Proceedings of the ASME 2013 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2013
August 4-7, 2013, Portland, Oregon, USA

DETC2013-13130

UNDERSTANDING ABSTRACTION IN DESIGN: A COMPARISON OF THREE FUNCTIONAL ANALYSIS METHODS FOR PRODUCT DISSECTION

Joran W. Booth*
boothj@purdue.edu
Graduate Student

Tahira Reid
tahira@purdue.edu
Assistant Professor

Karthik Ramani
ramani@purdue.edu
Professor

School of Mechanical Engineering
Purdue University
West Lafayette, Indiana, USA

ABSTRACT

In design classes, functional analysis is a process that is typically used to assist students with identifying essential functions to aid in the development of their concepts. However, it has been observed that students sometimes struggle with this part of the design process. In this study, a group of 26 students were studied in a 3-level within-subject study (n=78) to determine which of three common functional analysis approaches (i.e. top-down, energy-flow, and unstructured) was most effective. Participants were asked to dissect a hair dryer, power drill, and NERF pistol and generate function trees describing how these work. Measures of effectiveness include the number of functions generated, the number of errors, the number of levels of abstraction represented in the tree, and the number of unique subsystems and functions identified. No statistical difference between the approaches was found, and there was also no practical difference between the approaches. These results suggest that for novice engineers, there is no difference between methods used. This possibly indicates that for novice engineers, formal methods may not be any more effective than an unstructured approach.

INTRODUCTION

It has been observed in our classes and research that when students create function diagrams and corresponding morphological charts, most students struggle with identifying appropriate functions. Additionally, many students confuse function, objective, and part hierarchies. Often, portions of their function trees include part hierarchies or functions that do not correspond to their design problem. It has also been observed that students who have learned functional decomposition as sophomores do not remember having learned it once they reach senior level design courses, or apply it so ineffectively, that it is as if they have never been taught in the first place. Since functional decomposition is common in complex design, it is important to identify the best methods and practices associated with it [1-2].

As a result of our pilot study, we decided to narrow our study of functional decomposition to functional analysis of mechanical dissection scenarios and exclude new and iterative design scenarios. As will be discussed later, this study was narrowed to reduce the variance of data produced by the participants.

This study examines which methods are most effective for mechanical dissection and developing mechanical abstractions for novice designers. Future work in understanding design abstraction is also discussed. Identifying the most effective methods is also critical for future work in design abstraction.

Except when discussing terms used by other authors, “Functional analysis” is understood to be the process of identifying functions when reverse engineering (i.e. well-defined problems), and “functional synthesis” is defined as identifying functions for new or iterative design scenarios (i.e. ill-structured problems.) “Functional decomposition” describes both synthesis and analysis simultaneously, and is defined as the general method of identifying functions (for new design, iterative design, or reverse engineering).

The output of functional decomposition is understood to be “the solution-neutral [or embodiment-neutral] detailed description of what are the intentions for the products” [3]. It is one of several activities in design, including mathematical modeling [4-6], value analysis [1, 7], communication [8], ideation [9-10], visualization, etc., that represent how designers use abstraction to creatively solve problems. Abstraction hierarchies (i.e. function-means trees, and other function diagrams) are essential elements to help define design spaces,

* Corresponding author

and thus “designers… make extensive use of artificial symbol systems” [4].

Since abstraction is a feature of deep learning and problem solving, functional decomposition must be studied in the context of learning theories and cognition.

This paper regards design to be primarily an engineering activity which solves ill-structured problems [11], and which is composed of various elements which are not exclusive to it [4]. However, the authors recognize that design can also be interpreted to include other fields and disciplines [12].

An abstraction is an easily manipulated and generalized “idea [or] representation… [which] become[s] the basis for elements in the configuration space” [13]. Abstraction is also a flexible representation of knowledge associated with deep learning, which allows transfer to new domains [14]. This property allows abstractions to be dynamically modified to fit new situations, and potentially improve creativity, as seen in ideation or visualizing mental images. One particular property of abstraction is that as new learners develop into skilled learners or experts, their abstraction effort moves from effortful learning to immediate perception [14].

BACKGROUND

Abstractions are found extensively in design, including mathematical models, design spaces, and functional representations. It is used to map complex ideas from one domain to another. Designers use abstraction to develop mathematical and physical models (e.g. F=m*a) for later design [4-6], determine the value and costs associated with a design [1, 7], use visual tools for thinking and problem solving [15], use abstractions to communicate with other designers [8], and to transcend similar disciplines (e.g. the way that F=m*a is applied very differently in fluid mechanics, dynamics, statics, etc. and yet all derive from the same abstraction).

In the problem definition phase, several methods are used to elucidate the problem, including objectives trees, QFD, listing customer requirements, ethnographic observations, and functional decomposition. Functional decomposition is of particular interest since our anecdotal observations have shown that students do not understand this process nor how to effectively use it.

Abstraction as Functional Decomposition

Functional decomposition helps designers take developing design ideas and abstract them into a functional space [13] and then embody these into concepts. Idea generation generally represents the reverse process of generating a concrete design concept from an abstract basis [16].

A survey of engineering design textbooks and two papers revealed only three unique methods: energy-flow, top-down, and unstructured methods (Table 1). Although the majority of these methods are intended as a step prior to concept generation, many are conflated with methods for identifying functions in reverse engineering. There is no theory, nor reasoning, given in these texts to support the assumption that methods for new design should also be effective for reverse engineering. Additionally, there is no distinction made between “functional decomposition”, “functional analysis”, “reverse engineering”, and dissection”, except in a few cases. The ambiguity between these terms only adds to the conflation of these activities.

The majority of methods taught are derived from Pahl and Beitz who originally described the energy-flow approach [17]. This approach considers material, energy, and information flow through a system. The system and subsystems are divided into “black-boxes” whose inputs and outputs are defined by the flows into and out of them. These black-boxes can be examined in more depth, although most authors only discuss a primary and secondary set of functions (i.e. no further levels). This approach uses a flow-block diagram for each level, but makes no provision for diagrammatically combining the functions into a hierarchy. A variant of the energy-flow approach is the “black-box” approach, which does not make an emphasis on flow types, but still emphasizes the use of flow-block diagrams, as the energy-flow approach. There is significant variance in how these terms are applied throughout the various texts.

The second most common method was an unstructured method where the authors gave no direction regarding how to identify functions or simply suggested to the designer that they enumerate the functions. This approach sometimes recommends a list, and at other times recommends a function tree or function-means tree.

The least common method was the top-down approach, which prescribed selecting the top-most system and then breaking it into subsystems and discovering their functions. This approach uses function trees or function-means trees to organize its functions.

While there are many studies on abstraction activities in design, and many methods for abstracting, these methods are derived from observing industry, theory, or citing prior authors. There does not seem to be any cognitive theory that has enough detail to describe why functional decomposition works (or does not).

A Cognitive Model of Abstraction

A functional understanding of a device is an abstraction, and thus, any discussion of functional decomposition is incomplete without being grounded in cognitive psychology. Jansson and Smith [13] proposed a cognitive model for understanding abstraction in design. They describe the generation of new ideas as moving from one point in the configuration space to another by abstracting to a concept space first.

A more modern, but less complete version of this original model can be found in the bridge model proposed by Dubberley, Evenson and Robinson [16]. This model describes moving from “what is” to a “model of what is” (abstracting). Then the model is expanded into a “model of what could be” and finally embodied in “what could be”. The authors also propose generalized methods for abstracting ideas.

TABLE 1 - ENGINEERING DESIGN TEXTBOOKS AND THEIR TREATMENT OF FUNCTIONAL DECOMPOSITION

Authors	Method	Reverse Engineering	New/Iterative Design	Wording Used in Text
Dym and Little [18]	Black-box	x	x	Black boxes / transparent boxes
Ulrich and Eppinger [19]		x	x	Functional decomposition
Cross [20]			x	Functional analysis
Stoll [7]			x	Functional analysis / functional decomposition
Ullman [9]			x	Functional modeling / functional decomposition
Pahl and Beitz [17]	Energy-flow	x		Functional interrelationship
Ullman [9]		x		Product decomposition
Dieter [1]			x	Functional decomposition
Hyman [2]			x	Functional analysis
Dym and Little [18]	Function-means		x	Function-means tree
Cunniff et al. [10]	Top-down	x	x	Functional decomposition / reverse engineering
Phillips [21]		x		Functional decomposition
Dym and Little [18]	Unstructured	x	x	Enumeration of functions
Dym and Little [18]		x		Reverse engineering / dissection
Horenstein [22]		x		Reverse engineering
Sheppard [23]		x		Mechanical dissection
French [24]			x	Functional analysis
Magrab [25]			x	Functional decomposition / functional analysis
Priest and Sánchez [26]			x	Functional allocation

Design as Abstraction and Fixation

Design fixation is the inhibition of design abstraction [13]. As seen in Table 2, design abstraction is loosely associated with certain design phases. However, just as in learning, if a prior level of understanding is not fully achieved before moving on, subsequent methods and phases in design will not be effectively executed. For example, if a designer does not fully understand the patents they have collected (concrete levels), and how these fit into the broader picture of the design (transition level), the designer cannot transfer principles from these patents to new situations (i.e. the design is functionally fixated.) Thus, activities whose goal is to help a designer develop abstractions of knowledge, such as functional decomposition, may actually reduce the effects of fixation.

According to Jansson and Smith, there are two types of design fixation: functional fixation and mental set fixation. Mental set fixation is a temporary inhibition in thinking that may be overcome easily by stepping away for a moment and coming back to the problem. An example of this is when solving 10 math problems where the first 9 are solved the same way and the 10^{th} is solved a different way. Many people are stuck on the 10^{th} problem, but may solve it easily after a brief rest period.

Functional fixation, on the other hand, is not temporary, and is much more difficult to overcome. Functional fixation refers to the inability to see potential functions in an object. An example of this is a problem where two strings hang from a ceiling and a person must grasp both strings, but the strings are spaced such that a person cannot grasp one string and then grasp the other. If a pair of pliers is provided, most people will attempt to use the pliers as an extension of their grip, since pliers are typically used to grip things. However, the solution to the problem is to use the pliers as a pendulum so as to bring the other string within reach, thus using the pliers for a function well outside their obvious use. The inability for most people to consider function independent of form is referred to as functional fixation [13].

Jansson and Smith leave us without any methods to overcome functional fixation, and considerable research has been conducted to further investigate its nature or how to overcome it (e.g. [27]). Although this category of research has deep implications for design abstraction, research in abstraction activities, such as functional decomposition [28-29], have been conducted in nearly complete isolation. To date, we have found only two papers that have begun explore how well abstraction activities, in this case product dissection, serve to reduce fixation [30-31].

TABLE 2 - BLOOM'S TAXONOMY AND A PROGRESSION OF RELATED DESIGN PHASES

Abstraction Level	Bloom's Taxonomy	Related Phases in Design
Concrete	Knowledge – Ability to recall facts, details, etc.	Data gathering, developing customer requirements, developing engineering specifications, etc.
	Comprehension – Understanding the meaning of the facts, and able to summarize	
Transition	Application – Ability to transfer concrete ideas to new situations using rules, concepts, principles, laws, or theories	Benchmarking, patent analysis
Abstract	Analysis - Ability to break ideas into component parts and understand their organization	Product dissection, engineering modeling, problem definitions, systems management
	Synthesis – Ability to develop a new set of abstract relations, or reorganize these into a new set	Concept generation and ideation
	Evaluation – the ability to judge the value of something	Valuation of ideas, market trend analysis

Additionally, our literature review failed to uncover subsequent research in fixation that distinguishes which type of fixation is observed in the study. This is a serious oversight considering the very different properties of functional and mental set fixation.

Abstraction as a Learning Activity

In addition to viewing abstraction as a cognitive phenomenon, abstraction may also be viewed as deep learning. The concept of abstraction is often described as the deepest levels of learning, such as in Bloom's taxonomy of learning [32] and Skill Theory [33]. Skill Theory, for example, describes a progression from concrete facts and details as the lowest level of learning to a generalized or abstracted form of ideas that are able to be synthesized into new ideas. The design process can be seen as a learning process, and design phases can be mapped to the learning levels (see Table 2.)

From this cognitive learning view of abstraction, learning theories such as constructionism [34] and constructivism [35] suggest that activities such as sketching, prototyping, and other participant-involved activities can lead to improved abstraction. This can be seen in a study of a middle-school science program where prototyping and design skills were used as scaffolding for the students, who spontaneously discovered abstract scientific principles for themselves [36].

Learning theories also tell us that different classes of problems require different levels of thought and learning. Ill-structured problems require the most complex types of thinking, whereas other types do not require the same depth [11]. Education can be considered to be training students to solve problems [11], and abstraction is the key that allows students to solve the most complex forms of problems [14]. However, not all problems within engineering and design education are of the same nature; some are ill-structured, while others are well-defined [37].

Ill-Structured Synthesis Vs. Well-Defined Analysis

One limitation in studying functional decomposition is that the nature of product dissection is well-structured (there is a coherent answer) whereas design problems are ill-structured [11]. Although many authors promote the idea that functional decomposition works equally well for well-defined and ill-structured problems, past product dissection methods for determining functionality were not viewed as compatible with design ideation methods [20]. According to Stoll, "there is no really objective or systematic way for dividing a given design problem into sub-functions." [7]. This disparity in problem type means that the results of this study cannot be extrapolated to design problems [4]. While these two forms of functional decomposition are fundamentally different, product dissection is still valuable to study as a design tool since it is a common and well-established process in industry [3]. As observed in the pilot study, it would be more appropriate to call functional decomposition in reverse engineering scenarios "functional analysis" and in design scenarios "functional synthesis". This would be more in line with the distinction made between kinematic synthesis and analysis and other uses of "synthesis" and "analysis."

METHODOLOGY

When this study was first started, it had been anecdotally observed that senior-level engineering students rarely participated in functional decomposition activities. Many students didn't even know what functional decomposition was, despite having been taught previously in a sophomore-level class. It seemed clear that teaching functional decomposition on the sophomore level was important to later success.

At the beginning of this study, we sought to understand how to best teach functional decomposition such that students understand it and are able to effectively apply it when designing new products.

An initial pilot study was conducted in a sophomore-level design course to test some of the hypotheses in a casual setting and collect qualitative observations from several design instructors, including the authors of this paper. The course teaches functional decomposition for new product design as described in Ullman [9].

Pilot Study

In the sophomore-level design course, students are asked to design a new product, which in this case was a mechanically interactive toy. Functional decomposition was taught both in lecture and reinforced in lab (which was taught by the first and second authors.) This setting was chosen since the mechanical complexity of the course projects was similar to those that would be dissected by research participants. Students were given the same explanations of the function tree and function-verb lists that the study participants received.

In some of the sections, the design students were instructed to do a second iteration of the function tree after completing a round of ideation so their functions could be more specific. Overall, function trees seemed to work more readily for some teams than others.

For the teams who struggled with using function trees, the process of creating the tree did not seem to be a natural interface for organizing the ill-structured thoughts. One team reported, “I feel like the structure of the function tree process is actually hindering us from coming up with concepts. I feel like you have to already know what you are building in order to do it.” When the hindered team was asked how they would do it without function trees, they responded that they would focus on customer requirements directly and not focus on functions.

Some of observed teams used the function tree as a sort of objectives-ideation session. If their function was "provide sensory stimulation" for a toy, sub-functions included "stimulate visual senses", "stimulate auditory senses", and "stimulate tactile senses". The final toy concept they actually design may not incorporate all three of these, but students reported that exploring the breadth gave them new ideas. Additionally, some teams who were stuck seemed to understand the function tree much easier when the functions were explained to be like "goals" for their future design. This was done with questions like, "What would you like your toy be able to do? What goal should it accomplish?"

One intervention proved particularly useful in distinguishing functions from objectives or concepts. A few groups were shown examples of morphological charts and told, "If you cannot conceptualize or if you can use a sketch to describe what you wrote, then you don't have a function". Students seemed to find this approach helpful in making the distinction between some of the items they listed in their trees.

Overall, enforcing the verb-phrase format was simple and students seemed to understand that concept relatively easily if they received immediate feedback on their work. Students did not like the format, though. One student reported, “It is just awkward to have to think of actions and verbs.”

Through the course of the pilot study, a few things became apparent.

1. The function-verb list [38] doesn't support behavior change or psychological functions such as “encourage cooperation”, “provide entertainment”, or “being fun”
2. The function trees couldn't be as detailed as the ones in the study since many of the functions are concept dependent
3. There is often overlap between customer requirements and required design functions, but no clear way to indicate these, such as a requirement for safety. [1]
4. A completely different approach was needed, and that energy-flow, top-down, or any dissecting method could not provide for the synthesis of functions.
5. Toys carry complex and deeply social meanings [39] that similarly complex products such as a hair dryer do not. As a result, designing a toy will have different implications than designing a hair dryer or power drill.
6. A design context has much less structure to the problem than a reverse engineering task.

The results of the pilot study showed that functional synthesis in design produced highly variable results. In order to control an experiment with functional decomposition, mechanical dissection was chosen as the vehicle for further testing of the various methods. This is because the results are naturally convergent. Additionally, before further research could be conducted on how to best teach functional decomposition to engineers, we needed to identify the best methods for novices. This led us to our refined research question.

Which of the common functional decomposition methods is most effective?

Design of Experiment

The research question was converted into four hypotheses (in addition to the null hypothesis.) These hypotheses permitted a statistical analysis to aid in answering the research question.

- **H_0** – There is no difference between any of the methods as measured by total number of functions, number of unique functions, number of errors, number of levels of abstraction, and usefulness to the students.
- **H_1** – The top-down approach will prove to be more effective at a.) generating the total number of functions, b.) generating the fewest errors, and c.) generating the most levels of abstraction.
- **H_2** – The energy-flow approach will generate the most unique functions
- **H_3** – The unstructured approach will generate a.) the fewest functions and b.) the fewest unique functions.
- **H_4** – Older students will perform better than younger students.

[1] This is probably due to Ullman calling functions the “whats” of a design, and this same word being used to describe the customer requirements in the QFD process [9]. Hyman clarifies this confusion with his definition of a function. “A particular function may not be explicitly related to any of the objectives, but may be necessary in order to achieve an objective.” [2]. In addition to this, confusions between objectives and functions in novices and fledgling designers may be due to a “newness” of these concepts and the similarity between tools focusing on these two concepts [18].

TABLE 3 - EXPERIMENTAL LAYOUT OVER 4 WEEK PERIOD

	Week 1: Hair Dryer	Week 2 : Power Drill	Week 3	Week 4: NERF Pistol
Session A	Top-Down	Energy-Flow	N/A	Unstructured
Session B	Unstructured	Top-Down	N/A	Energy-Flow
Session C	Energy-Flow	Unstructured	N/A	Top-Down

In order to answer these, a quantitative design of experiment was used, while also utilizing qualitative methods for understanding the context and environment and enriching the meaning of the experiment. The experiment conditions were determined using a within-subject experimental structure with three-level factors. A within-subject experiment design was used to increase the amount of data and also reduce the effect of uncontrolled variables such as time of day and self-selection bias [40].

Participants were asked to dissect three products and use three methods for determining the functionality of those products. A session began with instruction on how to create a function tree and instruction on the decomposition method. Participants were then asked to physically dissect a product and determine its functionality using a top-down approach, an energy-flow approach [9], or an unstructured approach [28]. After disassembly, students were asked to diagram a function tree describing the functions of the dissected product. Students also completed an initial survey to determine basic demographic information and three post-surveys to evaluate the quality of the activity and prior exposure to disassembling the product. After the function trees were submitted, participants were shown how the product works and relevant engineering equations relating to certain physical phenomena.

In order to ensure consistency, instruction packets were given to each participant by a course instructor. A packet describing function trees and semantic formatting (verb-noun format) was given and explained to participants in each session. Participants were also provided with a list of function verbs. The function verb list was taken from the pruned function list by Caldwell et al. [38] though the hierarchy of the verbs was not retained. Participants were also given a packet explaining how to perform the method for the experimental session.

Data was collected over a four week period in October 2012, as shown in Table 3. All sessions were held on Thursdays at the same times each week.

Qualitative data from the study consisted of casual observations made by the authors and other test administrators. Observations were recorded immediately after testing periods. No formal protocol was used to complete the observations. Only field notes and debriefing notes were recorded. These observations were made at the length of the study.

Independent Variable - Abstraction Method

The independent variables used in the abstraction method were as follows:

- Top-Down – Start with the highest level of abstraction (the whole machine) and determine overall function. Break down into sub-systems and determine functions of each of these systems. Iteratively become more detailed for each level. Write these functions into a tree.
- Energy-Flow – Map the flow of mass and energy through a system. Each transformation of energy, mass, or information is a function. This should be done separately on various levels before constructing a tree.
- Unstructured (no method, control) – Write down relevant functions as they seem appropriate in whatever order they come to mind. Organize these into a tree. (Participants were told the name of this method was "important things first" so as to not bias them.)

The independent variable interventions used were chosen due to their common usage by engineering professionals. However, the top-down and energy-flow approaches tend to be used more commonly by more experienced professionals and unstructured approaches by recently graduated professionals [28].

Dependent Variables

The dependent variables measured include:

- Number of functions generated – the number of functions recorded on each function tree regardless of whether it is correct or not.
- Number of functions incorrect – the number of functions written incorrectly. Errors recorded include using a part name instead of a function, giving intentionally incorrect functions (i.e. "meow"), showing a clear misunderstanding of how a part works, and describing a behavior rather than a function. When possible, these errors are interpreted as functions (i.e. "grate" translates to "prevent debris from entering fan"). Incorrect functions are only deducted from the number of unique functions when there is not clear connection to a unique function.
- Number of unique functions identified – determined qualitatively by grouping all generated functions by keywords and context into a subset of functions and counting the number of unique functions generated by each participant.
- Levels of abstraction created – the maximum number of levels of abstraction in a particular function tree hierarchy.
- Survey results – the student responses regarding the usefulness of activity, out of 10, with 10 being high.

The level of understanding is triangulated with these several variables, each intending to show slightly different things. For example, the number of functions generated is a measure of detail, whereas the number of unique functions determines the breadth of understanding demonstrated by the participant.

The most important of these variables is the number of unique functions created, because this gives some indication of

TABLE 4 - AN EXAMPLE OF HOW TRANSCRIBED FUNCTIONS WERE CLUSTERED INTO A SINGLE UNIQUE FUNCTION. THE TABLE SHOWN IS TAKEN FROM THE DATA SET, AND SHOWS TWO EXAMPLES OF FUNCTION CLUSTERING IN GRAY.

		Error Type							
ID	**Participant-Generated Function**	**Wording**	**Described Part**	**Described Behavior**	**Incorrect Physics**	**Left Blank**	**Parent-Child Relationship**	**Generalized (Unique) Function**	**Tree Level**
A4	resistors give off heat			Yes			A4.2.1.1	generate heat	4
	activate switch to low to let less current flow to resistors						A4.2.2	change mode	3
	resistors give off heat		Yes	Yes			A4.2.2.1	generate heat	4
	press snowflake button to disconnect resistors from circuit		Yes	Yes			A4.2.3	switch heat mode	3
	direct air flow						A4.3	direct flow of air	2
	vents on fan help direct flow		Yes	Yes			A4.3.1		3
	contour of barrel concentrates flow towards center		Yes	Yes			A4.3.2		3
	vent at end of barrel further regulate air flow		Yes	Yes			A4.3.3	control flow rate	3
	vent on back end of barrel allows air from outside to be drawn in		Yes	Yes	Yes		A4.3.4	keep debris out of fan	3
	mesh distributes incoming air evenly across fan		Yes	Yes	Yes		A4.3.5		3
	hold pieces together and prevent breaking						A4.4	fasten components	2
	heat shield prevents plastic from melting		Yes	Yes			A4.4.1	deflect heat	3
	guides and clips hold components in place		Yes	Yes			A4.4.3	fasten components	3

the level of overall understanding the participant had of the device, whereas other measures only give an idea of how accurate or how detailed a participant can be. The number of functions, number of errors, and levels of abstraction indicate how concise a particular method was in finding specific functions, and help measure the depth of detail.

Function trees generated are not compared to a master function tree as done in other studies since there is not a single, unique solution that is most appropriate [10, 19]. To adequately test an engineer's functional understanding, an appropriate measure must qualitatively evaluate how thoroughly the engineer understands the functionality. The number of unique functions allows dramatically different hierarchies to be compared with a uniform measure. Additionally, we did not compare generated trees to a master tree, because the study is not about which method helps novices match some standard, but rather which method helps students generate their own functions.

Measurement and Data Transcription of Dependent Variables

At the completion of the study, the hand-written function trees produced by the students and survey results were transcribed into a database with anonymous identifiers. Each function was recorded verbatim and linked to its parent and any children. The individual functions were also grouped into generalized functions based on verbal affinity clustering (conducted by the authors). These generalized functions were refined through three iterations, the first of which was conducted at the initial transcription. The generalized functions were created by clustering every function generated, regardless of the participant who generated it. One effect of this analysis is that one participant may have many functions combined into a single function.

Table 4 shows two examples of how this affinity clustering took place. In the first example, shown in gray, subject A4 writes three different functions, each of which describe a more general function of directing air flow through the hair dryer. These are then described as a single unique function. Similarly, the second sets of functions (in gray) both describe "keeping debris out of the fan". There is a minor sacrifice in fidelity in doing this; however the loss of information is minimal. In the case of the second set of functions, errors are also mitigated (the mesh may distribute the air across the fan, but this is NOT an intended function of it.) However, as in the example shown, erroneous functions are still within the domain of the more generalized function.

Table 4 also shows several examples of errors, such as syntactical errors, descriptions of behaviors rather than functions, and a lack of understanding of the operation of the device.

Finally, the functions were also marked for common mistakes, such as using a part name instead of a function, a clear lack of functional understanding (i.e. “power solenoid to move air” to describe the operation of a motor and fan.), problems with wording, describing a behavior instead of a function, intentional errors (one student put “meow” as a function), or omitting a primary function and only listing secondary ones.

Covariates

- Class level – freshman, sophomore, junior, or senior.
- Frequency of disassembly outside of class – how often participants take things apart outside of class, on a three point scale (never/rarely, sometimes, often).
- Prior experience with disassembling product – if the participants have taken apart the device for a particular session before. Data is binary (yes/no).
- Learned functional decomposition before – if the participants have been exposed to a functional decomposition strategy before. In theory, every junior and senior has been exposed to the energy-flow approach in the sophomore level design class at Purdue. Data is binary (yes/no).
- Rough draft used – whether a participant created a rough draft before making a function tree. Data is binary (yes/no).
- How helpful was the activity? – self-reported on a scale from 1 (low) to 10 (high).

Population

Each session consisted of varying numbers of participants due to how scheduling for the class was conducted. The first session, held at 9:30AM had 8 participants; the second session, held at 11:30AM had 12 participants; and the third session, held at 1:30PM, had 6 participants. All sessions were held on Thursdays. Between all sections, 10 students identified as sophomores, 8 as juniors, and 7 as seniors with one not reporting and no freshmen (although the class is open to them). All participants were in mechanical engineering. More than half (14) participants reported not having learned functional decomposition before. This result is particularly surprising since more than half the students had taken ME 263, where functional decomposition is explicitly taught and included in two significant work assignments. Many of the juniors and seniors reported not having learned functional decomposition before. Participants were not provided books, materials, or electronic devices from which to work.

The population studied is a convenience sample. Participants were recruited from a product dissection class offered through the Mechanical Engineering Ambassadors at Purdue University. The class is taught by undergraduate students and does not include tests or homework. Grading is based on in-class attendance, participation, and a final project where students must choose a device to analyze. Students in the class are not taught any method to analyze functions.

Survey results showed that many students choose to take the class because they would like to understand how things work, to have a lighter course load, add an extra credit, take and easy class, or to have fun.

Although the general engineering population was not sampled, it appears that these differences may be minimal. The participants may be interested in understanding product functions, recognize a need for functional understanding, and be more assertive in their education than the general mechanical engineering body. These biases may influence the population to try harder than other students to apply the methods taught.

Additionally, the students in the course are divided into three sections, and the students may have self-selected on the basis of the time each section is held. This effect was controlled by using the within-subject study.

It is the judgment of the authors that the participants represent an adequate sample of novice engineers, though validation of this judgment was not made.

Testing Environment

Since the environment can have an effect on occupants and their work [41], it is important to describe the room in which the class is taught. The class is taught in lecture room with wooden floors, no windows, chalkboards, and students sit in metal chairs with attached desks. The attached desks are no larger than 10 inches by 10 inches and at a slight angle such that round objects roll toward the seated student. There is no convenient way for students to work as teams, and taking apart devices poses a challenge due to the limited workspace. Occasionally, when parts accidentally spill, students will work on the floor to reinsert ball bearings into a slot. The workspace environment may lead to poor results and poor learning gains as compared to an environment better suited to product dissection. It should be noted that the course is relatively new, and the administration is actively working on allocating space to better accommodate the course.

Finally, due to the budget of the course, not all students used the exact same brand or model of hair dryer or power drill. Some differences in the results of the function trees may be due to this difference.

QUANTITATIVE RESULTS

Descriptive statistics were calculated as seen in Qualitative Results. Several examples of the function trees produced are found in Appendix B.

Since a within-subject experimental design was used, a univariate ANCOVA was used with the participant code and the device dissected as blocking factors. Since the time factor is conflated with the device factor, there is no way to separate these two effects. The model for the statistical analysis considered main effects only due to the experimental structure. For this analysis, SAS was used.

Validation of ANOVA Assumptions

ANOVA-family analyses required certain assumptions to be met, or the analysis is significantly weakened. These assumptions are that the data must be normal and satisfy the homogeneity of variance criteria [42].

The dependent variables were tested for normality by comparing their histograms to a normal curve and using a “fat pencil” test as well as checking for linearity on a Q-Q plot. A “fat pencil” test is a common qualitative test used in statistics to determine if the deviation from normality is reasonably small enough to consider the data normal. The number of functions generated and the number of errors committed were both slightly skew-left, though these are qualitatively close enough to be treated as normal distributions. The other dependent variable responses were considered to be normal as assessed by the authors and statistics consultants at Purdue. The effect of a violation of normality is that the statistics will be more likely to detect a statistical difference where there is none [42].

In addition to normality, the variance between variables must be similar to avoid an increased risk of a type-I error [43]. Assuming an alpha level of 0.05, only one parameter violated this assumption, and only two others even came close, as seen in.

ANCOVA Results

Although the study is exploratory, a significance of 0.05 is used to determine significance. The p-values for the independent variable (method) and several covariates are categorized by their effects on the several dependent variables, as seen in Table 6. Effect size is not reported since the sample size (n=78) is less than 100 and statistical significance is not sufficiently affected by n.

There are no significant effects by the method used on any of the measured responses (see Table 6). A few effects from other covariates and blocking factors were found (see Table 8 in Appendix A): there was an effect between a.) the device used and the number of functions recorded, b.) the class level and the levels of abstraction used, c.) prior experience with a particular device and the number of errors committed, and d.) prior experience with a device and the perceived usefulness of an activity to an individual. All other effects from covariates were insignificant.

TABLE 5 - SIGNIFICANT OR NEAR-SIGNIFICANT SAMPLE VARIANCES. OTHER VARIANCES ARE NOT SHOWN. VIOLATING VARIANCES ARE NOT CRITICAL TO THE STUDY, BUT DO INFLUENCE ANY RESULTS FOR THOSE DEPENDENT VARIABLES.

Source (DV vs. Fixed Factors)	DF	F Value	P Value
Device Dissected vs. # Functions Total	2	3.63	0.0352
Taken Apart Device Before vs. Errors	1	3.90	0.0550
Taken Apart Device Before vs. Survey Result	1	3.27	0.0785

TABLE 6 - SIGNIFICANCE OF MAIN EFFECTS ON SEVERAL RESPONSE VARIABLES FROM THE INDEPENDENT AND COVARIATE FACTORS

Effect	Response	DF	F Value	P Value
Method	Unique	2	0.01	0.9879
	# Functions Total	2	0.35	0.7039
	# Errors	2	0.28	0.7541
	Levels of Abstraction	2	0.66	0.5229
	Survey Results	2	1.12	0.336

TABLE 7 - AVERAGE VALUES FOR VARIOUS LEVELS OF FACTORS FOUND SIGNIFICANT IN THE ANOVA ANALYSIS

Average # of Unique Functions by Method	
Top Down	8.96
Energy-Flow	8.92
Unstructured	9.08

Average # of Levels of Abstraction by Class Level	
Freshman	0
Sophomore	3.5
Junior	4.166667
Senior	4.842105

Average # of Functions Total by Device Dissected	
Hair dryer	15.52
Power Drill	11.52
NERF gun	11.36

Average # of Errors by Those with Prior Experience with Device	
Yes	5.333333
No	2.758065

Average Survey Results from Those with Prior Experience	
Yes	5.25
No	6.577586

QUALITATIVE RESULTS

Several examples of the function trees produced are found in Appendix B.

Observations of Students

Participants seemed to particularly struggle with using the energy-flow approach since finding the energy, material, and information flows was not always an obvious task. Additionally, the energy flow approach did not seem to lend itself well to a tree hierarchy, which makes sense given that the method tends to focus on one or two levels of abstraction. Participants using the top-down approach seemed to be content with generalized functions, and did not seem motivated to find deeper functions since they felt that they had the gist of it.

Additionally, most participants seemed to struggle with identifying electrical components or their functions. The electrical functions identified focused around batteries, motors, switches, and wires. The participants seemed to focus on the mechanical aspects of the circuits.

Regardless of the method used, many participants struggled to distinguish between parts and functions, often conflating the two. This occurred despite a strong emphasis on distinguishing between parts and functions in the instructions, suggesting that the participants did not receive or understand the instructions as intended.

While the participants paid attention to the instructions the first time, the second time they did not. For example, during the second session, as the participants were putting the drills back together, one of the instructors asked one student, "Did you make a function tree?" The student replied, "We're supposed to make a function tree?"

By the end of the third round, some participants seemed tired of doing functional decomposition. In a survey taken by the course instructors, participants had a wide spread of feelings toward the utility of learning experience. After the experimental period, the course instructors continued to use function trees with the students, but allowing them to work in groups. They observed that proactive students tended to dominate in their groups [31], but that the students were much more tolerant of the extra work in the group setting.

DISCUSSION

The results of the pilot study showed that functional synthesis in design produces highly variable results. It also showed that novice designers have a difficult time identifying functions. Some of the difficulty came from the ambiguous functionality of a toy, but some seemed to also come from the ill-structured nature of the problem.

The results of the main study suggest that there is no difference between the methods. Hypotheses H_1 – H_4 are rejected because the affected responses do not show any significant effect from the method used. It should be noted that the p-value is high enough in most of these cases to accept the null hypothesis with a minimal chance of a type II error. Only a few covariate effects were significant. These are considered in turn.

The number of levels of abstraction increased with the class level of the participants. This could be due to an increase in maturity, or a deeper understanding of mechanical abstraction due to prior course work.

The number of functions generated decreased with each successive device dissected, as seen in Table 7. As mentioned earlier, the time variable is conflated with the device used in each iteration, and these effects cannot be separated. The average number of functions generated suggests one of two things: that the participants generated fewer functions with time, or that certain products had more functions to identify. It seems more likely that the number of functions generated decreases with time. This may be due to boredom, familiarity with the process, a desire to finish sooner, or other motives.

Having prior experience taking apart a particular device had an interesting effect on how participants viewed the experience as well as how many errors they committed. As would be expected, participants with prior experience did not find the activity to be as useful as those without. Interestingly, the number of errors committed by those who have previously dissected a device was much higher than those who had not done so. This may be due to a lack of focus or overconfidence in those who had previously taken apart the device.

H_4 is also rejected due to no significance being found; however, it is strange to note that so many juniors and seniors reported not having learned functional decomposition before. This is unexpected because all Purdue mechanical engineering students are required to learn it as a part of a sophomore design class. Since the students do not recall learning functional decomposition, it may be true that they did not learn it in a permanent way, in which case the analysis holds. However, this aspect of the analysis may be flawed since there may be undetected effects from prior exposure.

Hypotheses

- **H_0** – There is no difference between any of the methods (**failed to reject)**
- **H_1** – The top-down approach will prove to be more effective at a.) generating the total number of functions, b.) generating the fewest errors, c.) generating the most levels of abstraction, and d.) generating the most unique functions. (**rejected**)
- **H_2** – The energy-flow approach will generate the most unique functions (**rejected**)
- **H_3** – The unstructured approach will generate a.) the fewest functions and b.) the fewest unique functions. (**rejected**)
- **H_4** – Older students will perform better than younger students. (**rejected**)

These results imply that there may be no difference among abstraction methods for product dissection among novice engineers. This, of course, cannot be conclusive since a null hypothesis cannot be accepted. This result may be reasonable, though, since there was also no practical difference found between the methods. Follow up studies or a meta-analysis of prior studies should be conducted to support this claim.

This result may be due to failures in the testing procedure. It is possible that the complexity of the product was not sufficient for any of the methods to have any significant impact. It is also possible that not enough engineers were sampled. Also, since the sample was semi-random, the sampling method may be too non-random to hold any validity. If it is result true that the methods are no different, there may be a few reasons for it. First, abstraction by a particular method or general mechanical knowledge may be an expert skill that must be acquired with time. Second, the cognitive load of abstraction methods may be so high for novice engineers that there may be a sort of saturation limit for how many functions they can handle at once. Third, the limited number of functions may be due to mental-set fixation.

If a particular abstraction method is an expert skill, this would correspond with the findings by Eckert, et al., [28] that engineering professionals with more experience tend to use top-down or energy-flow analyses whereas newer professionals tend to use an unstructured approach. From this perspective, long-time professionals should have better mechanical abstraction skills than novices. While indirectly related, studies of fixation tend to show that professionals are also often blocked from abstraction [27]. Any relationship between abstraction skill level and professional level must be controlled for fixation effects. Similar to lacking expert skills, the participants may have also lacked expert knowledge; namely a general knowledge of mechanical and electrical parts. This would seem evident by the inability of most participants to even name most of the electrical elements in the systems.

If various abstraction methods demand a large amount of cognitive resources, this may fit with findings by Gero, et al., [44] which found that certain methods were better for ideation due to a lower cognitive load and an increased amount of time spent on problem definition. This perspective would also explain the observed difference between experts and novice engineers [28]. If there were an abstraction method that demanded less of a cognitive load, it should result in more unique functions identified by novice students.

If the participants were limited due to mental-set fixation, this would explain the same phenomena as the other two hypotheses, but would also be easily tested by removing a participant from the situation and returning them to the task after a period to allow the mind to change cognitive modes. If this hypothesis is true, the participant should be able to identify more unique functions after the first iteration, and possibly have fewer repetitive functions than a participant without such an intervention.

IMPLICATIONS

One of the most important implications of this study is that the common functional decomposition methods may not even work for reverse engineering. While functional decomposition methods may work well for design methods (though this is also untested); they do not appear to be effective for product dissection. The results of this study could have significant impact on the design abstraction community. Accordingly, more testing should be done to validate these results.

When reverse engineering, it was also seen that novices tend to fixate on the name of the part, rather than actually determining its function or meaning within the entire system. Identifying the function, or symbolic meaning, of each part or subsystem is naturally built into approaches such as the top-down or energy-flow approach, but these rely on existing mechanical knowledge and symbolic schema. Novices probably fail at these methods because they do not already understand the deeper meanings of each individual part, and thus cannot understand the higher-order meanings of their sub-systems and systems.

The pilot study further suggests that as design projects become more human-centered, exclusively mechanical functions will increasingly fail to meet the needs of new designers. There is a significant need to develop a function taxonomy that also supports psychological and social functions of a device. This type of taxonomy will be best served if developed from an interdisciplinary approach where expertise in social and psychological fields can be employed to inform the taxonomy.

These results seem to match other observations that in practice, designers rarely use functional decomposition during the ideation phase [45].

Based on the qualitative observations of this study, a new method may be needed to teach students how something works. Such a new method should encourage natural discovery processes and not rely on extensive mechanical knowledge or other expert knowledge. Children often explore new physical phenomena by conducting a series of micro-experiments to test the boundaries and constraints of a system [36]. For example, when presented with a new, exotic toy, children will see how each piece moves, and gradually build an understanding of the functionality of a toy. Additionally, a similar bottom-up process was implemented on a middle-school level, though this method was not strictly formal [23]

A Proposed Function Identification Method for Product Dissection

Instead of a top-down approach which would favor someone who already has a mechanical schema, a bottom-up approach may be more efficient in teaching students how to determine the functionality of a device. A bottom-up approach would start with a list of parts and a diagram of where they are located in the device. The participants should then determine the meaning or function of each part within the system. If a student can play with the actual device and see how it moves and interacts with other parts, the participant can then experiment with various meanings until they create their own. Then they should determine the interaction of each part with other parts to start to form clusters of functionality, or subsystems. This should continue until every part is linked into the whole. A proposed order of steps for this activity includes:

1. Determine what each part does
2. Determine how each part interacts with its neighbors
3. Cluster parts based on interactions with each other
4. Determine what each cluster does
5. Determine how each cluster interacts with its neighbors
6. Continue until all clusters and super-clusters are joined
7. Name functions for each part, cluster, and super-clusters.

This approach would better support abstraction because symbolic meanings would be developed explicitly before they must be arranged into a hierarchy. It allows for parts to have multiple functions, unlike a top-down approach, as seen in the pilot study. It also enables hierarchies to be readily formed, unlike the energy-flow approach, as seen in the experimental results. This also bridges the gap between the part tree and the function tree that seems to confuse students. Finally, this approach seems to best approximate how children and students play with a machine and each part until they determine how it works, thus mimicking natural mechanical investigations.

Finally, the method for recording function hierarchies can vary, but due to the similarities between sketching cognition and abstraction cognition, some sort of sketching activity should be included. This happened spontaneously in the study by Eckert et al. [3], emphasizing the utility of sketching as a bridge between the symbolic meanings and the organized depiction of a hierarchy. Thus, this natural inclination should be capitalized into a naturalistic form of recording functionality.

CONCLUSIONS

The results of this study suggest that formal functional decomposition methods do not work for novice engineers performing reverse engineering or product dissection. This may be due to the cognitive load, lack of expert knowledge, or mental set fixation.

Additionally, as no studies have directly compared the effectiveness of whole methods, there is little evidence that current functional decomposition methods even work or significantly impact the design process. From a psychological point of view, it is clear that designers need abstraction, but how to best implement this is still elusive.

Abstraction is an essential design element that is understudied [46]. Design studies in this field have largely focused on proposing new methods, new taxonomies, or fixation effects, and only initial steps have been made into creating a theoretical cognitive model to understand how abstraction operates and enhances design. Abstraction and fixation studies have not uniformly distinguished between problem types (i.e. well-defined vs. ill-structured) or fixation types (i.e. functional vs. mental set), creating an incoherent background on the topic. Consequently, design research in abstraction is often far behind industry practice where practicality forges its own methods [47].There are some limitations to the study. Although a within-subject design was used, the sample size is still relatively small. Additionally, the sample was a convenience sample, and no measures were taken to characterize differences between the sample and the general engineering population. The session sizes were not uniform, and this could lead to some effect on the results, though this is minimized by the within-subject design. The order of products was conflated with time, making it impossible to separate time effects or effects due to the product. This; however, allowed a much simpler design to be used. To separate the time and product variables, a much larger study would need to be done, either as a between-subject study with a large sample size, or as a full-factorial design; both of which would be quite unwieldy. The number of errors and the number of unique functions were determined using only one judge, and a more objective or more rigorous method for developing these data would improve fidelity of the results. It is also possible that using different products or different levels of complexity would yield different results.

There are several steps that should be taken to deepen our understanding of design abstraction. A meta-analysis of past abstraction studies would greatly clarify the field. Research should also start with validating current methods, exploring methods used in industry, and developing methods that effectively aid novice designers. A generalizable cognitive model of design abstraction is needed if functional decomposition methods are to be improved. Grounded abstraction methods should be proposed and tested, and ultimately, the principles behind abstraction should be discovered. These principles can then be applied to many design situations and potentially improve the entire design process.

ACKNOWLEDGMENTS

The authors would like to thank Katherine Frangos and the instructors of ME 290 for their cooperation and help and the participants who contributed their time and efforts. The authors would also like to thank Senthil K. Chandrasegaran and others for their help in executing the experiment and in providing feedback on the draft.

REFERENCES

[1] George E. Dieter, *Engineering Design: A Materials and Processing Approach*, Kevin T. Kane, Ed.: McGraw-Hill, 2000.

[2] Barry Hyman, *Fundamentals of Engineering Design*.: Prentice Hall, 2003.

[3] Claudia Eckert, Thomas Alink, Anne Ruckpaul, and Albert Albers, "Different notions of function: results from an experiment on the analysis of an existing product," *Journal of Engineering Design*, vol. 22, no. 11-12, pp. 811-837, 2011.

[4] Vinod Goel and Peter Pirolli, "The Structure of Design Problems Spaces," *Cognitive Science*, vol. 16, pp. 395-429, 1992.

[5] Eric Brewe, "Modeling Theory Applied: Modeling instruction in introductory physics," *American Journal of Physics*, vol. 76, no. 12, pp. 1155-1160, 2008.

[6] Stephen P. Hoover, James R. Rinderle, and Susan Finger, "Models and abstractions in design," *Design Studies*, vol. 12, no. 4, pp. 237-245, 1991.

[7] Henry W. Stoll, *Product Design Methods and Practices*.: Marcel Dekker, Inc., 1999.

[8] Nancy J. Stone, "Windows and Environmental Cues on Performance and Mood," *Environment and Behavior*, vol. 30, no. 3, pp. 306-321, 1998.

[9] David G. Ullman, *The Mechanical Design Process*, 3rd ed., Bill Stenquist, Ed.: McGraw-Hill, 2003.

[10] Patrick F. Cunniff, Jeffery W. Herrmann, LInda C. Schmidt, Guangming Zhang, and James W. Dally,.: College House Enterprises, LLC, 1998, ch. Chapter 5: Functional Decomposition, pp. 81-93.

[11] David H. Jonassen, "Toward a Design Theory of Problem Solving," *Educational Technology: Research and Development*, vol. 48, no. 4, pp. 63-85, 2000.

[12] Bryan Lawson and Kees Dorst,.: Architectural Press, 2009, ch. 2: Understanding Design, pp. 23-50.

[13] David G. Jansson and Steven M. Smith, "Design fixation," *Design Studies*, vol. 12, no. 1, pp. 3-11, 1991.

[14] National Research Council, *How People Learn: Brain, Mind, Experience, and School*, John D Bransford, Ann L Brown, and Rodney R Cocking, Eds.: National Academy Press, 2000.

[15] Jonathan Fish and Stephen Scrivener, "Amplifying the MInd's Eye: Sketching and Visual Cognition," *Leonardo*,

vol. 23, no. 1, pp. 117-126, 1990.

[16] Hugh Dubberly, Shelley Evenson, and Rick Robinson, "On modeling: The analysis-systhesis bridge model," *interactions*, vol. 15, no. 2, pp. 57-61, 2008.

[17] G. Pahl and W. Beitz, *Engineering Design: A Systematic Approach*.: Springer, 1996.

[18] Clive L. Dym and Patrick Little, *Engineering Design: A Project-Based Introduction*.: John Wiley \& Sons, 2000.

[19] Karl T. Ulrich and Steven D. Eppinger, *Product Design and Development*.: McGraw-Hill, 2000.

[20] Nigel Cross, *Engineering Design Methods*.: John Wiley \& Sons, 2000.

[21] A.M. Phillips, "Functional decomposition in a vehicle control system," , vol. 5, 2002, pp. 3713 - 3718 vol.5.

[22] Mark N. Horenstein, *Design Concepts for Engineers*, Marcia J. Horton, Ed.: Prentice Hall, 2002.

[23] Sheri D Sheppard, "Mechanical Dissection: An Experience in How Things Work," *American Society for Engineering*, 1992.

[24] Michael French, *Conceptual Design for Engineers*.: Springer, 1999.

[25] Edward B. Magrab, *Integrated Product and Process Design and Development: The Product Realization process*.: CRC Press, 1997.

[26] John W. Priest and José M. Sánchez, *Product Development and Design for Manufacturing: A Collaborative Approach to Productibility and Reliability*, 2nd ed.: Marcel Dekker, Inc., 2001.

[27] A.T. Purcell and J.S. Gero, "Drawings and the design process: A review of protocol studies in design and other disciplines and related research in cognitive psychology," *Design Studies*, vol. 19, no. 4, pp. 389-430, 1998.

[28] Claudia Eckert, Anne Ruckpaul, Thomas Alink, and Albert Albers, "Variations in Functional Decomposition for an exisiting product: Experimental Results," *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing*, vol. 26, pp. 107-128, 2012.

[29] Kevin Otto and Kristin Wood, *Product Design: Techniques in Reverse Engineering and New Product Development*. Upper Saddle River, NJ: Prentice Hall, 2001.

[30] Christine Toh, Scarlett Miller, and Gül Kremer, "Mitigating design fixation effects in engineering design through product dissection activities," , 2012.

[31] Christine Toh, Scarlett Miller, and Gül E Okudan, "The Impact of Product Dissection Activities on the Novelty of Design Outcomes," , 2012.

[32] B. S. Bloom, *A Taxonomy of Educational Objectives Handbook I : Cognitive Domain*. New York: McKay, 1965.

[33] Kurt W Fischer and Susie D Lamborn, "Mechanisms of variation in developmental levels: cognitive and emtional transition during adolescence," in *Transition Mechanisms in Child Development: The Longitudinal Perspective*, Anik De Ribaupierre, Ed. New York: Press Syndicate of the University of Cambridge, 1989.

[34] Seymour Papert, *Mindstorms: Children, Computers, and Powerful Ideas*.: Basic Books, 1980.

[35] E Gillett, "Relativism and the Social-constructivist Paradigm," *Philosophy, Psychiatry, & Psychology*, vol. 5, no. 1, pp. 37-48, 1998.

[36] Philip M Sadler, Harold P Coyle, and Marc Schwartz, "Engineering Competitions in the MIddle School Classroom: Key Elements in Developing Effective Design Challenges," *The Journal of the Learning Sciences*, vol. 9, no. 3, pp. 299-327, 2000.

[37] Kees Dorst, "The Problem of Design Problems," *Journal of Design Research*, vol. 4, no. 2, 2004.

[38] Benjamin W Caldwell, Jonathan E Thomas, Chiradeep Sen, Gregory M Mocko, and Joshua D Summers, "The Effects of Language and Pruning on Function Structure Interpretability," *Journal of Mechanical Design*, vol. 134, pp. 1-11, 2012.

[39] Barry M Kudrowitz and David R Wallace, "The Play Pyramid : A Play Classification and Ideation Tool for Toy Design," *International Journal of Arts and Technology*, vol. 3, no. 1, pp. 36-56, 2010.

[40] Howard J Seltman, Experimental Design and Analysis, 2009.

[41] T. R. V. Davis, "The influence of the physical environment in offices," *Academy of Managment Review*, vol. 9, pp. 271-283, 1984.

[42] Carl J Huberty et al., "Statistical Practices of Educational Researchers: An Analysis of Their ANOVA, MANOVA and ANCOVA Analyses," *Review of Educational Research*, vol. 68, no. 3, pp. 350-386, 1998.

[43] J. W. Mauchly, "Significance Test for Sphericity of a Normal n-Variate Distribution," *The Annals of Mathematical Statistics*, vol. 11, no. 2, pp. 204-209, 1940.

[44] John S. Gero, Hao Jiang, and Christopher B. Williams, "Does using different concept generation techniques change the design cogition of design students?," , 2012.

[45] Benjamin W Caldwell, Raveesh Ramachandran, and Gregory M Mocko, "Assessing the Use of Function Models and Interaction Models Through Concept Sketching," , 2012.

[46] Andreea Danielescu, Mahmoud Dinar, Christopher MacLellan, Jami J Shah, and Pat Langley, "The Structure of Creative Design: What Problem Maps Can Tell Us About Problem Formulation and Creative Designers," , 2012.

[47] Marco Aurisicchio, Rob Bracewell, and Gareth Armstrong, "The Function Analysis Diagram," , 2012.

APPENDIX A – ADDITIONAL STATISTICS

TABLE 8 - ADDITIONAL MAIN EFFECTS FROM ANOVA ANALYSIS

Effect	Response	DF	F Value	p Value
Device Used / Week Studied	Unique	2	0.99	0.3787
	# Functions Total	2	4.95	0.0117
	# Errors	2	0.15	0.8573
	Levels of Abstraction	2	0.77	0.4679
	Survey Results	2	0.79	0.4602

Effect	Response	DF	F Value	P Value
Class Level	Unique	1	0.28	0.601
	# Functions Total	1	0.16	0.6916
	# Errors	1	1.81	0.1861
	Levels of Abstraction	1	7.51	0.009
	Survey Results	1	0.29	0.5944

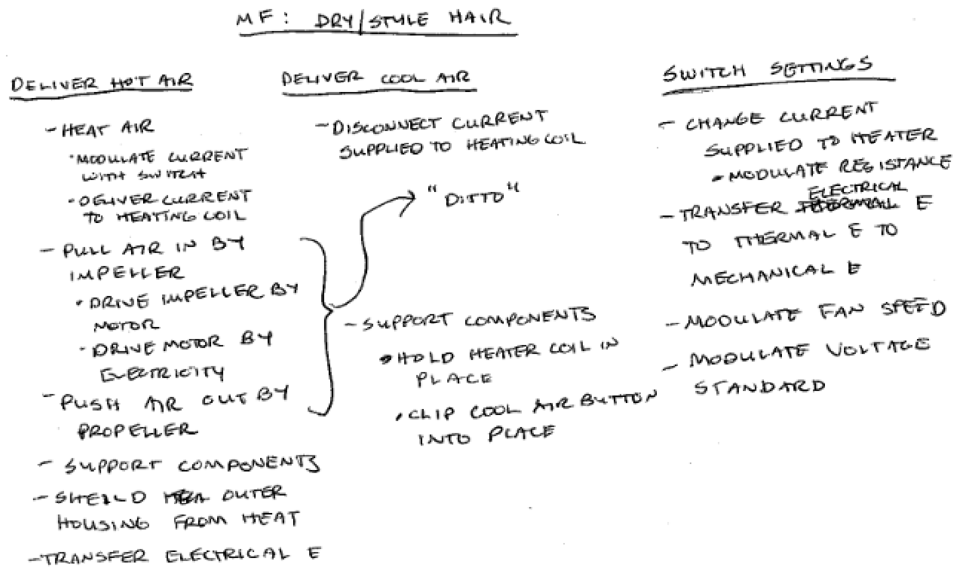
Effect	Response	DF	F Value	P Value
Frequency Participant Takes Things Apart on Own	Unique	1	0.03	0.8641
	# Functions Total	1	0	0.9926
	# Errors	1	1.27	0.2665
	Levels of Abstraction	1	0	0.9923
	Survey Results	1	1.96	0.1701

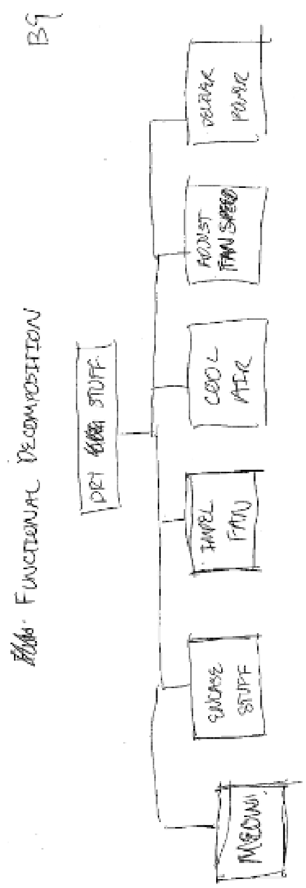
Effect	Response	DF	F Value	P Value
Has Participant Taken Apart Device Before?	Unique	1	0.89	0.3522
	# Functions Total	1	1.46	0.233
	# Errors	1	4.6	0.0377
	Levels of Abstraction	1	0.23	0.6375
	Survey Results	1	5.05	0.0304

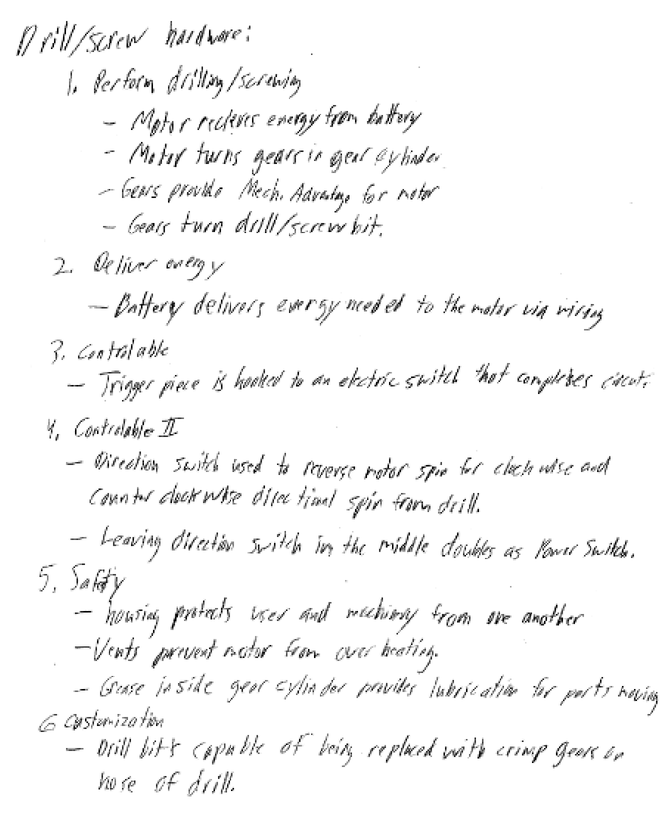
Effect	Response	DF	F Value	P Value
Has Participant Learned Functional Decomp. Before?	Unique	1	0.03	0.854
	# Functions Total	1	0.34	0.5634
	# Errors	1	0	0.9475
	Levels of Abstraction	1	2.21	0.1448
	Survey Results	1	1.77	0.1911

Effect	Response	DF	F Value	P Value
Did the participant use a draft?	Unique	1	0.75	0.3921
	# Functions Total	1	0.49	0.4898
	# Errors	1	1.04	0.3139
	Levels of Abstraction	1	0.73	0.3966
	Survey Results	1	0	0.9893

APPENDIX B – EXAMPLES OF FUNCTION TREES PRODUCED BY PARTICIPANTS

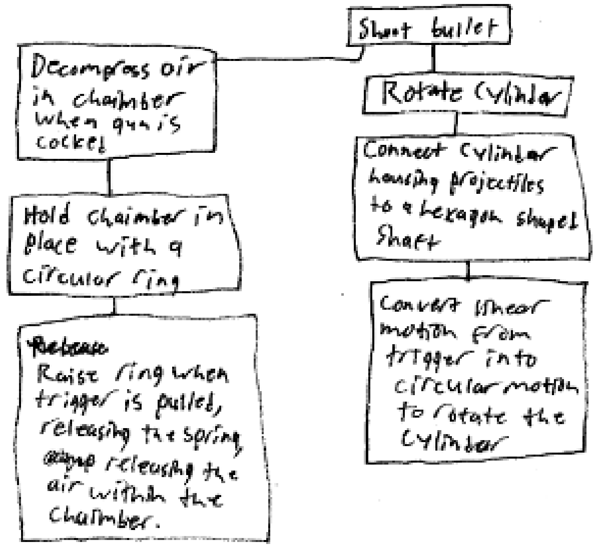






Nerf Gun

- Function - fire dart
- fire ~~gu~~ dart in single direction
- store energy in ~~air~~ spring
- force air through the piston into the chamber
- minimize part wear
- release the air using trigger
- spin the chamber after each shot
-



A4