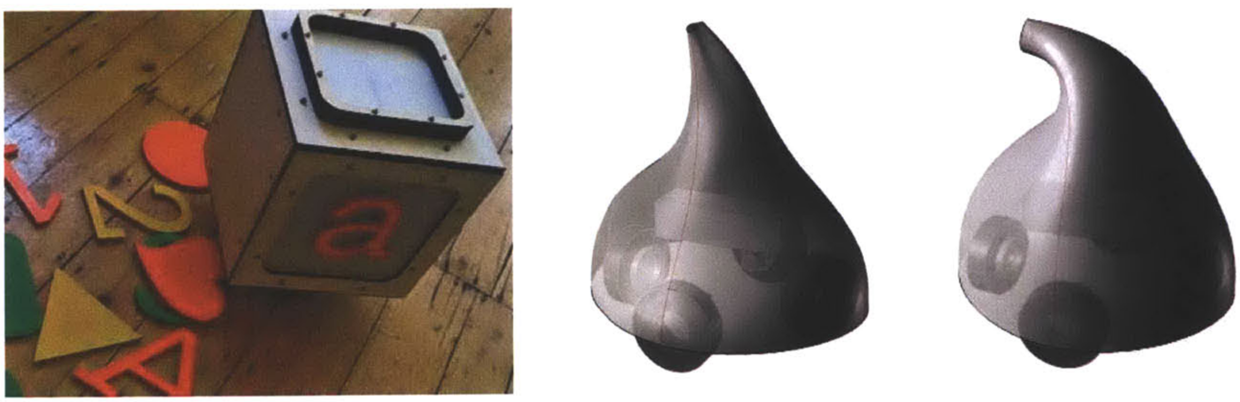


Figure 5.90 (From Left to Right) Alphabot by David Robert, two iterations of Dumple Bot

In brainstorming on the object-oriented mechatronics tool following a demonstration of its parametric nature, David proposed building Alphabots that are easily tailored to a particular audience. Every user would be able to draw up to four lines which will then be lofted into a robot body shape by a paramatric setup also involving the paraServo tool. The resulting Dumple Bots will be different for each users input and completely create themselves through the parametric backbone. For David the outcome is a kind of "procedural robotics" where particular groups or individuals can engage with their specific Dumple Bot that is not duplicated anywhere.

The shape selected for the Procedural Dumple Bot is still closely tied to its original box-shape shell. However, using profile curves the corners and top are more organically shaped. The dumpling shape thus results from modifying the original cube and making it possible to customize future iterations of the robot.

Example 3: Dis(Course) 4 Design Development for Actuation by James Coleman

James Coleman, a graduate student in the Dept. of Architecture at MIT, developed a 3D surface joint made from flat sheet material adapting to a wide range of complex surface curvature. This system is intended to be passively configured into application-specific requirements. Integrating actuation using conventional approaches would result in a huge undertaking. The originalMachines approach was a perfect match because James had already developed his project in Rhino and Grasshopper including simulations which could easily be adapted for actuation control. James iterated on his original joint design by augmenting each one with an originalMachine paraServo component connected to the three sides of the joint by rods. A circular motion contracts or extends the rods forcing the joint to open or close. Arrayed into a surface this method holds the promise of creating endless variations of curved surfaces driven by external sensors or pre-programmed cycles depending on the context. For example, in its current location (see adjacent photograph) the entire structure could potentially contract and expand as people walk down the stairs around the suspended surface.

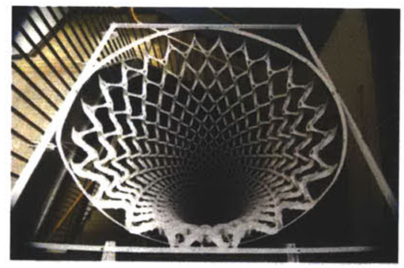
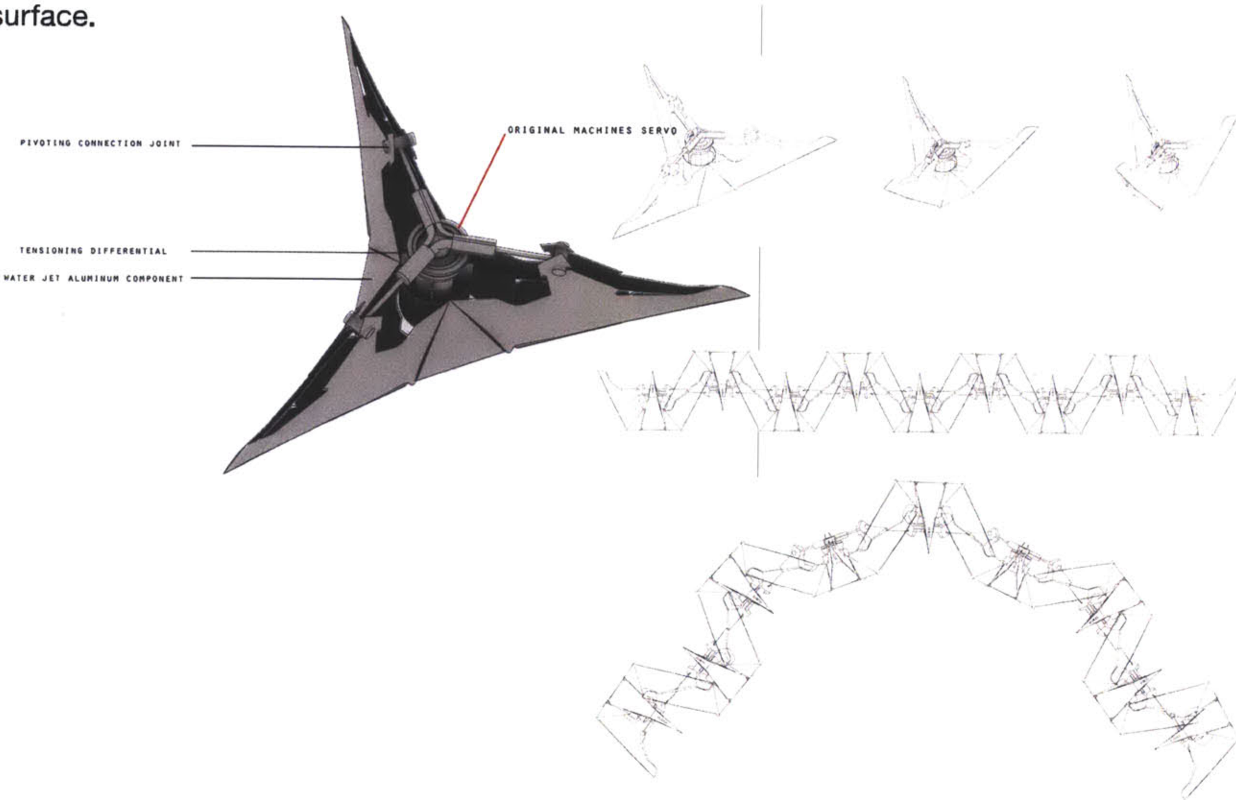


Figure 5.91 (Above) Photograph of (Dis)Course 4 installation at the MIT 150th Anniversary Festival of Art Science and Technology (FAST). (Right) Detail of an actuated component and its various stages of opening and closing.

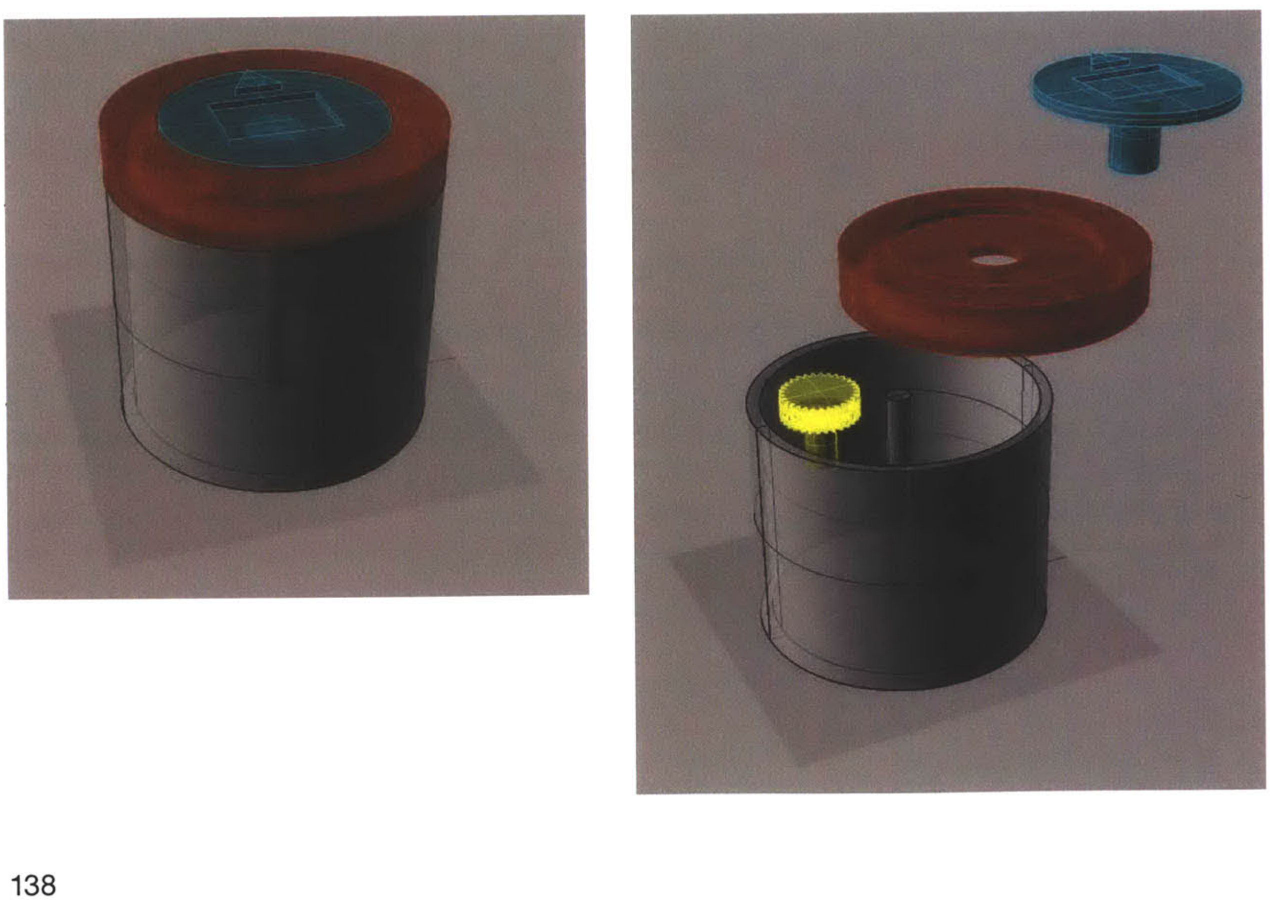


Example 4: Augmented Tangibles by Anthony DeVincenzi

Tony DeVincenzi, a communication designer and graduate student in Prof. Hiroshi Ishii's Tangible Media Group at the MIT Media Lab describes his project as follows: "Augmented Tangibles aim to explore the causal relationship between virtual information and its physical counterpart. We look at explicit coupling between tangible XYZ coordinates and virtual XYZ coordinates, with focus on dynamic shape, adaptive rendering and perspective based on the factors of proximity and orientation. Ultimately, Augmented Tangibles provides a multi-user, multi-perspective environment where both physical and digital objects can be collaboratively explored and manipulated."

In order to achieve the specific user interaction blending physical and virtual space, Tony faced the common problem of ideas and design goals that exceeded the possibilities available to him with 2D fabrication methods. His proposal would have called for cutting many individual layers of material and painstakingly gluing them together to achieve the desired approach. The object-oriented mechatronics tools generate gears and bearings for 3D printing

Figure 5.92 Tangible puck shown assembled and exploded to highlighting the paraGear (yellow), customized paraGear to create an internal ring gear which also contains the paraBearing (red), and display surface (blue)



and enabled him to approach his project holistically focusing on the interaction design rather than the fabrication method. After only one day of design iterations, the final tangible puck (the device is a handheld object) was ready for printing.

Reflecting on the Design Explorations

The design explorations described above are all at various stages of completion and illustrate the qualities of originalMachines as well as the differences and similarities between traditional design processes and a more parametric, object-oriented approach to mechatronics.

Lowering the Barrier to Entry

The tools developed for the object-oriented mechatronic approach help overcome the barriers of engaging with complex mechanical assemblies. No up-front knowledge about mechanisms is needed to start planning, using and implementing servos and bearings in projects. Andy Payne, the author of the Firefly Experiments for Grasshopper, reacted to the design tool in the following informal comments: "I can see a lot of applications for things like this....When I teach a lot of these workshops on Firefly... I always introduce Servos and show what they can do... but a lot of people get stuck because they think that they can only turn an arm back and forth. They often don't have any idea about gearing and bearings, etc. **With the proper setup, you can do a lot of really great things with simple rotation and I think this could be a big step in helping people get a handle of how to start prototyping more interesting mechanisms.** (...)" (emphasis mine) Payne's observation illustrates how the tool could introduce interested users more quickly to the complexities of mechatronic assemblies.

Nevertheless the need for additional skills such as CAD undoubtedly limits the potential user base. Transferring the tool to another platform could be more inviting for other users. Though Rhino and Grasshopper have a broad user base within design and architecture, they are less commonly used by mechanical engineers who specify components in Solidworks. An object-oriented mechatronics tool has the potential to bridge between these two disciplinary preferences for a particular platform.

A more flexible platform would also address the tool's reliance on the computational resources of the particular computer or environment. Delayed geometry updates and frozen outputs are caused by the large amount of computation required to create gears and servos in a dynamic parametric way.

Supporting a Unified Workflow

The typical design process for robots requires using many different software packages to complete all the necessary specifications, designs and code. In the examples shown above, there are no breaks in the chain from design to fabrication. There is no need for specialists to take over the fabrication process at a given moment. Designing and fabricating are at the fingertips of a single user who controls every step of the process. This integration makes it possible to change early design decisions much later than usual along the road from design to fabrication. The unified workflow has the potential to increase a user's ability to identify strongly with a design enhancing the likelihood for expressive machines. It is also easier to pay attention to details without being bogged down by the fine details of component specifications. The dual benefits of heightened control and increased flexibility can then allow users to truly create new kinds of originalMachines.

Generating Diverse Artifacts

Through the ease of use and low barrier of entrance new kinds of objects are created which otherwise would have to be compromised by fabrication for mechanical assembly constraints. The object-oriented mechatronic workflow in combination with 3D printing creates a quicker turn around for mechanical assemblies benefitting more iterations and refinements and ultimately leading to a reduced development time and more diverse outcomes.

Due to the fragmented nature of 3D printing methods, fine tuning and general offset matching for each method's particularities are required to further refine the originalMachine tools. Currently PolyJet and SLA seem to be the only successful methods for which the tool can always create reliable print files. Ideally, SLS should be used in the future because the material is more durable allowing people to create final products rather than prototypes.

From the experiences described in this chapter, the full promise of digital fabrication is yet to be fulfilled. While the goals and capabilities of direct digital manufacturing are clearly defined and described the practical results and pipelines are quite challenging to navigate, especially as a novice. Printed parts do not always match the submitted files and 3D printing service providers recommend ordering multiples made in order to fine-tune the offsets (at additional costs to the user).

Expanding the Tool

Making more expressive originalMachines would be easier for new users if the tool included a parametric definition for the geometries around the actuators. Along the same lines, more sophisticated presets would guarantee a basic result for every user. Specific domains such as architecture or robotics might require subsets of the tool with specialized parametric definitions.

The complexity involved in creating mechatronic artifacts cannot be fully removed with the introduction of object-oriented mechatronics. Still, the examples described above provide a visual and physical model for the diversity of possible outcomes when some barriers are reduced. Personal skills and aesthetic languages should not be lost in creating more generic parametric definitions. Still, many communities stand to benefit from a common platform for engaging with each other on the form and function of originalMachines.

Chapter 6

Conclusion

Contributions

The thesis makes several contributions to the field of digital fabrication, computational design, and new approaches to building kinetic objects. Part of the contributions are the detailed descriptions of and experiences with many different rapid prototyping technologies. Another part relates more to the implications for design and the qualities of future robotic artifacts.

Cook Book

This dissertation presents a "cook book" for object-oriented mechatronics by translating mechanical components, power transmission elements, actuator typologies and integrated assemblies from current state of the art fabrication methods into direct digital fabrication processes meaning stronger, integrated loops between design and fabrication. The cook book documents more and less successful implementations allowing others to retrace the same steps and learn from them to create new translations for their own projects.

The cook book offers many tips and tricks for using digital fabrication tools. Many of the required machines depend on specific knowledge to achieve the necessary precision for mechatronic elements. Using multiple machines for a single object further complicates the fabrication process because many features like offsets may vary. In a public-access workshop, these difficulties are compounded. 3D printing offers a partial solution to many of these challenges. As soon as a part or complete object has been created in a digital design environment it is ready to be handed off to the 3D printer without requiring further interventions from the user.

3D Printing for Parametric Kinetic Objects

This thesis contributes towards the future use of rapid manufacturing techniques and especially 3D printing to replace injection molding and expensive tooling processes which are often removed from users and owned by experts thereby making them more accessible for expert and non-expert end-users. Often the eureka moment of having a great idea is followed by the discouraging phase of relating the idea to what can actually be built with the available resources. This phase often draws the attention away from the initial spark and the core intention behind the design. 3D printing allows users to move directly from their representation in CAD to a printable object. While this transition requires significant skills it takes place within the same environment. Thus 3D printing supports individuals in regaining more freedom over the development of complex mechanical ideas and project.

Parametric Design Tool

As 3D printing relies on files that are generated from CAD programs they can be connected with parametric tools within those software packages. In other words, parametric design tools can easily generate the kinds of diverse content appropriate for 3D printing which supports one-off manufacturing and the creation of unique artifacts. And parametric design approaches support the kinds of complexity which result from making many different versions of the same object.

Building upon the results from the cook book in connection with direct digital fabrication this thesis introduces parametric design tools that allow users individually or collaboratively to create and fabricate mechatronic, near-complete assemblies more easily with a lower barrier to entry and little expert knowledge. To use the tool at this stage, users still need strong CAD skills, however, no formal mechanical engineering skills are necessary. The parametrically encoded knowledge on servomechanisms makes it possible to ease users into the creation of complex mechanical objects. Coupled with direct digital fabrication the tools presents a novel design space.

Design Process and Workflow

The tool's power derives from the design process and workflow it enables. It integrates early ideation with final specifications for a fully functional artifact. This link makes it easy for users to rethink decisions from the earliest phases of the design process which often become foregone conclusions in other workflows. For example, it can become very difficult to change a part late in a development process if it would require significant retooling on a mass-manufacturing line.

A method and workflow for object-oriented mechatronics is established throughout this thesis augmenting and improving upon currently fragmented design approaches. Within one CAD environment objects representing mechanical and mechatronical instances can be called and linked to design driving aspects. Rather than constraining the design as is the case with pre-made components, the object-oriented mechatronic instances stay malleable, resituate and update themselves within the project's specifics. They enable iterating through different scales and versions while maintaining functional parameters and control system compatibility.

The art pieces I created during my fine arts program were challenged by repurposing and reusing pre-made components and ready-mades into the different context of a kinetic artwork. It required significant efforts to make them appear new and aesthetically pleasing. This process has similarities to the engineering approach where off the shelf components are made with the intention to fit a project's specifications. The object-oriented mechatronic workflow enables intention-driven creation from the ground up rather than specification matching of core elements. Idling is different from 001#00 as it is not made from gathered parts and components connected to represent a flower but Idling actually is what it intends to be. All components are seamlessly connected and tightly linked to the overall intention of the piece.

Design Explorations

The design process described allows creators to explore mechatronics for digital fabrication fluidly thereby unleashing the true potential of these techniques on the creation of kinetic objects. In the thesis, many examples of custom and scalable actuator assemblies are described that account for fabrication method, materiality, holistic design integration and user experience. The parametric nature of the design tool makes it more likely that users will generate new types of objects and go beyond existing mechanisms and assembly strategies. Changing parameters, spatial configurations, scale and dependencies without the usual constraints of a computational environment and prototyping implications connects the design explorations more tightly with ideas and content qualities. This flexibility should encourage bolder steps towards completely new mechanical assemblies that are not simply translations from conventional knowledge. Going forward the results of object-oriented mechatronics will not necessarily need to measured by their ability to compete with conventional mechanisms. Even the motor itself could become a parametric definition within the tool.

In the examples, components also loose their definition. Rather than thinking about individual parts that are assembled after the fact the designer can start with an overall appearance that does not have to be compromised or subdivided into discrete elements for manufacturing. Sometimes this freedom can enable the type of homogeneity of Idling with seamless boundaries between four servo-mechanisms. Other times it can serve to highlight boundaries such as the red balls in the bearing for the differential drive robot. In industry, it could prove difficult to remove all component differentiation because many companies are highly specialized and only create a particular part. The blurring of boundaries among parts could overturn a century-old paradigm.

Interdisciplinary Projects

The design examples described throughout this thesis show how roboticists and designers might collaborate through shared parametric definitions specific to various domains. Currently, most mechanical engineers rely on Solidworks, a CAD software feeding directly into industrial manufacturing processes like injection molding. Most designers use Rhino, a more intuitive and open-ended environment. While Solidworks is well-suited for understanding specific performance measures Rhino makes it very easy to sketch and explore ideas in geometry that readily translate into STL files for 3D printing. The object-oriented mechatronics design tools play a connecting role between the two extremes of Rhino and Solidworks. Though currently only linked to Rhino, there may be ways to expand the possibility for interdisciplinary conversations by making it standalone compatible with many CAD environments.

The bundled mechanical knowledge available in a design-oriented CAD environment creates a common ground for interdisciplinary peer groups such as mechanical engineers, roboticists, industrial designers, artists and makers. Every discipline can contribute domain knowledge: mechanical engineers can refine and optimize the parametric definitions while designers can arrange and augment the mechanical parametric definitions with geometry generating definitions and presets. The exchange of knowledge expands the types of robotic artifacts around us.

Almost more valuable than the robotic artifacts are the parametric blueprints created by the tool for them. The “mindmaps” of a particular creation are a map of the design process and can be used for sharing knowledge or teaching. They also reveal what is usually hidden from the observer, namely how an artifact came to be the way it appears.

Road Map for the Future of Object-Oriented Mechatronics

The originalMachine approach holds the potential of redefining and reinventing mechanical assemblies and products to a point where unprecedented and novel machines and mechatronical objects can be created. It spans open a new space situated between high-end industrial mechanics and those product types in which a gear is already too complex and costly. Diverse materials ranging from affordable plastics to expensive high-grade metals will further enrich the types of products made possible with 3D printing.

The design tools also mediate between industrial controlled fabrication and individual, artist- and maker-driven fabrication. Everyone capable of using a computer to engage with and program electronics, robotics and control will also be able to program the physical incarnations of the robots and machines he or she wants to engage with.

Transitioning from On-Demand Parts to User-Created Digital Products

There is a trend towards web-based 3D printing service providers enabling everyone to have access to those resources. Shapeways, Ponoko, i.materialise are just some examples. In order to encourage users to engage with the resources galleries and libraries of user-created content exist. In some cases these galleries can be used as business platforms for users to launch their own design stores. This model is geared towards products made from one part like jewelry for example where a ring is 3D printed and sold through the printing provider's web page.

Most common products, however, require many elements working together. The idea of a "digital product" (Ponoko MIT Media Lab Sponsor Week 2011) has been proposed as a container for multiple parts involved with an object even covering software and instruction manuals. For mechanical artifacts, the digital product definitions could rapidly become overly complex. Object-oriented mechatronics through its parametric nature could automatically create all the necessary information without requiring the user's direct control. Instead, the user can continue to focus on the central design intention.

Going Beyond the Limits of Fabrication Machines

Enabling people to use fabrication machines and make their own ones is inspiring and educational. However, it may not be enough to support the kind of thinking required to create new kinds of artifacts. Rather than exploring the boundaries of machine capabilities our focus should be on exploring the boundaries of what actuated artifacts can be in the future. Teaching people how to build their own tools will get them stuck being a machine builder and operator rather than enabling them to become authors of unique, personal and original content created on fabrication tools

Building Deeper Relationships with Products

Companies like Nike and Ikea but also car manufactures increasingly engage with the concept of mass customization. Consumers are enabled to express themselves by choosing a personal set of product-related parameters like color, detail features, size, etc. Though delightful and often quite popular, these types of presets do not truly allow consumers to act as authors or creators of new products. Instead, mass-customization encourage discrete component parts and industrial-scale manufacturing. Most choices are driven by market research and presented to the consumer as limited array of choices.

If selecting certain small parameters is not the answer, neither is a completely undefined work environment. Users need access to high-end resources in terms of manufacturing quality and the corresponding design tools to become empowered as creators. Increasingly, companies and people can become mature and equal partners in the process of making digital products.

Partners in Making

Object-oriented mechatronics and the types of originalMachines it generates could seed a community of makers interested in bridging the gap between industrial-grade production and DIY (do-it-yourself). Both experts and non-experts should be able to participate in contributing at different levels. In fact, making as much knowledge about different areas such as mechanical engineering, electrical engineering, design or art available to everyone through parametric design tools could create a basis for their interactions. By allowing technical specialists, designers, artists and non-experts to connect on the same platform, a better exchange could be enabled. To achieve such a large vision would require moving to a more generic and powerful platform than the current Rhino/Grasshopper-based tools. Such a community could then create a new demand for more service providers like Ponoko and Shapeways in specialized areas who partner wtih their clients in making new kinds of products.

Mechanical engineers are taught to use off-the-shelf ball bearings. The necessary precision required to assemble one cannot be replicated without industrial grade manufacturing capabilities for steel. This taken-for-granted assumption has removed an entire component from the reach of designers. In contrast, object-oriented mechatronics encourages users to create their own vocabularies of moving parts including even the most sacred parts like ball bearings. Now conventional and standard mechanical components can be thrown up in the air and re-imagined as originalMachines.

List of Images

References

Arduino, open-source electronics prototyping platform. http://arduino.cc/

Antonelli, P. (2008). Design and the elastic mind. New York: Museum of Modern Art.

Asperl, A., Bentley, D., Hofer, M., Kilian, A., and Pottmann, H. (2007). Architectural geometry. Exton, Pa: Bentley Institute Press.

Bergdoll, B., Christensen, P., & Broadhurst, R. (2008). Home delivery: Fabrica.. ng the modern dwelling. New York: Museum of Modern Art.

BiTeam, www.biteam.com

Botha, M. & Sass, L. (2006). The instant house: Design and digital fabrication of housing for developing environments. In Proc. of CAADRIA (Japan), 209-216.

Breazeal, C. L. (2004). Designing sociable robots. Intelligent robots and autonomous agents. Cambridge, MA: MIT Press.

Buechley, L. and Hill, B. M. 2010. LilyPad in the Wild: How Hardware's Long Tail is Supporting New Engineering and Design Communities. In Proceedings of Designing Interactive Systems (DIS).

Buechley, L., Hendrix, S., and Eisenberg, M. (2009). Paper, Paint, and Programs: First Steps Toward the Computational Sketchbook. In Proceedings of Tangible and Embedded Interaction (TEI), Cambridge, UK, February 2009, pp. 9-12.

Cache, B. (2009). "Digital Semper." http://fielddesignlab.files.wordpress.com/2009/07/digital-semper2.pdf

Cole D. Mass MOCA, www.massmoca.org/event_details.php?id=37

Contour Crafting, www.contourcrafting.org

Davies, C. (2005). The prefabricated home. London, UK: Reakon Books.

DFab Lab at ETH Zürich, Switzerland. www.dfab.arch.ethz.ch

Di Stefano, J. J., Stubberud, A. R., and Williams, I. J. (1995). Schaum's outline of theory and problems of feedback and control systems. New York: McGraw-Hill.

Dobson, K. (2007) Machine Therapy. MIT Media Lab, PhD Dissertation. http://hdl.handle.net/1721.1/44329

Dshape, http://d-shape.com

Dynamixel, www.tribotix.com/Products/Robotis/Dynamixel/DynamixelIntro.htm

Essinger, J. (2007). Jacquard's web: How a hand-loom led to the birth of the information age. Oxford: Oxford University Press.

Extreme Textiles. (2005). Cooper-Hewitt National Design Museum, NY.

http://cooperhewitt.org/EXHIBITIONS/extreme_textiles/site/index.htm

Freedom of Creation. www.freedomofcreation.com/

Gershenfeld, N. A. (2005). Fab: The coming revolution on your desktop--from personal computers to personal fabrication. New York: Basic Books.

Helmes, J., Taylor, A. S., Cao, X., Hook, K., Schmitt, P. & Villar, N. (Jan, 2011). Rudiments 1, 2 & 3: Design Speculations on Autonomy. Conference on Tangible Embedded, Embodied Interaction, TEI '11, pp. 145-152.

Hollerbach J., Hunter I. and Ballantyne J. (1992). A comparative analysis of actuator technologies for robotics. O. Khatib, J. Craig and Lozano-Perez Eds, The Robotics Review 2, MIT Press, Cambridge MA 1992, pp. 299-342.

Horowitz, P., and Hill, W. (1989). The Art of Electronics. Cambridge, England: Cambridge University Press.

Hsiu, T., Richards, S., Bhave, A., Perez-Bergquist, A., and Nourbakhsh, I. (2003). Designing a low-cost, expressive educational robot. In Proc. of IROS'03 (Las Vegas, NV).

Hustwit, G. (2009). Objectified. Documentary Film. www.objectifiedfilm.com/

Igoe, T. (2007). Making Things Talk. Make Publishing.
Igoe, T. and O'Sullivan, D. (2004). Physical Computing: Sensing and Controlling the Physical World with Computers. Course Technology PTR.
Ishii, H. (2008). Tangible bits: beyond pixels. In Proceedings of the 2nd international Conference on Tangible and Embedded interaction (Bonn, Germany, February 18 - 20, 2008). TEI '08. ACM, New York, NY, xv-xxv. DOI= http://doi.acm.org/10.1145/1347390.1347392
Ishii, H. and Ullmer, B. (1997). Tangible bits: towards seamless interfaces between people, bits and atoms. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Atlanta, Georgia, United States, March 22 - 27, 1997). S. Pemberton, Ed. CHI '97. ACM, New York, NY, 234-241. DOI= http://doi.acm.org/10.1145/258549.258715
Ingold, T. (2000). The perception of the environment: Essays on livelihood, dwelling and skill. London, New York : Routledge
Iso Truss, www.isotruss.org
Jones, J. L., and Flynn, A. M. (1993). Mobile Robots: Inspiration to Implementation. Wellesley, MA: A.K. Peters.
Kieran, S., & Timberlake, J. (2004). Refabricating architecture: How manufacturing methodologies are poised to transform building construction. New York: McGraw-Hill.
Khoshnevis, B. (2004) Automated construction by contour crafting – related robotics and information technologies. Automation in Construction – Special Issue: The best of ISARC 2002, 13(1), 5-19.
Khoshnevis, B. & Bukkapatnam, S., Kwon, H., Saito, J. (2001). Experimental investigation of contour crafting using ceramics materials. Rapid Prototyping Journal, 7(1), 32.
Kolarevic, B. & Klinger, K. (2008) Towards a digital materiality. In B. Kolarevic & Klinger (eds.), Manufacturing Material Effects: Rethinking Design and Making in Architecture (pp. 103-118). London, UK: Routledge.
Kilian, A. (2006). Design exploration through bidirectional modeling of constraints. PhD Thesis. Massachusetts Institute of Technology. http://hdl.handle.net/1721.1/33803
KUKA CAMRob,
www.kuka-robotics.com/NR/rdonlyres/26E622E0-8D38-4376-8A6D-6FE13072B0B0/0/KUKA_CAMROB_en.pdf
Lieberman, J. I. (2004). Teaching a Robot Manipulation Skills

through Demonstration. MS Thesis. Massachusetts Institute of Technology.

McCullough, M. (1998). Abstracting craft: The practiced digital hand. Cambridge, MA: MIT Press.

McQuaid, M., and Beesley, P. (2005). Extreme textiles: Designing for high performance. New York: Smithsonian Cooper-Hewitt, National Design Museum.

Mitchell, W. J. (1992). The reconfigured eye: Visual truth in the post-photographic era. Cambridge, MA: MIT Press.

OpenServo Community-based Project. www.openservo.com

Perner-Wilson, Hannah. (2011). Kit-of-No-Parts. MIT Media Lab. http://web.media.mit.edu/~plusea/?p=265

Poupyrev, I., Nashida, T., and Okabe, M. (2007). Actuation and tangible user interfaces: the Vaucanson duck, robots, and shape displays. In Proceedings of the 1st international Conference on Tangible and Embedded interaction (Baton Rouge, Louisiana, February 15 - 17, 2007). TEI '07. ACM, New York, NY, 205-212. DOI= http://doi.acm.org/10.1145/1226969.1227012

Qi, J. and Buechley, L. (2010). Electronic popables: Exploring paper-based computing through an interactive pop-up book. In Proceedings of the Fourth international Conference on Tangible, Embedded, and Embodied interaction (Cambridge, Massachusetts, USA, January 24 – 27, 2010). TEI '10. ACM, New York, NY, 121-128.

Raffle, H.S. 2008. Sculpting behavior : a tangible language for hands-on play and learning. PhD Dissertation. Media Arts and Sciences. Massachusetts Institute of Technology http://hdl.handle.net/1721.1/44912

Raffle, H. S., Parkes, A. J., and Ishii, H. (2004). Topobo: a constructive assembly system with kinetic memory. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Vienna, Austria, April 24 - 29, 2004). CHI '04. ACM, New York, NY, 647-654. DOI= http://doi.acm.org/10.1145/985692.985774

Reas, C., Fry, B., and Maeda, J. (2007). Processing: A Programming Handbook for Visual Designers and Artists. Cambridge, MA: MIT Press.

Reconfigurable Robotics Workshop, San Diego, CA, USA.

Resnick, M. and Silverman, B. (2005). Some reflections on designing construction kits of kids. In Proc. of IDC'05

(Boulder, CO).
Ritter, A. (2007). Smart materials in architecture, interior architecture and design. Basel: Birkhäuser.
R-O-B, Mobile Fabrication Unit, DFab ETH. www.gramaziokohler.com/
Rusk, N., and Resnick, M., Berg, R., Pezalla-Granlund, M. (2008). New Pathways into Robotics: Strategies for Broadening Participation. Journal of Science Education and Technology, 17(1), 59-69.
Sachs, E.M. & Cima M. J., Williams, P, Brancazio, D., Cornie, J. (1992). Three dimensional printing: Rapid tooling and prototypes directly from a CAD model. Journal of Engineering Industry, 114(4), 481-488.
Sass, L. & Botha, M. (2006). Instant house: Onsite manufacturing of housing with digital fabrication. International Journal of Architectural Computing, 4(4), 109-123.
Sass, L. yourHouse, MIT Design and Fabrication, http://mit.edu/yourhouse/index1a.html
Schmitt, P. and Seitinger, S. 2009. Plywood punk: a holistic approach to designing animated artifacts. In Proceedings of the 3rd international Conference on Tangible and Embedded interaction (Cambridge, United Kingdom, February 16 - 18, 2009). TEI '09. ACM, New York, NY, 123-126. DOI= http://doi.acm.org/10.1145/1517664.1517695
Schodek, D. L. (2004). Digital design and manufacturing: CAD/CAM technologies in architecture. Hoboken, N.J.: John Wiley and Sons.
Semper, G., Mallgrave, H. F., and Robinson, M. (2004). Style in the technical and tectonic arts, or, Practical aesthetics. Texts and documents. Los Angeles: Getty Research Institute.
Semper, G. (1989). The four elements of architecture and other writings. RES monographs in anthropology and aesthetics. Cambridge [England]: Cambridge University Press.
Shape 3, http://shape3.com/Frameset_Shape3.htm
Siegwart, R., and Nourbakhsh, I. R. (2004). Introduction to autonomous mobile robots. Intelligent robots and autonomous agents. Cambridge, MA: MIT Press.
Sutherland, L. (2002). Masters of structure: Engineering today's innovative buildings. London: Laurence King.

Helmes, J., Taylor, A. S., Cao, X., Hook, K., Schmitt, P. & Villar, N. (Jan, 2011). Rudiments 1, 2 & 3: Design Speculations on Autonomy. Conference on Tangible Embedded, Embodied Interaction, TEI '11, pp. 145-152.

Taylor, A. S. (April, 2009). Machine intelligence. Conference on Human Factors and Computing systems, CHI '09, pp. 2109-2118.

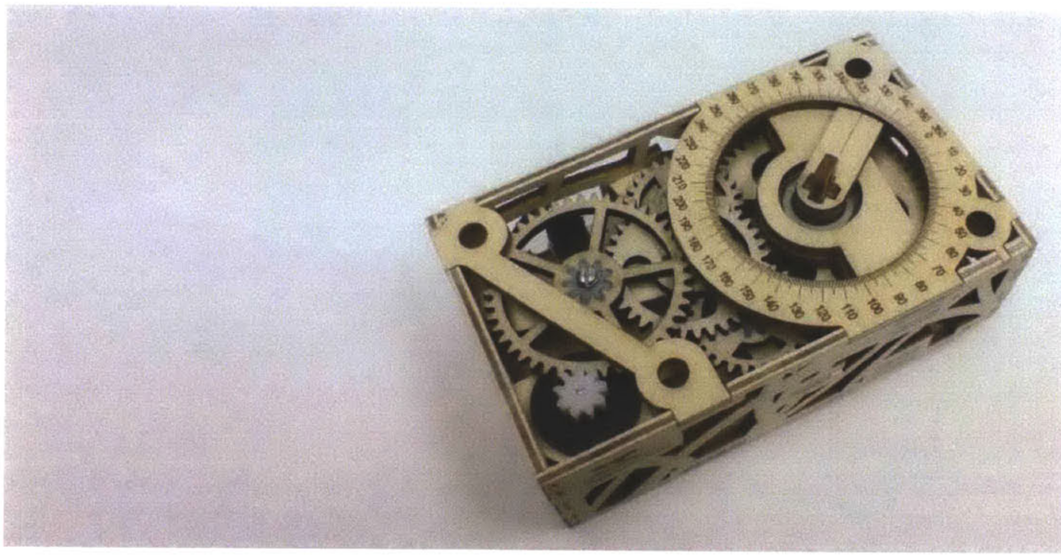
Underkoffler, J. and Ishii, H. (1999). Urp: a luminous-tangible workbench for urban planning and design. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: the CHI Is the Limit (Pittsburgh, Pennsylvania, United States, May 15-20, 1999). CHI '99. ACM, New York, NY, 386-393. DOI= http://doi.acm.org/10.1145/302979.303114

Wescott, T. PID without a PhD. Embedded Systems Programming. www.embedded.com/2000/0010/0010feat3.htm

Zykov, V., A. Chan, et al. (2007). Molecubes: An Open-Source Modular Robotics Kit. IROS 2007: Self-

ZCorporation, www.zcorp.com/en/home.aspx

Appendix

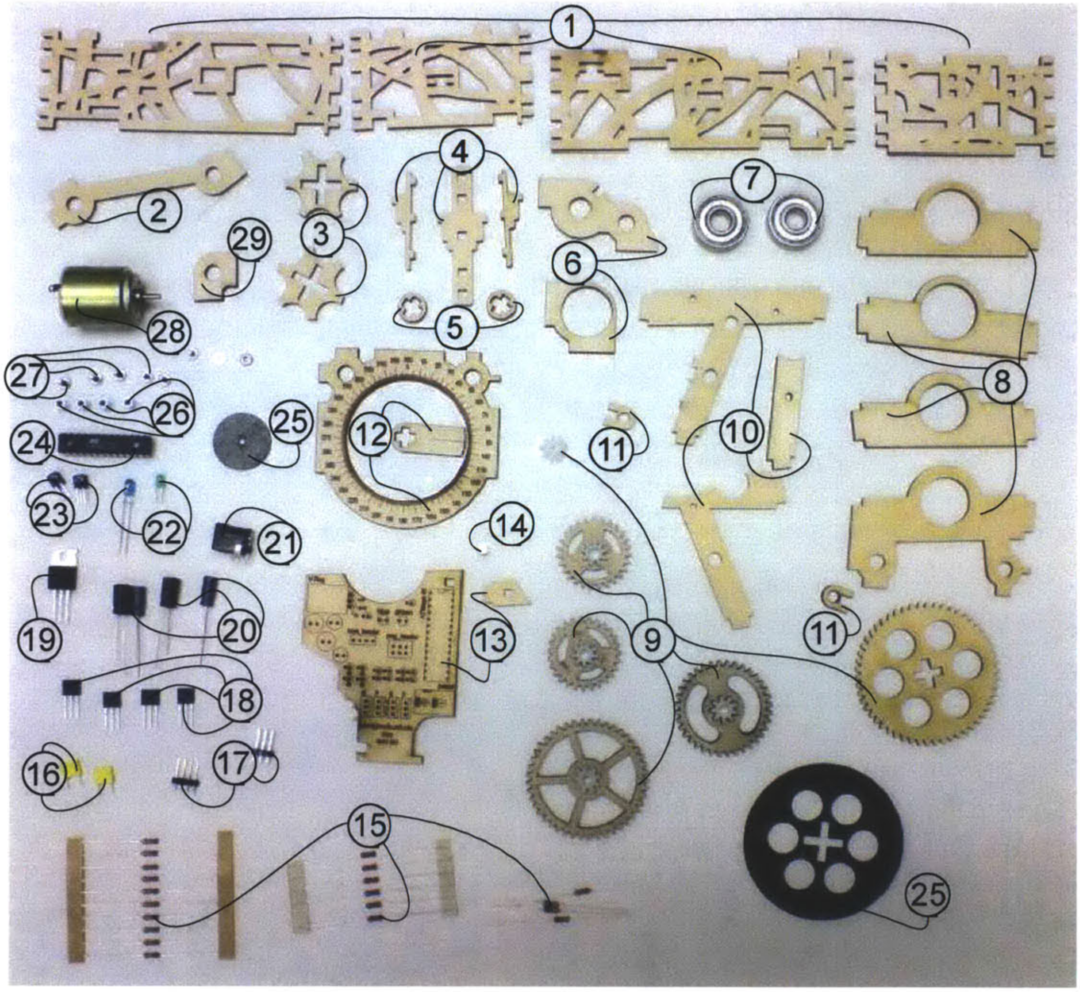


plywood servo kit: user guide

by peter schmitt

contents

1. Component Description

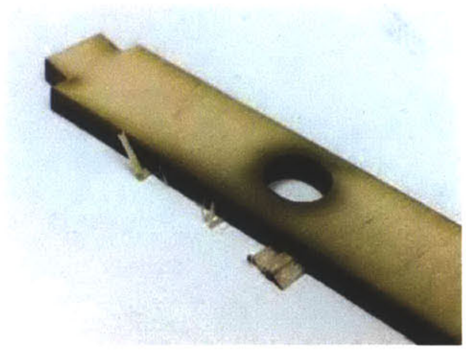


1. side pieces
2. top screw piece
3. encoder disc support ribs
4. main axle body pieces
5. main axle spacer
6. motor ribs
7. ball bearings
8. ball bearing support ribs
9. gears
10. gear support ribs
11. electronic flapp hinges
12. dial and index piece
13. electronic flap and flap stop
14. magnets
15. resistors
16. motor filter capacitors
17. prog. and com. headers
18. N and P Channel MosFets
19. Voltage Regulator
20. Capacitors
21. power plug
22. LEDs
23. optical interruptor
24. ATMega 48 microcontroller
25. encoder disc
26. gear axles
27. screws
28. motor
29. bottom screw hole piece

2. Preparation

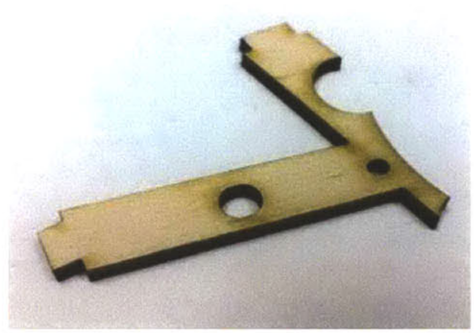
Deburring:

Due to inconsistencies thru out the plywood some areas might not get cut as well as others. In this cases deburring is needed using an sharp knife.

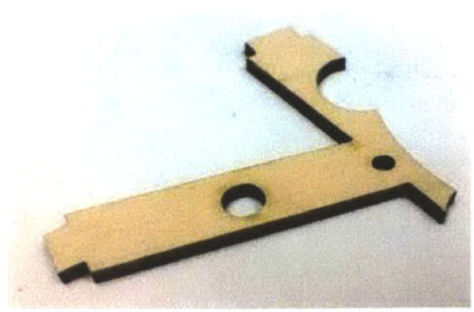


Sanding:

The laser leaves Burn marks on the plywood. Sanding the parts before assembly will help to achieve a nicer and cleaner result.

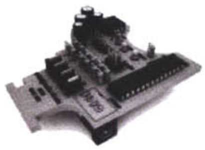


before



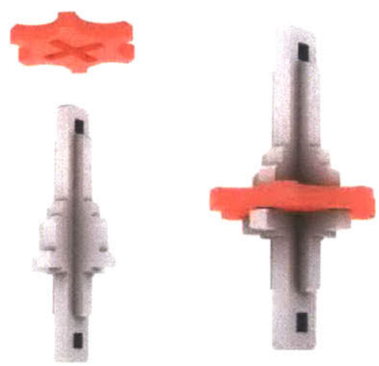
after

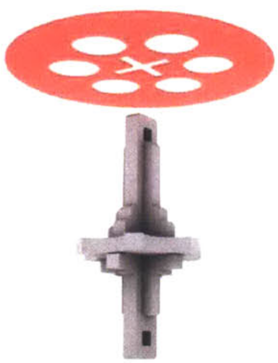
3. Mechanical Assembly



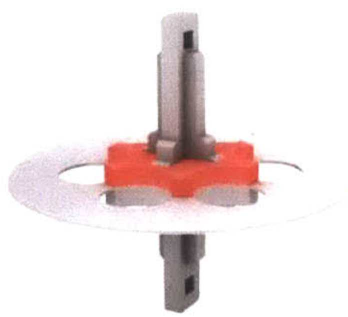
Start the assembly by building up the main axle body pieces. Using wood glue in addition to the press fit connection assures a safe operation of the finished servo.

Slide the encoder disc support rib over the main axle body piece all the way down till it hits the hard stop.



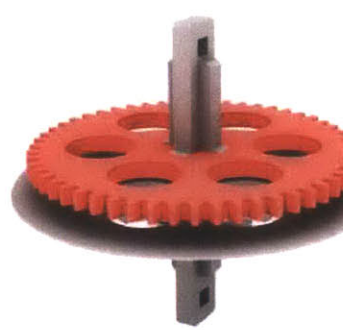


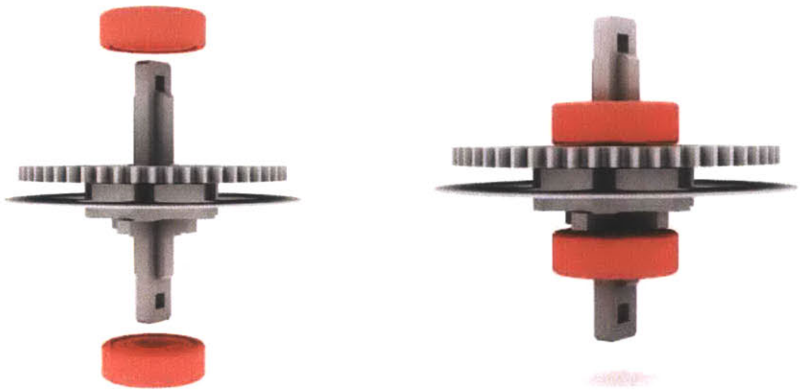
Now slide the encoder disc over the axle.



Add the second support rib on top. The round cut outs should aligne.

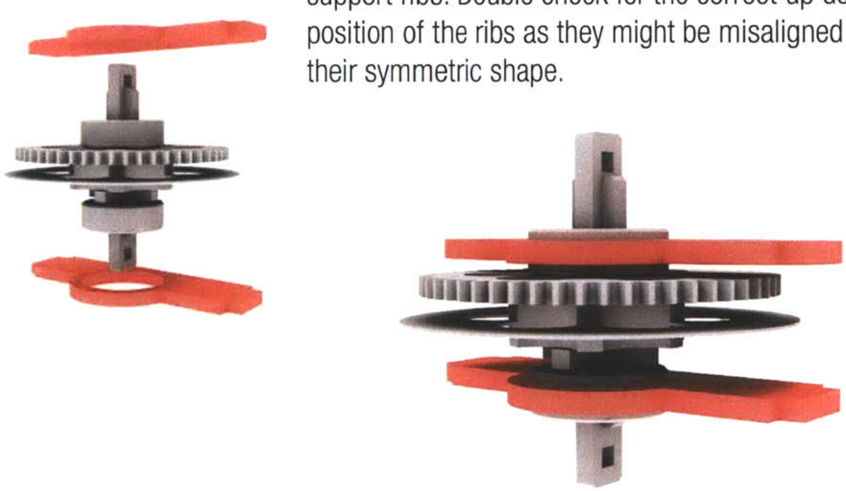
In the next step, slide the main gear into its final position. Make sure it sits tight be-cause this connection has to withstand the biggest torques.





The ball bearings also slide over the main axle into their final positions where the hit a hard stop.

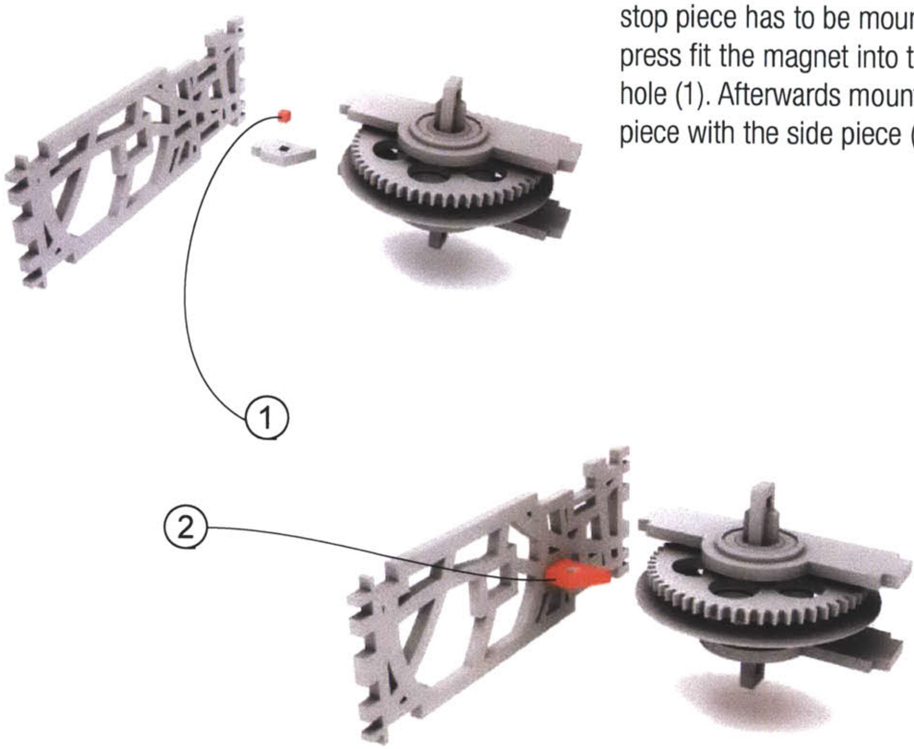
Finalise the main axle assembly by adding the ball bearing support ribs. Double check for the correct up down facing position of the ribs as they might be misaligned due to their symmetric shape.



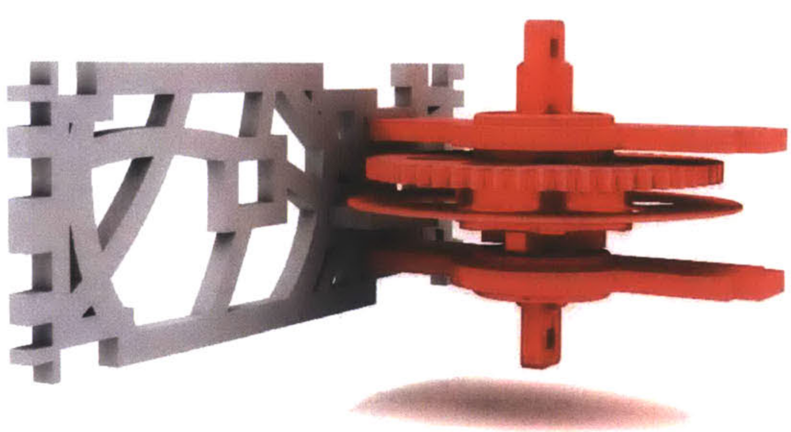
correct position of ball bearing support ribs.

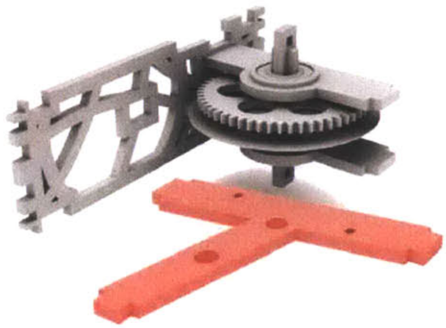
wrong position of ball bearing support ribs!

Before combining the main axle with the side pieces the electronic flap stop piece has to be mounted. First press fit the magnet into the squared hole (1). Afterwards mount the stop piece with the side piece (2).

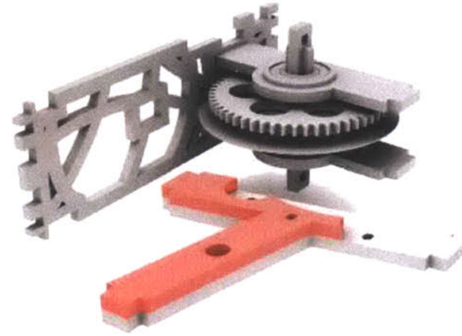


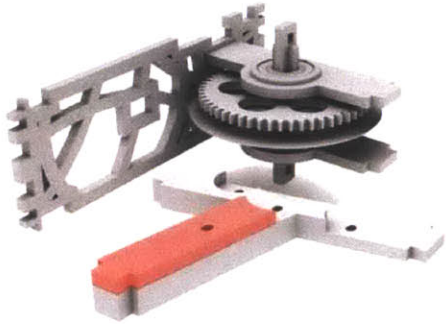
Now the main axle assembly can be combined with te side piece.

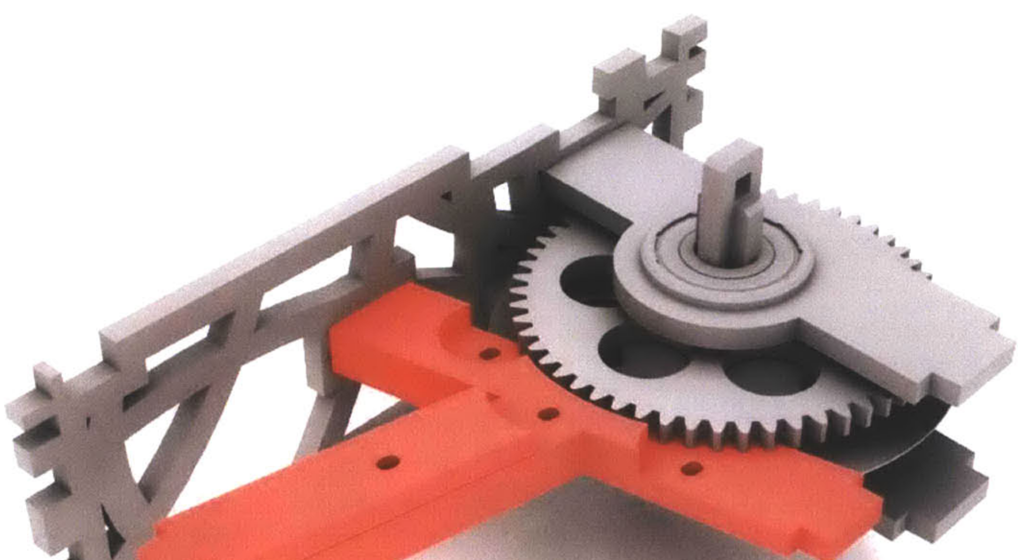




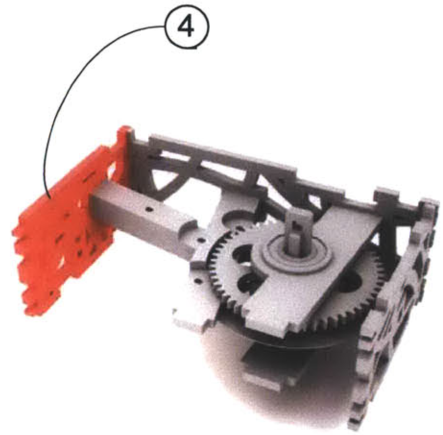
The gear support ribs will have to be build up by gluing the three separate pieces on top of each other before adding them to the side piece.



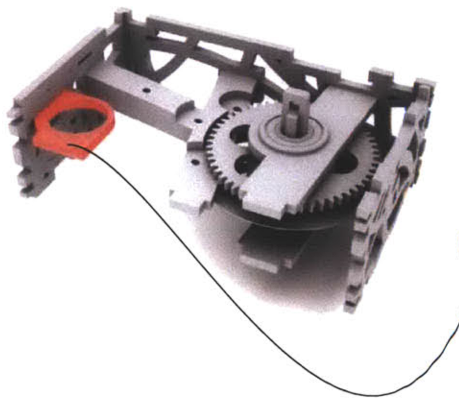


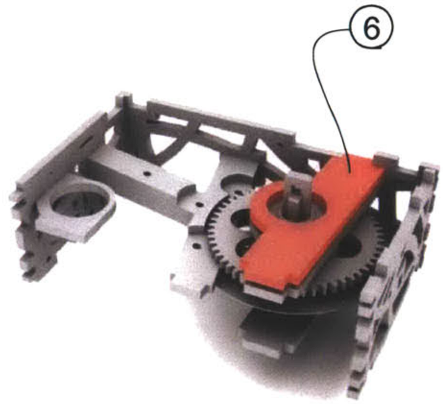


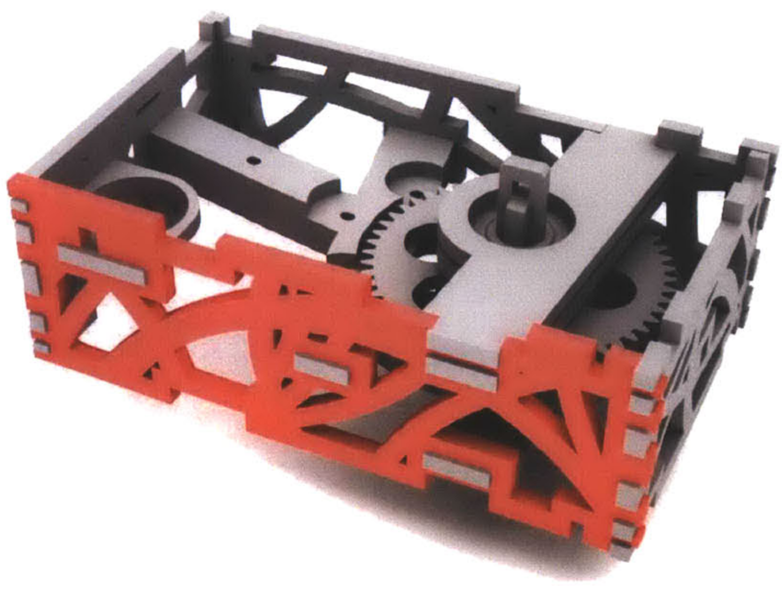
Add side piece 2 (3) and 3 (4).

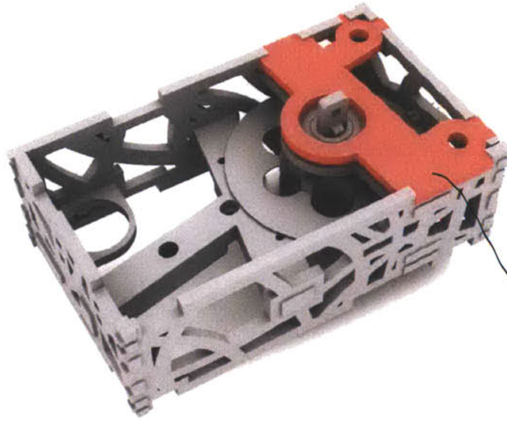


Before mounting side piece 4 put in the motor rib (5) and the top ball bearing support rib (6).



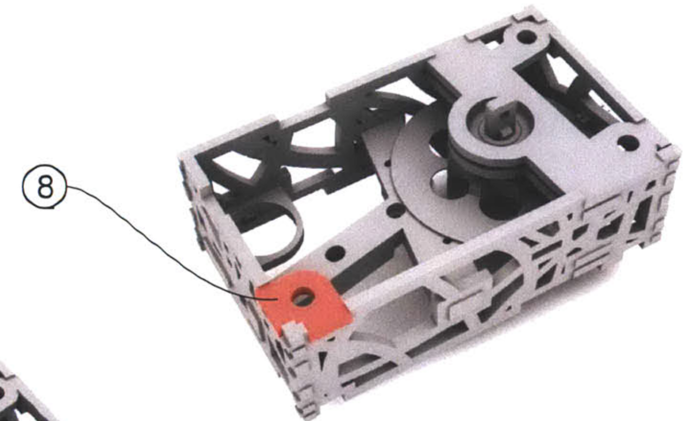


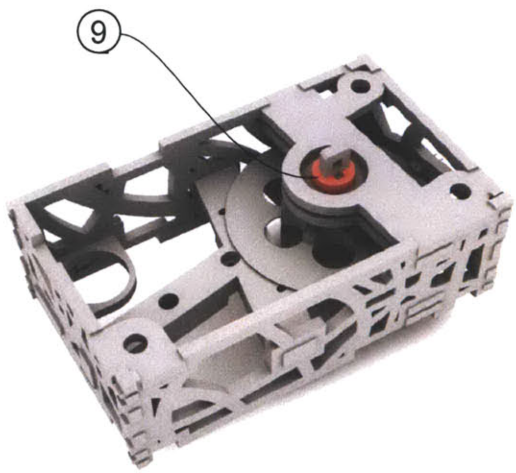




Bottom view:

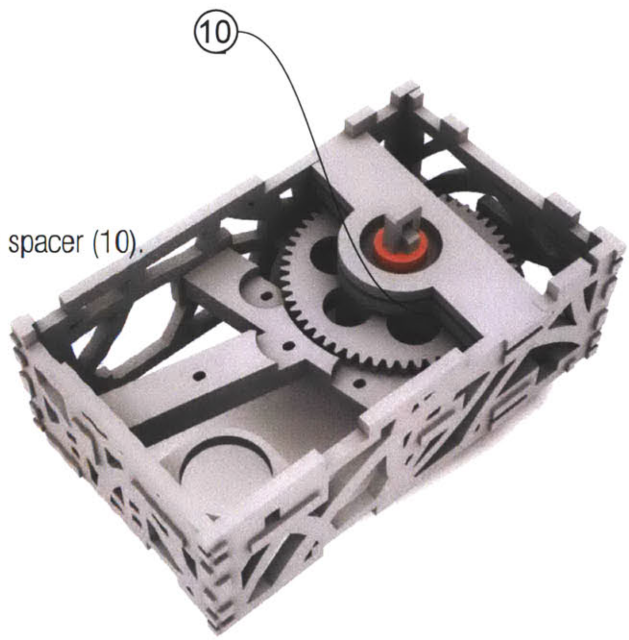
Add the lower ball bearing support rib (7), the lower screw hole piece (8) and the lower main axle spacer (9).





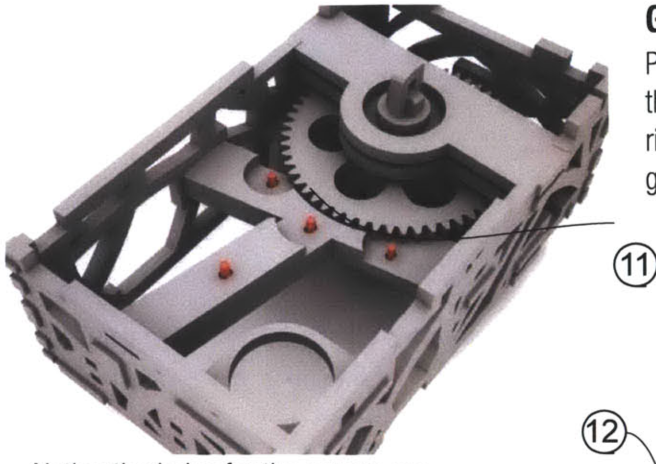
Top view:

Also add the top main axle spacer (10).

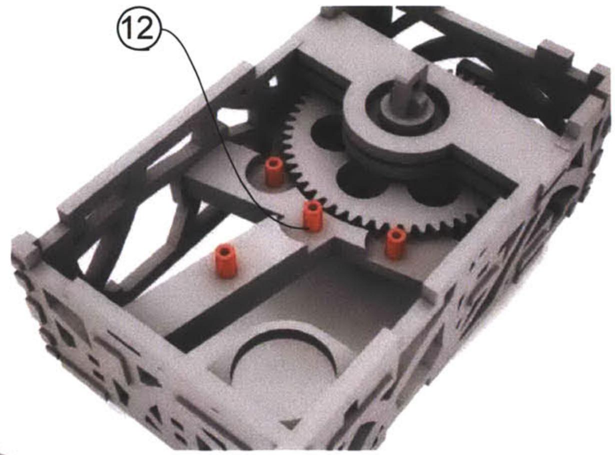


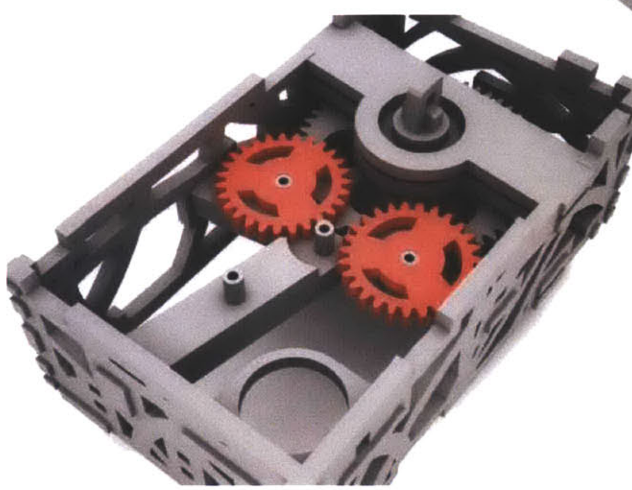
Gear assembly:

Push the screws for the gear axles thru the holes in the gear support ribs (11) and connect them to the gear axles (12).

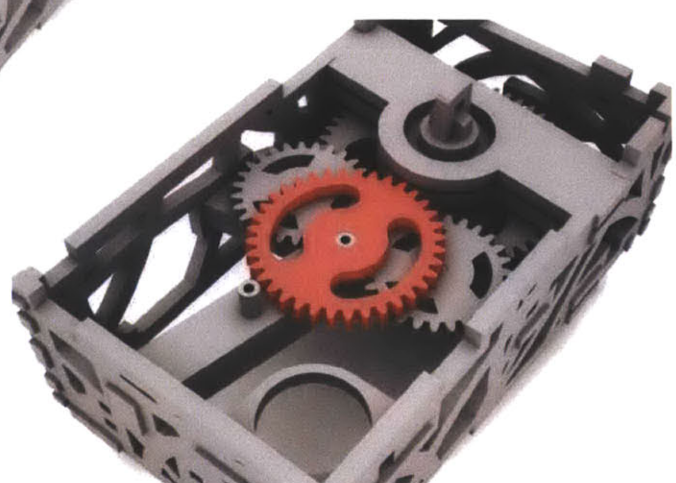


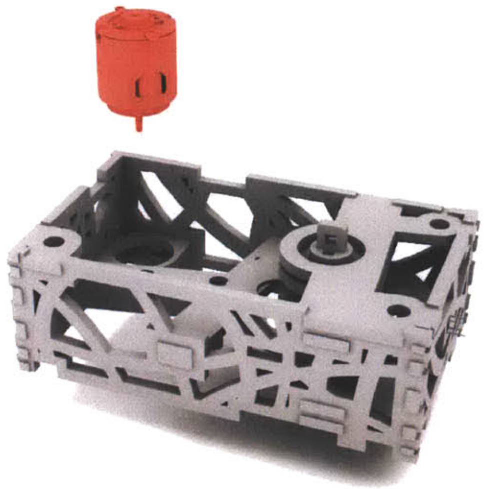
Notice the holes for the screws are slots in order to allow for adjusting the position of the gear axle which will enhance smooth gear operation.





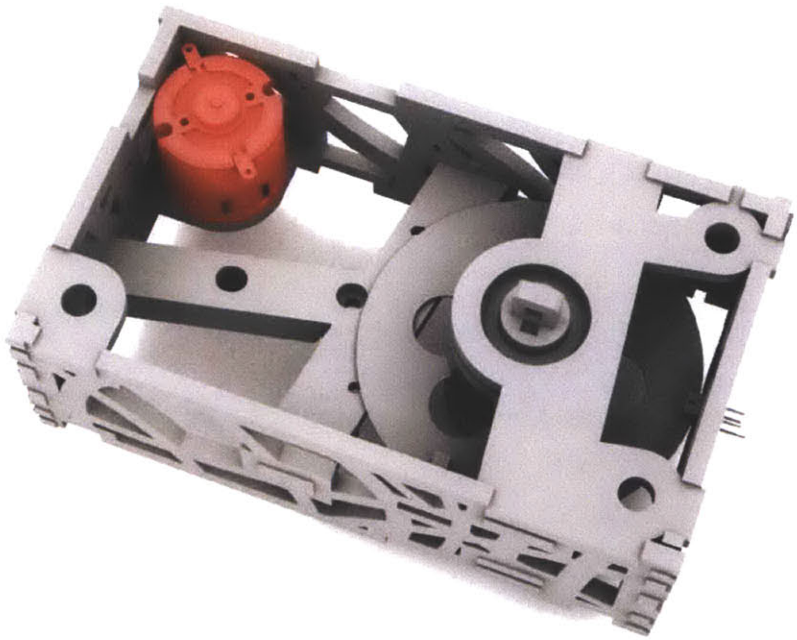
The gears should spin smoothly with almost no friction. Double check full gear revolution for tight spots and correct gear axle position if necessary.

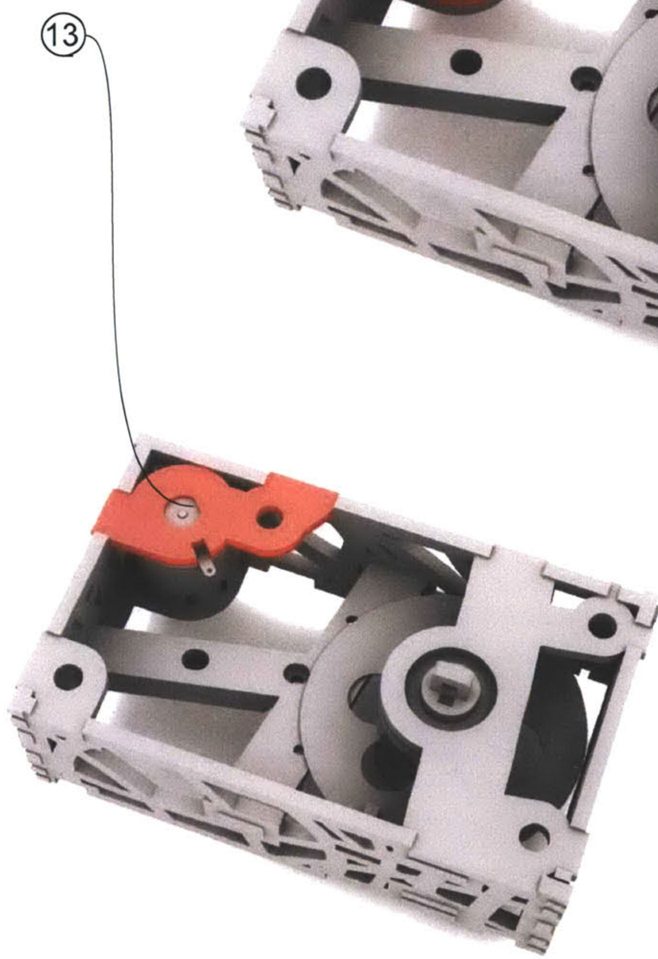




Bottom view:

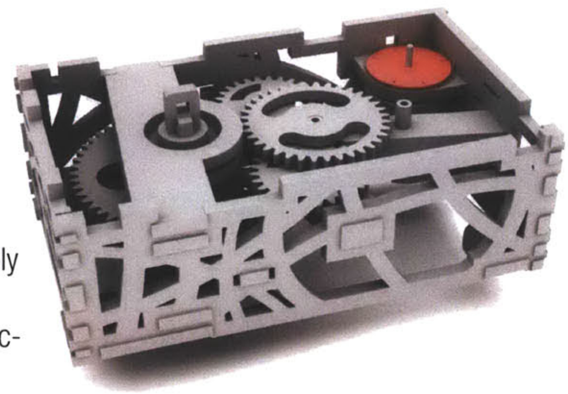
Before adding the final gear on the top side the motor has to be mounted. From the bottom side slide the motor into the motor rib. Fix the motor using the second motor rib. Notice this rib has a keyed connection with the motor which allows for only one motor position. (13)

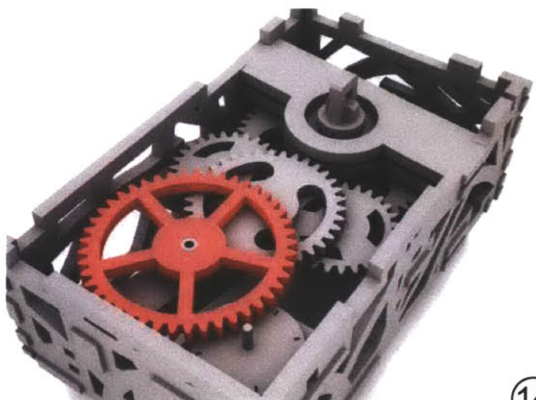




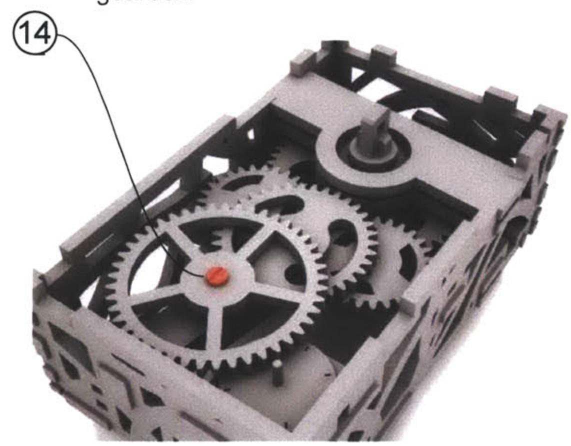
Top view:

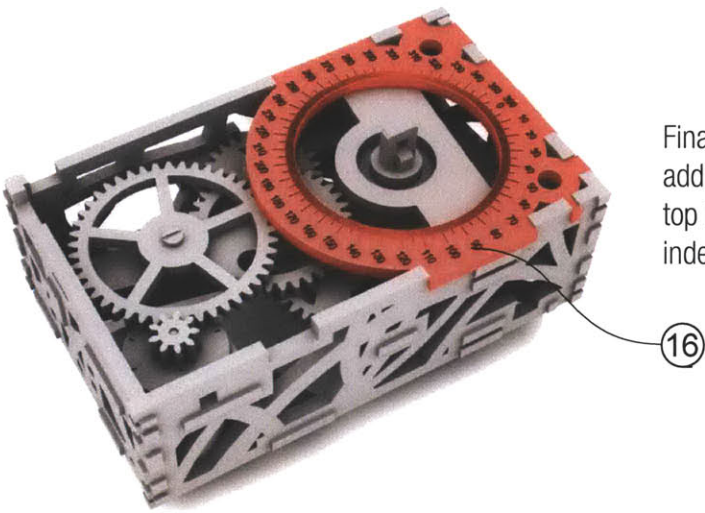
Also before continuing gear assembly the motor encoder disc need to be added now. If possible fix the connection using super glue.



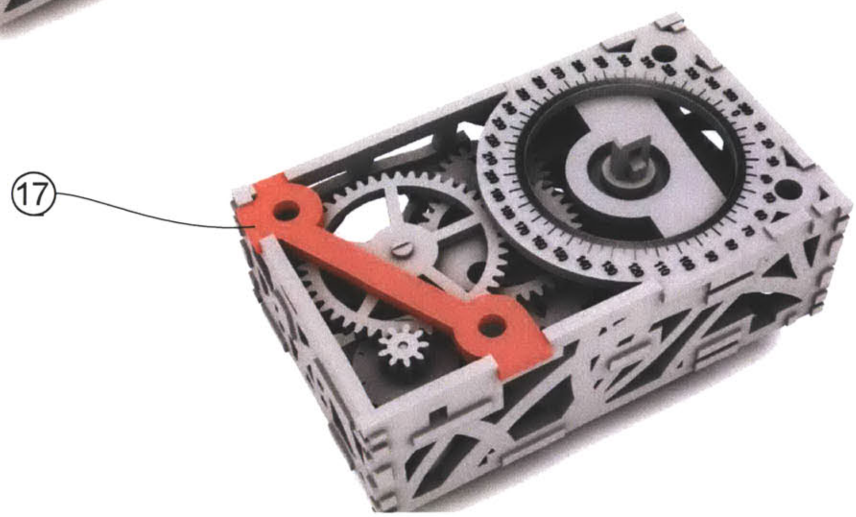


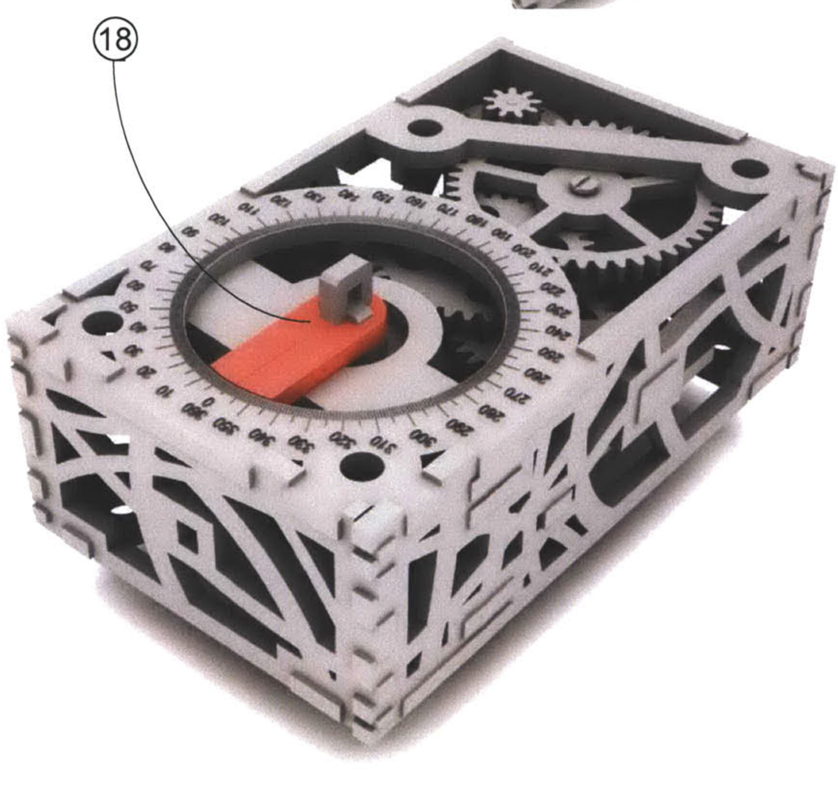
Now the final gear can be put into place. Use either a small screw or a longer one with a hex nut to limit the play\backlash of the gear (14). Complete the gear assembly by mounting the motor gear (15). Double check smooth operation of gearbox.

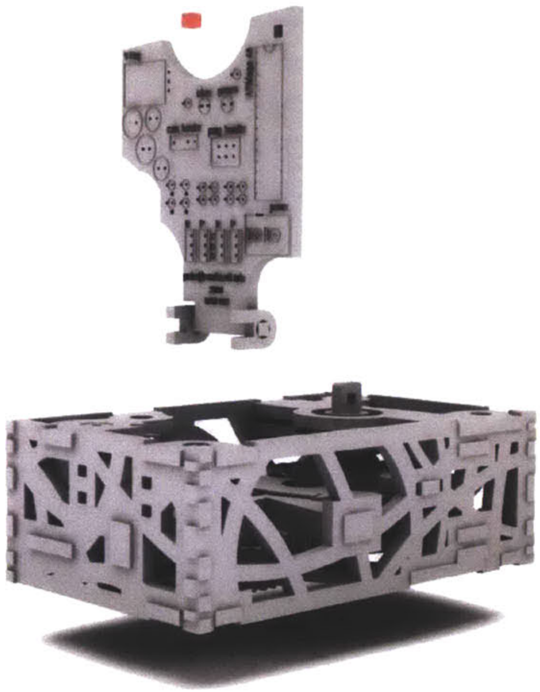




Finally complete the top side by adding the dial piece (16), the top hole piece (17) and the dial index (18).

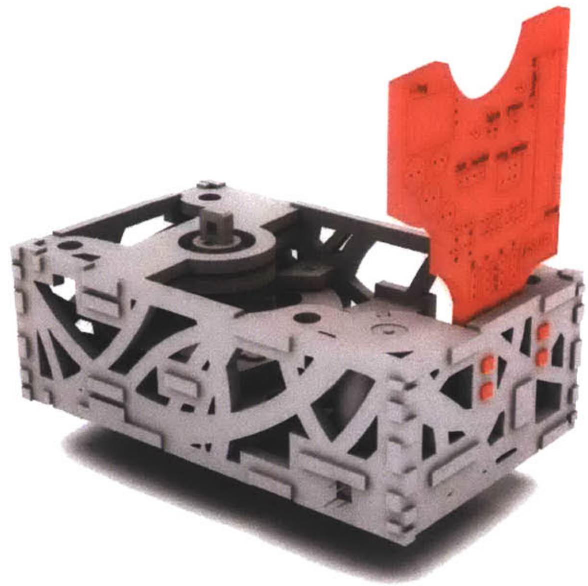


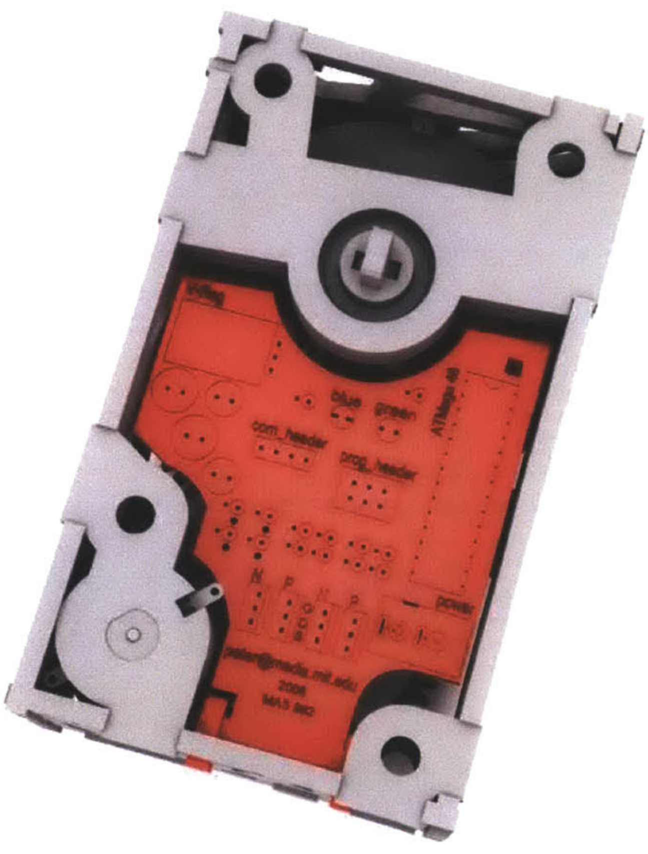




Bottom View:

Finalize the bottom side by installing the electronic flap. Press fit the second magnet into the squared hole. Notice to double check the orientation of the magnet so flap and flap stop magnet will attract each other rather than pushing each other apart.





The press fit connection of the flap hinges might require more tools that just a hammer in order to reach the points to which the pressure has to be applied. An Allen wrench, a screw driver or pliers might be of help.

4. Electronics

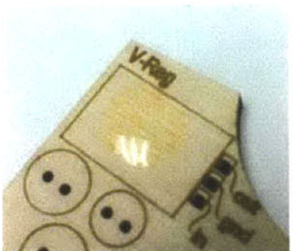
Tips:

Smooth Gear Operation:

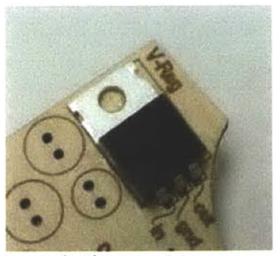
In Order to guarantee stable operation the motor should consume not more than 100 mA while running @ 9-12 V DC. Double check and adjust the gearbox for smooth spinning gears. If necessary increase play between gears to compensate for improper spinning gears.

Fixing Parts using Glue:

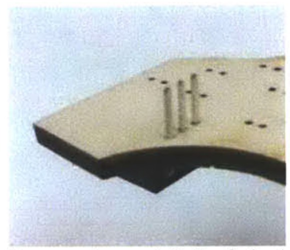
It will be of great help to fix the through hole parts using a drop of super glue while trying to solder them from the backside of the flap.



apply a drop of super glue

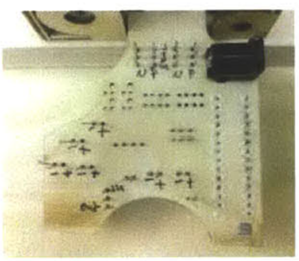


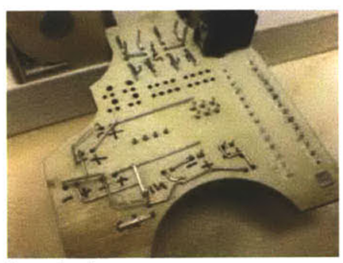
attach the part to the plywood

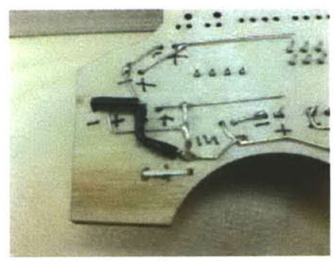

the part wont fall of during soldering

Bending Leads and using Shrink Wrap Tubing:

Instead of cutting of leads on the backside of the plywood flap they can be used to create initial wiring while insolated with shrink wrap tubing where necessary.


leads sticking out on the backside

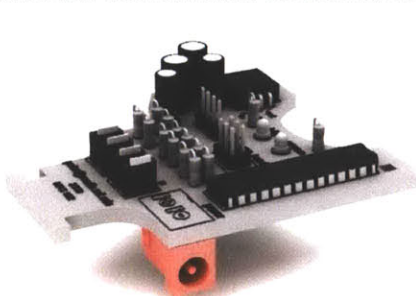

bended leads

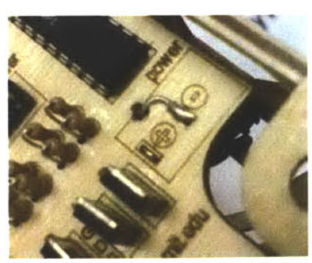

insolated leads using shrink wrap tubing

Special cases:

Power Plug:

The power plug has to be mounted from the lower side in order to provide access for the power jack. Cut off the terminal on the side of the plug and keep both middle terminals. Push the terminals through the plywood board and use the slot for the now cut off terminal as wire duct for the GND connection.

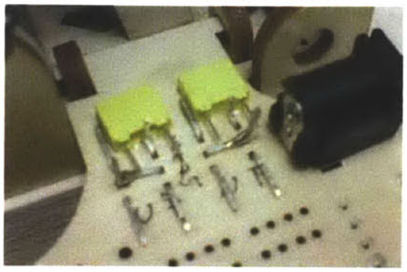

the power plug mounted from underneath



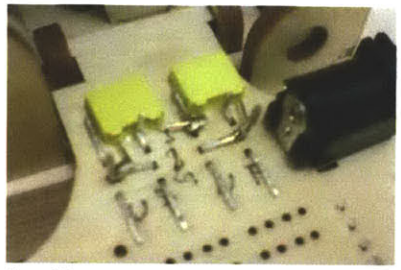
the GND connection

Motor Filter Capacitors:

The motor needs to be shielded to prevent noise in the servo circuits. Two filter capacitors will be added on the bottom side of the plywood board. They are non polar and one side of the capacitors will be connected to the motor terminal connection while the other side of the capacitor will be connected to GND.



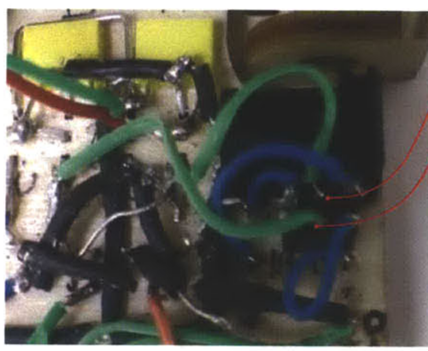
the filter capacitors in their position



the filter capacitors connected to motor and GND

Additional N-Channel MosFets:

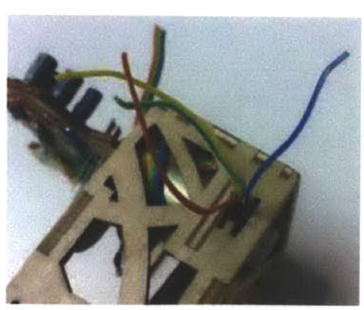
Due to a minor mistake in the circuit additional N-Channel MosFet's are needed. The additional MosFet's will drive the P-Channel MosFet's of the H-Bridge. The image shows a how the MosFet's are super glued to the side of the power plug. Both Source pins are connected to GND. The Drain pins are connected to the Gates of the P-Channel MosFet's belonging to the H-Bridge. The Gates of the additional MosFet's are connected to Pin B0 and Pin B1 of the ATMega 48.



Mounting the Optical Interrupter:

The Optical Interrupter has to be pushed through the side of the casing in which it will be embedded. The slots to both sides allow for the cable to connect to the leads from the inside.

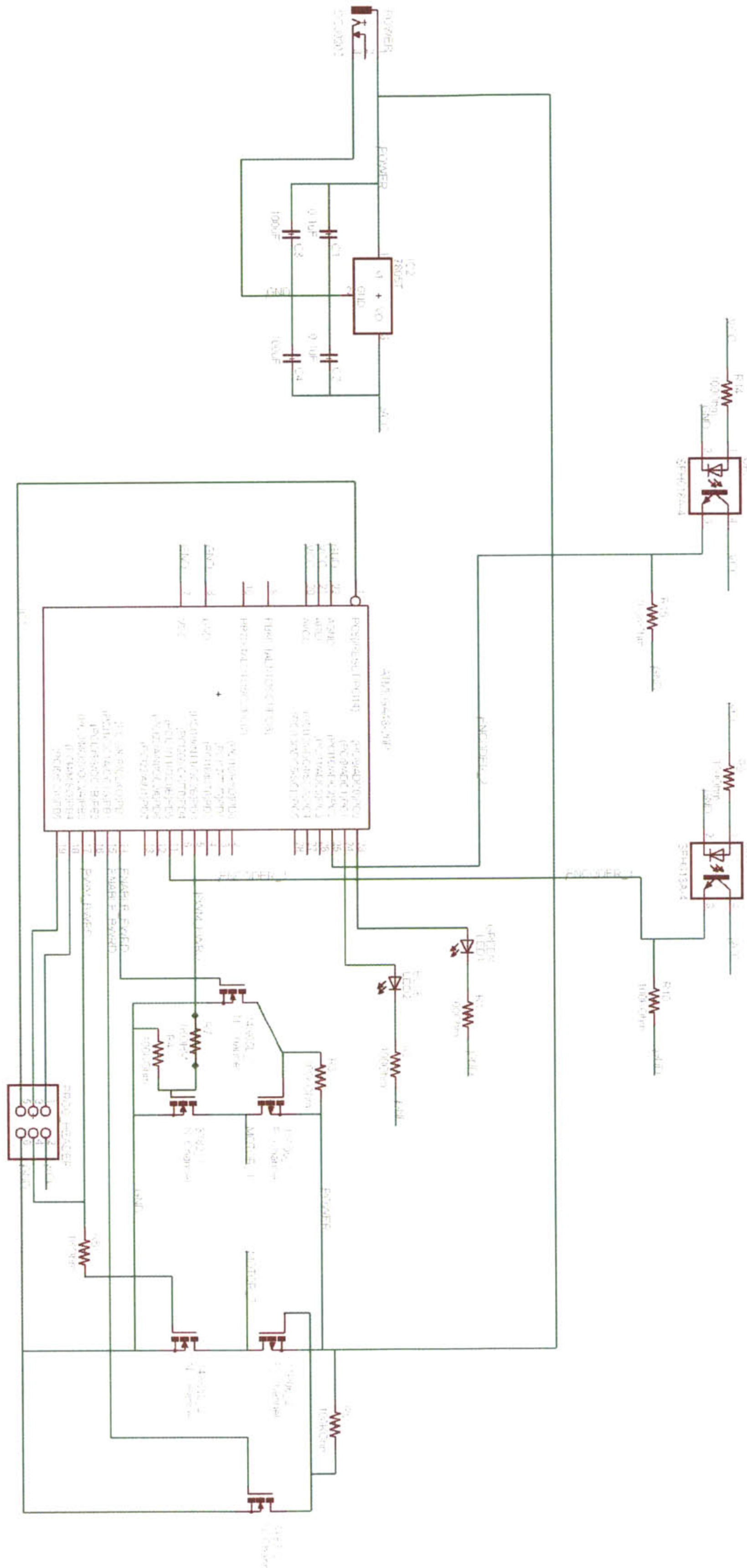
the optical interrupter in it's correct position



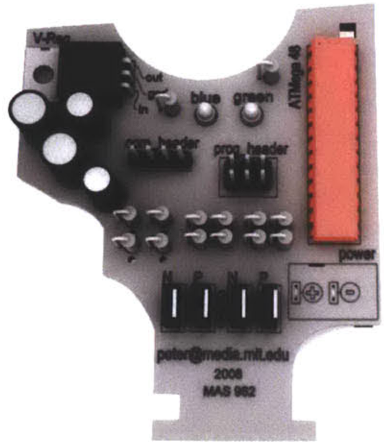
the wires reaching out from the inside

the wires soldered to the leads

Schematic

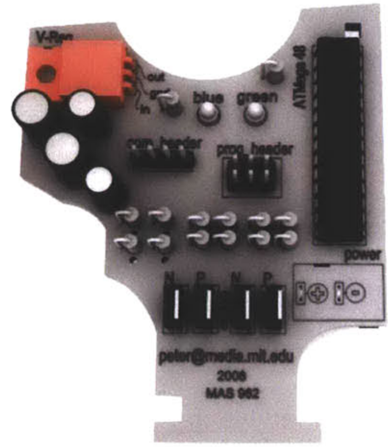


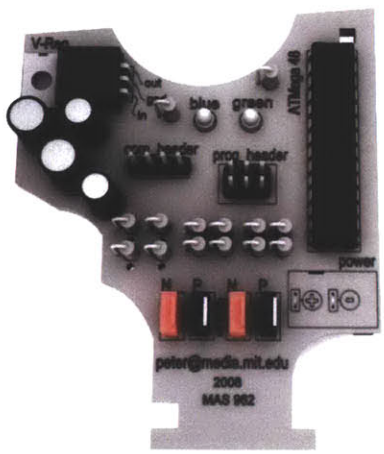
Part Placement



ATMega 48

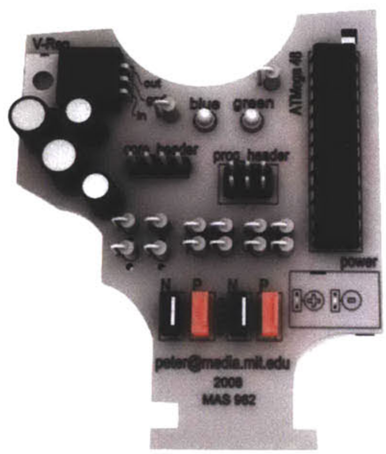
Voltage Regulator, 7805T

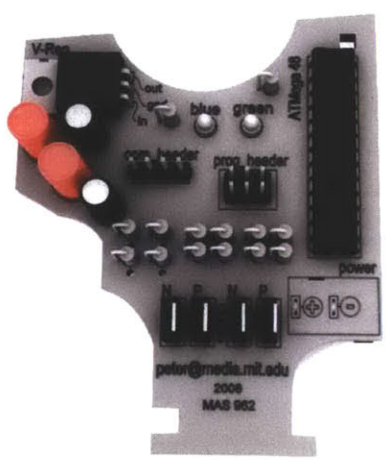




N-Channel MosFet's, 8782

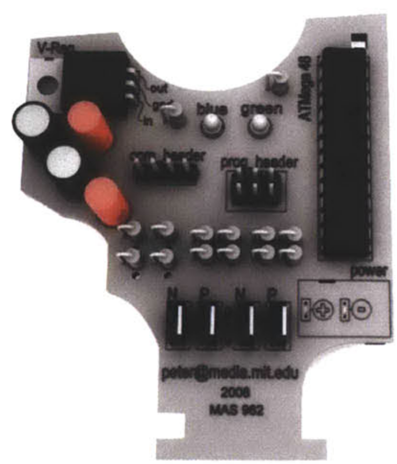
P-Channel MosFte's, 11P06

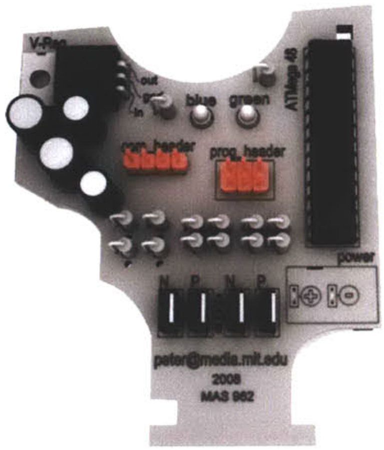




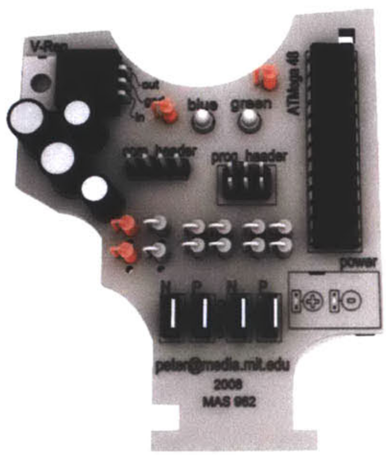
Capacitors 100uF

Capacitors 0.1uF

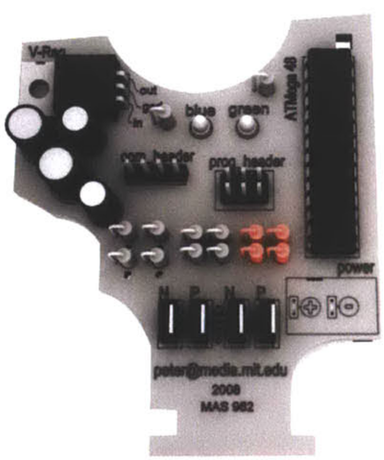




Pin Headers for Programmer and Communication

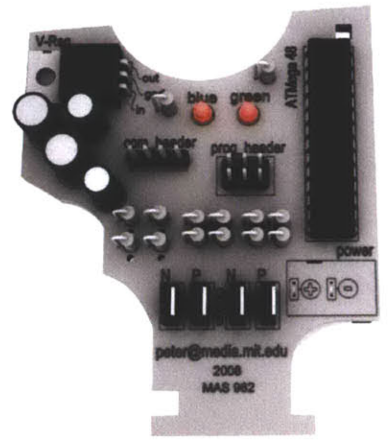


100 Ohm Resistors

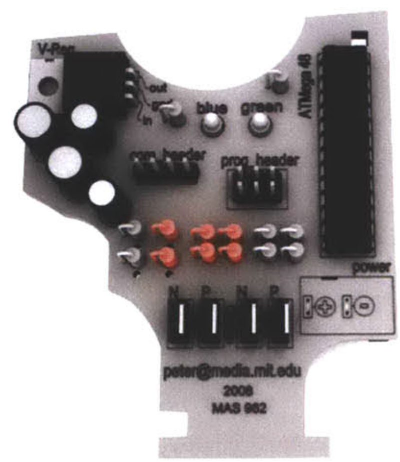


1 KOhm Resistors

Green and Blue LED

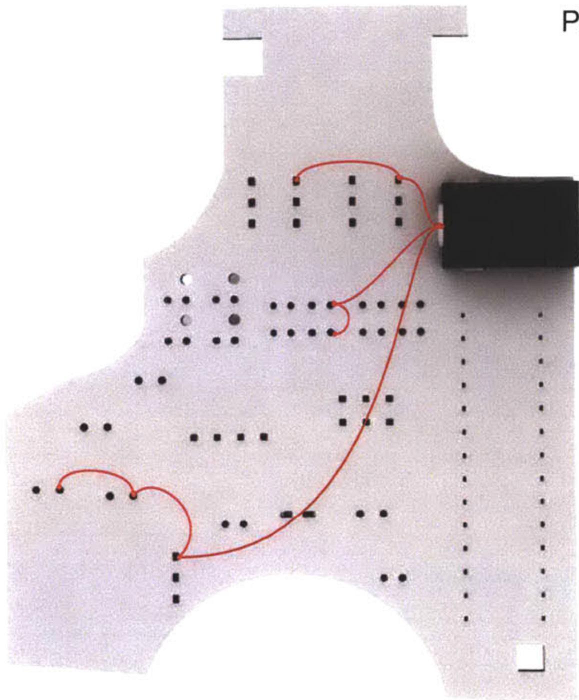


100 KOhm Resistors

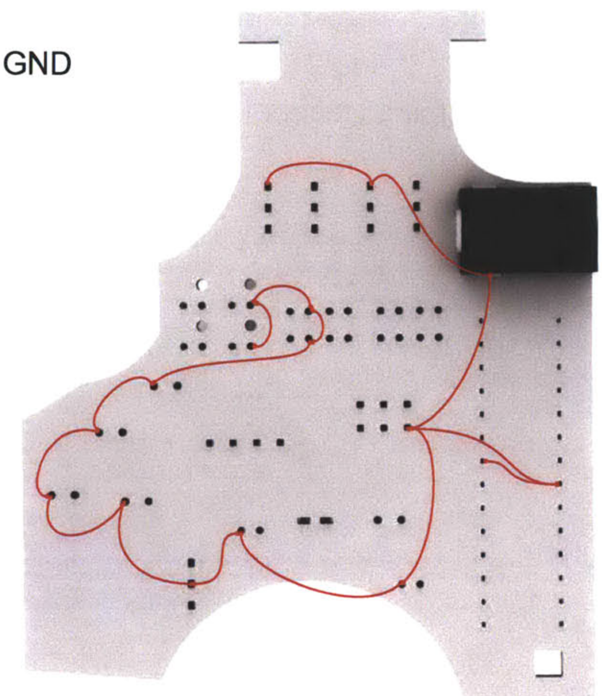


Wiring

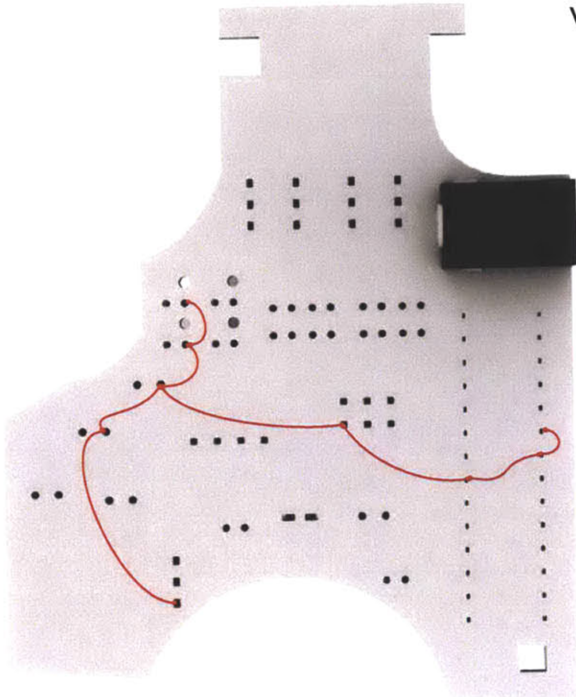
Power



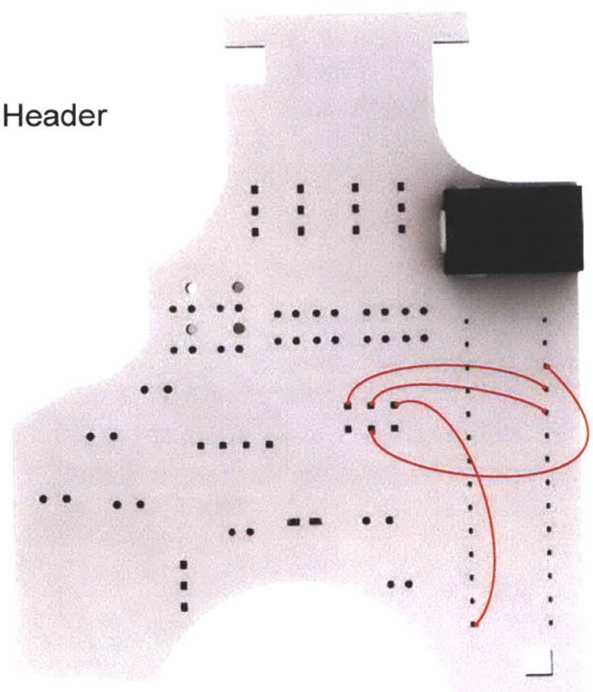
GND

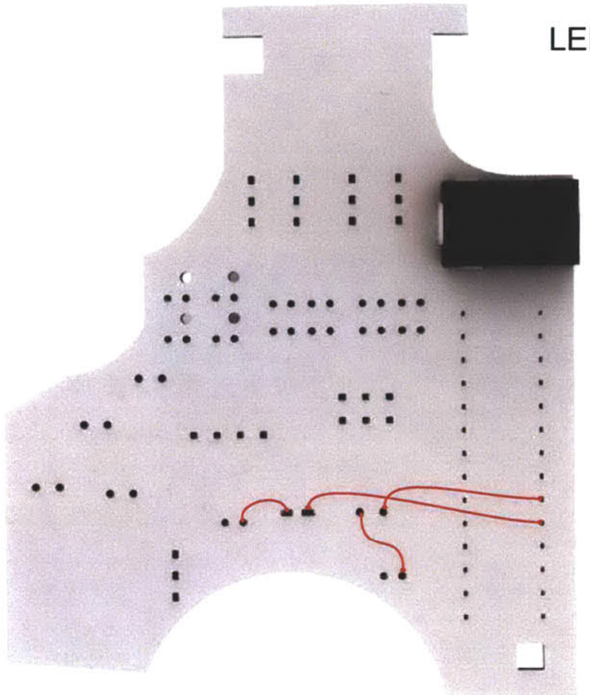


VCC



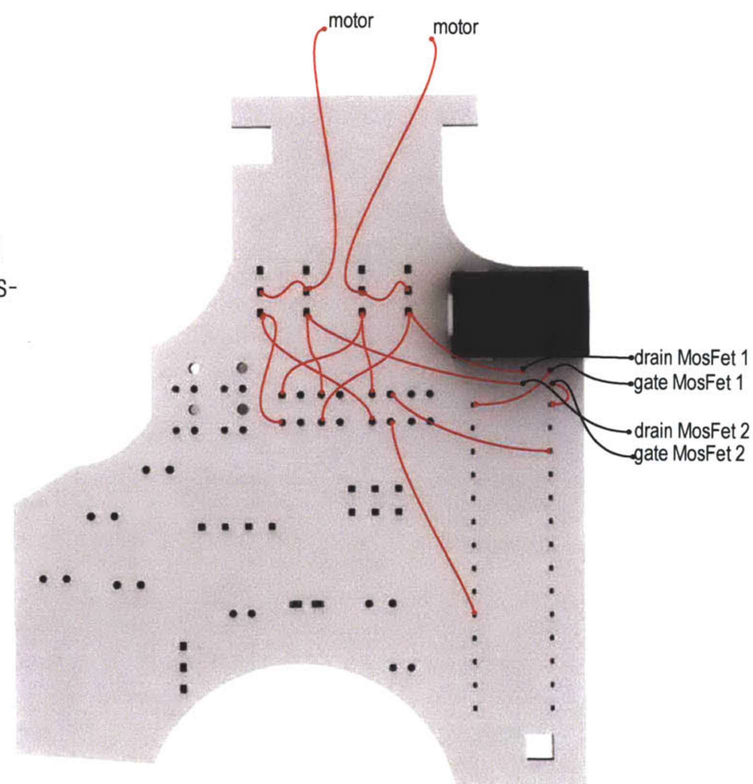
Programming Header





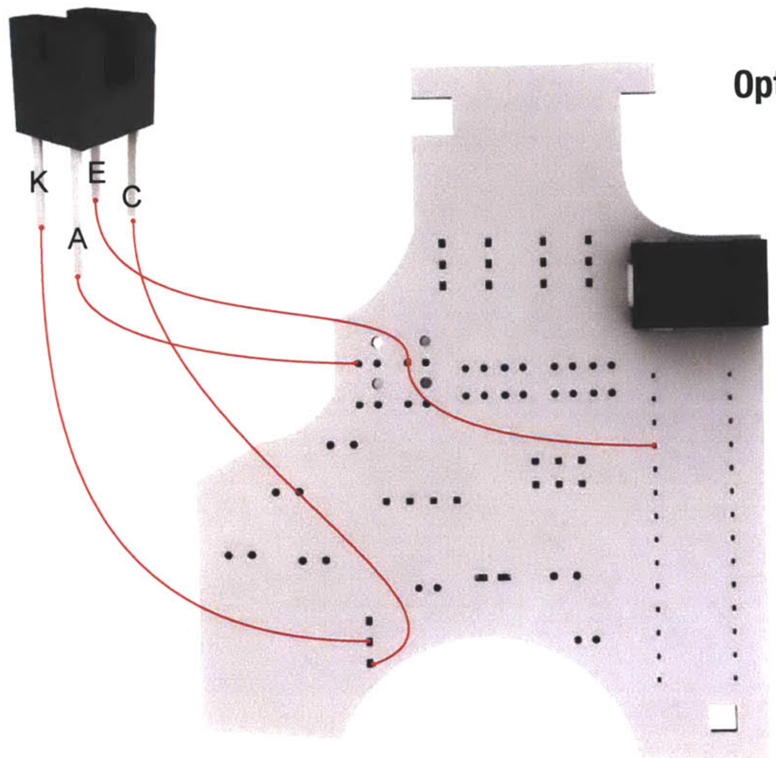
LED's

(At this point the LED test code can be uploaded and used to debug if necessary.)



H-Bridge

(At this point the H-Bridge test code can be uploaded and used to debug if necessary.)



Optical Interrupter

(At this point the Encoder test code can be uploaded and used to debug if necessary.)

5. Software
Pin Configuration:

PWM_fwrd	PIN B3	GREEN_LED	PIN C0
PWM_bwrd	PIN D3	BLUE_LED	PIN C1
ENABLE_fwrd	PIN B1	ENCODER_INPUT	PIN D
ENABLE_bwrd	PIN B0		

Header File:

```
////////////////////////////////////////
// main_prg.h
// SLAVE
//
// version 0.1
// Author: peter@media.mit.edu
// last edited: today
////////////////////////////////////////

#ifndef __MAIN_PRG_H__
#define __MAIN_PRG__

/// Register PWM_fwrd
#define DDR_PWM_fwrd DDRB
#define PWM_fwrd_PORT PORTB
#define PWM_fwrd_BIT _BV(PB3)

/// Register PWM_bwrd
#define DDR_PWM_bwrd DDRD
#define PWM_bwrd_PORT PORTD
#define PWM_bwrd_BIT _BV(PD3)

/// Register ENABLE_fwrd
#define DDR_ENABLE_fwrd DDRB
#define ENABLE_fwrd_PORT PORTB
#define ENABLE_fwrd_BIT _BV(PB1)

/// Register ENABLE_bwrd
#define DDR_ENABLE_bwrd DDRB
#define ENABLE_bwrd_PORT PORTB
#define ENABLE_bwrd_BIT _BV(PB0)

/// Register GREEN_LED
#define DDR_GREEN_LED DDRC
#define GREEN_LED_PORT PORTC
#define GREEN_LED_BIT _BV(PC0)

/// Register BLUE_LED
#define DDR_BLUE_LED DDRC
#define BLUE_LED_PORT PORTC
#define BLUE_LED_BIT _BV(PC1)

/// Register ENCODER_INPUT
#define DDR_ENCODER_INPUT DDRD
#define ENCODER_INPUT_PORT PORTD
#define ENCODER_INPUT_BIT _BV(PD1)

// Communication Address
#define TWI_SlaveAddress 0x01
#endif
```

LED and Encoder Test:

```
*/ Authors: peter@media.mit.edu.
*/ last edited: today
*/ structure of code:
*/       include header files
*/       declare global variables
*/       declare routines
*/       main
*/       interrupt handlind routines
*/////////////////////////////////////////////////////////////////////

*/ includes beginn
#include <avr/interrupt.h>
#include <avr/wdt.h>
#include "delay.h"
#include "main_prg.h"
#include <avr/io.h>
#include <util/delay.h>
#include <stdint.h>
#include <util/twi.h>
#include "TWI_Slave.h"
#include <string.h> // needed for memcopy needed for TWI communication
*/ includes end

*/////////////////////////////////////////////////////////////////////////////////////////////////

*/ global variables beginning
olatile uint16_t blink = 0; // used for LED alife blinking
olatile uint8_t speed = 0;
olatile uint8_t intense = 230; // intensity of brake
olatile uint8_t direction = 0; // direction 0 is fwrd, direction 1 is bwrd
olatile uint8_t count_up = 1;// count_up if 1 and count_down if 0
olatile uint16_t count = 0;
*/ global variables end

*/////////////////////////////////////////////////////////////////////////////////////////////////


*/ routines beginn
oid Motor_turn (uint8_t (speed))
    {
    if ( direction == 0)// means motor turn fwrd
        {
        OCR2A = speed; // create PWM for forward MosFETs
        }
    if ( direction == 1)// means motor turn bwrd
        {
        OCR2B = speed;
        }

    }

oid Motor_direction ( uint8_t (direction))
    {
    if ( direction == 0)// means motor turn fwrd
        {
        OCR2B = 0; // make sure the other side of the H-Bridge is deactivated!
        ENABLE_bwrd_PORT &= ~ENABLE_bwrd_BIT;// turn off bwrd
        ENABLE_fwrd_PORT |= ENABLE_fwrd_BIT;// turn on fwrd
        }
    if ( direction == 1)// means motor turn bwrd
        {
        OCR2A = 0;// make sure the other side of the H-Bridge is deactivated!
        ENABLE_fwrd_PORT &= ~ENABLE_fwrd_BIT;// turn off fwrd
        ENABLE_bwrd_PORT |= ENABLE_bwrd_BIT;// turn on bwrd
        }

    }

oid init_hardware(void)
    {
    //
    cli();// stop all interrupts

    //LEDs
    DDR_GREEN_LED |= GREEN_LED_BIT;// Data Direction (DDR) Output
    DDR_BLUE_LED |= BLUE_LED_BIT;

    // H_Bridge output and controll Pins
    DDR_PWM_fwrd |= PWM_fwrd_BIT;
    DDR_PWM_bwrd |= PWM_bwrd_BIT;
    DDR_ENABLE_fwrd |= ENABLE_fwrd_BIT;
    DDR_ENABLE_bwrd |= ENABLE_bwrd_BIT;

    // ENCODER_INPUT
    DDR_ENCODER_INPUT &= ~ENCODER_INPUT_BIT;

    //timers
    //timer counter 2, 8bit, PWM signal generation for H-Bridge opertaion
    TCCR2A |= _BV (COM2A1); // clear output pin on compare match when upcounting and set output pin on compare match when downcounting
    TCCR2A |= _BV (COM2B1); // same for output B
    TCCR2A |= _BV (WGM20); // activate phase correct PWM generation with 0xff as top
    TCCR2B |= _BV (CS20); // clock select prescaler 1, results in 8000000 and with 512 in 16khz PWM frequenzy
    TCNT2 = 0x000; // initialise timer counter with 0
    OCR2A = 0; // initialise OCR0A with 0, no speed (255 full speed)
    OCR2B = 0; // same for output B
    //timer counter 1, 16bit, motor encoder capturing
    TCCR1B |= _BV (CS10); //
    TCCR1B |= _BV (CS11); //
    TCCR1B |= _BV (CS12); // timer is increased on rising edge of pin TC1, encoder channel1
    TCNT1 = 0x0000; // initialise timer counter with 0
    //TIMSK1 |= _BV(TOIE1); //Timer Counter 1 overflow Interrupt enable
    //TIFR1 |= _BV(TOV1); //Timer Counter 1 Overflow Flag


    //internal Clock
    CLKPR = 0x80; // change internal clock prescaler to 1, so chip runs on 8MHZ instead of 1MHZ
    CLKPR = 0;

    SREG |= (1 << 7); // enable global interrupt
    sei(); // enable global interrupt
    }
*/ routines end

*/////////////////////////////////////////////////////////////////////////////////////////////

*/main program start

nt main (void)

*/ start up procedures, will only be executed once while powering on chip
nit_hardware();// initialise hardware

    // main loop
    while (1)
    {

        //comm_tasks();

        //LED alife blink
        blink ++;
        if ((blink > 60000) && (blink < 65000))
            {
            GREEN_LED_PORT |= GREEN_LED_BIT;// turn on RED_LED
            }
        else
            GREEN_LED_PORT &= ~GREEN_LED_BIT;// turn off RED_LED
        if (blink > 65000)
            blink = 0;

        //encoder read out
        if (TCNT1 > 7)
            {
            //BLUE_LED_PORT |= BLUE_LED_BIT;//turn on LED
            BLUE_LED_PORT = (BLUE_LED_PORT & ~BLUE_LED_BIT) | (~BLUE_LED_PORT & BLUE_LED_BIT);
            TCNT1 = 0;
            }


    }

*/main programm end

*/////////////////////////////////////////////////////////////////////////////////////////////
```

H-Bridge Test:

```
//////////////////////////////////////////////////////////////////////////////
// Breadboard code (TWI SLAVE)
// ATmega48
// Authors: peter@media.mit.edu,
// last edited: today
// structure of code:
//      include header files
//      declare global variables
//      declare routines
//      main
//      interrupt handlind routines
//////////////////////////////////////////////////////////////////////////////


// includes beginn
#include <avr/interrupt.h>
#include <avr/wdt.h>
#include "delay.h"
#include "main_prg.h"
#include <avr/io.h>
#include <util/delay.h>
#include <stdint.h>
#include <util/twi.h>
#include "TWI_Slave.h"
#include <string.h> // needed for memcopy needed for TWI communication
// includes end

////////////////////////////////////////////////////////////////////////////////////////////////////

// global variables beginning
volatile uint16_t blink = 0; // used for LED alife blinking
volatile uint8_t speed = 0;
volatile uint8_t intense = 230; // intensity of brake
volatile uint8_t direction = 0; // direction 0 is fwrd, direction 1 is bwrd
volatile uint8_t count_up = 1;// count_up if 1 and count_down if 0
volatile uint16_t count = 0;
// global variables end

////////////////////////////////////////////////////////////////////////////////////////////////////


// routines beginn
void Motor_turn (uint8_t (speed))
    {
    if ( direction == 0)// means motor turn fwrd
        {
        OCR2A = speed; // create PWM for forward MosFETs
        }
    if ( direction == 1)// means motor turn bwrd
        {
        OCR2B = speed;
        }

    }

void Motor_direction ( uint8_t (direction))
    {
    if ( direction == 0)// means motor turn fwrd
        {
        OCR2B = 0; // make sure the other side of the H-Bridge is deactivated!
        ENABLE_bwrd_PORT &= ~ENABLE_bwrd_BIT;// turn off bwrd
        ENABLE_fwrd_PORT |= ENABLE_fwrd_BIT;// turn on fwrd
        }
    if ( direction -- 1)// means motor turn bwrd
        {
        OCR2A = 0;// make sure the other side of the H-Bridge is deactivated!
        ENABLE_fwrd_PORT &= ~ENABLE_fwrd_BIT;// turn off fwrd
        ENABLE_bwrd_PORT |= ENABLE_bwrd_BIT;// turn on bwrd
        }

    }

void init_hardware(void)
    {
    //
    cli();// stop all interrupts

    //LEDs
    DDR_GREEN_LED |= GREEN_LED_BIT;// Data Direction (DDR) Output
    DDR_BLUE_LED |= BLUE_LED_BIT;


    // H_Bridge output and controll Pins
    DDR_PWM_fwrd |= PWM_fwrd_BIT;
    DDR_PWM_bwrd |= PWM_bwrd_BIT;
    DDR_ENABLE_fwrd |= ENABLE_fwrd_BIT;
    DDR_ENABLE_bwrd |= ENABLE_bwrd_BIT;

    //timers
    //timer counter 2, 8bit, PWM signal generation for H-Bridge opertaion
    TCCR2A |= _BV (COM2A1); // clear output pin on compare match when upcounting and set output pin on compare match when downcounting
    TCCR2A |= _BV (COM2B1); // same for output B
    TCCR2A |= _BV (WGM20); // activate phase correct PWM generation with 0xff as Top
    TCCR2B |= _BV (CS20); // clock select prescaler 1, results in 8000000 and with 512 in 16khz PWM frequenzy
    TCNT2 = 0x000; // initialise timer counter with 0
    OCR2A = 0; // initialise OCR0A with 0, no speed (255 full speed)
    OCR2B = 0; // same for output B


    //internal Clock
    CLKPR = 0x80; // change internal clock prescaler to 1, so chip runs on 8MHZ instead of 1MHZ
    CLKPR = 0;

    SREG |= (1 << 7); // enable global interrupt
    sei(); // enable global interrupt
    }

// routines end

//////////////////////////////////////////////////////////////////////////////////////////////////////////


//main program start

int main (void)
{

// start up procedures, will only be executed once while powering on chip
init_hardware();// initialise hardware

    // main loop
    while (1)
    {

        //comm_tasks();

        //LED alife blink
        blink ++;
        if ((blink > 60000) && (blink < 65000))
            {
            GREEN_LED_PORT |= GREEN_LED_BIT;// turn on RED_LED
            }
        else
            GREEN_LED_PORT &= ~GREEN_LED_BIT;// turn off RED_LED
        if (blink > 65000)
            blink = 0;
```

```
            // turn motor fwrd
            if (direction == 0)
            {
                Motor_direction (direction);
                // accelerate and decelerate Motor
                if (count_up == 1)
                {
                    if ( speed < 200)
                        {
                        if ( count < 500)
                            {
                            count ++;
                            }
                        if ( count == 499)
                            {
                            count = 0;
                            speed ++;
                            Motor_turn (speed);
                            }
                        }
                    else
                        count up = 0;
                }
                if (count_up == 0)
                {
                    if ( speed > 5)
                        {
                        if ( count < 500)
                            {
                            count ++;
                            }
                        if ( count == 499)
                            {
                            count = 0;
                            speed --;
                            Motor_turn (speed);
                            }
                        }
                    else
                        {
                        count_up = 1;
                        direction = 1;
                        }
                }
            }
            // turn motor bwrd
            if (direction == 1)
            {
                Motor_direction (direction);
                // accelerate and decelerate Motor
                if (count_up == 1)
                {
                    if ( speed < 200)
                        {
                        if ( count < 500)
                            {
                            count ++;
                            }
                        if ( count == 499)
                            {
                            count = 0;
                            speed ++;
                            Motor_turn (speed);
                            }
                        }
                    else
                        count_up = 0;
                }
                if (count_up == 0)
                {
                    if ( speed > 5)
                        {
                        if ( count < 500)
                            {
                            count ++;
                            }
                        if ( count == 499)
                            {
                            count = 0;
                            speed --;
                            Motor_turn (speed);
                            }
                        }
                    else
                        {
                        count_up = 1;
                        direction = 0;
                        }
                }
            }

        }
}
//main programm end
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
```