International Journal of Engineering Education Vol. 34, No. 2(B), pp. 567–573, 2018
Printed in Great Britain
0949-149X/91 $3.00+0.00
© 2018 TEMPUS Publications.

Prototyping and the Engineer of 2020*

REID BAILEY
University of Virginia, 151 Engineer's Way, Charlottesville, VA 22904, USA. E-mail: rrbailey@virginia.edu

MATTHEW E. MCFARLAND
Alarm.com, Tysons, VA 22102, USA. E-mail: mcfarlandme@gmail.com

The central thesis of this paper is that prototyping abilities are important for engineers in 2020 even though prototyping is not prominent in the Engineer of 2020 reports. While design itself is not cited as a separate attribute in the Engineer of 2020, its role is clear through several attributes such as "practical ingenuity" and "creativity." Prototyping and making skills, on the other hand, are not even hinted at as an important skill of the engineer of 2020. We assert that one reason for the lack of focus on prototyping is the context and timing of the report, which was published in 2004. Not only was the report written as engineering education continued its movement away from the hands-on and towards the mathematical and theory-based, it was also written when a technological revolution that made prototyping quicker and more accessible was only in its infancy. In this paper, we present results from a study of how engineering students design. Thirty-six undergraduate engineering seniors worked on teams of four for three hours to design a product while being video recorded. The students were provided with a space to work, a whiteboard, and prototyping supplies ranging from office supplies to more technological items such as sensors. One of the most surprising results was that teams spent over half (56%) of their time either building or testing prototypes. Working to better understand the problem and doing conceptual design work only accounted for 44% of their time. While prototyping can be used for many reasons, teams in this study used prototyping as a primary means of evaluating performance of designs. This activity—evaluating performance—is linked only to analytical skills in the Engineer of 2020 report. As we look to the engineer of 2040, we should embrace the idea that hands-on prototyping also plays a critical role.

Keywords: prototyping; engineering design; verbal protocol analysis; Engineer of 2020

1. Motivation: Prototyping abilities are important for engineers of 2020

In engineering practice, engineers are expected to integrate prototyping into their design process. The most highly innovative firms that do engineering design, such as DEKA Research and Development and IDEO, incorporate prototyping into design, encourage failure as a path to quick success [1], and live by mantras such as "never go to a meeting without a prototype" [2]. The DARPA Grand Challenges have successfully advanced autonomous vehicle design through having dozens of teams make different prototypes [3]. And there are extreme examples—outliers of prototyping use—such as Wikispeed, the group that developed a 100 mpg car in 3 months using a prototyping-centric design process [4].

In this context, we present the results of a verbal protocol analysis (VPA) study in which design teams were given not only all of the normal things given to subjects in a VPA study (i.e., a design prompt, a white board, the opportunity to ask for additional materials, etc.), but also materials for prototyping. The objective is to describe how prototyping was integrated into and influenced the design process. The hypothesis of the study is that prototyping will be used extensively by teams.

2. Prior work and context: The Engineer of 2020 reports place little emphasis on the role of prototyping in engineering design

2.1 Prototyping as "realization" has many roles in design, with studies showing that early prototyping leads to better design performance.

There are many different definitions of "prototype." Ulrich and Eppinger purposefully include concept sketches, mathematical models, and implementations of an idea in their definition [5]. Tim Brown focuses more on physical prototyping by describing it as "the willingness to go ahead and try something by building it" [6]. The definition we most closely adhere to is that of Tom and David Kelley in *Creative Confidence*, namely that a prototype is an "embodiment of your idea" [2]. Examples by Kelley and Kelley make it clear that they are not limiting prototypes to designs that can be physically "built" (e.g., it could be Post-it notes to simulate a user interface) nor are they extending prototypes to include mathematical analysis. As used in this study, prototyping involves *realization* of an idea that goes beyond a sketch or an analytical representation of the idea. For a physical design, this would involve physical making. For software, this would involve coding. For a service, prototyping could involve a range of techniques

* Accepted 12 January 2018.

that bring a conceptual idea for that service to reality.

Prototyping, in the context of engineering, is an activity done during design. Design, too, is defined many different ways. Perhaps one of the most compelling, due in part to its simplicity, is Sheppard's definition that, in designing, engineers "scope, generate, evaluate, and realize ideas" [7]. The Accreditation Board of Engineering and Technology defines design as "the process of devising a system, component, or process to meet desired needs" [8]. Dym, et al., assert that "engineering design is a systematic, intelligent process in which designers generate, evaluate, and specify concepts for devices, systems, or processes whose form and function achieve clients' objectives or users' needs while satisfying a specified set of constraints" while going on to add that a "designer has a client (or customer) who, in turn, has in mind a set of users (or customers) for whose benefit the designed artifact is being developed" [9]. Pahl and Beitz divide design into four majors stages: clarifying the task, conceptual design, embodiment design, and detail design [10].

Prototyping can have many different roles in design. At its simplest, prototyping is a means to "realize" a design (referring to Sheppard's definition of design cited in the prior paragraph). Ulrich and Eppinger include an entire chapter about how prototyping can be used in design, with their discussion of realization-focused prototypes addressing the use of such prototypes later in design as a team navigates progressively more refined versions of a design [5]. Otto and Wood provide an extensive treatment of prototyping's role in engineering design, including a broad range of uses of prototypes that may occur earlier in a design process (e.g., proof of concept) or later (e.g., alpha and beta prototypes) [11]. Numerous studies have explored the relationship between prototypes and design performance, including Youman's work shows that working in a physical prototyping environment leads to less design fixation and better designs [12], Jang and Schunn's work which demonstrates that consistent use or prototypes throughout a design process led to greater team success [13], and Yang and Epstein's work showing that simpler prototypes resulted in better design outcomes [14].

2.2 The Engineer of 2020 report is tacit on the importance of prototyping

Design itself is not listed as a separate attribute of the Engineer of 2020, instead being integrated into the other attributes such as "practical ingenuity," "creativity," and "strong analytical skills." The report does make it clear that design is an important overarching part of engineering: "At its core, engineering employs principles of science, mathematics, and domains of discovery and design to a particular challenge and for a practical purpose" [15].

In contrast with design, prototyping and realization are never directly or indirectly referred to as an attribute of an engineer in *Engineer of 2020*. In *Educating the Engineer of 2020*, the only reference to prototyping is in Recommendation 4, where "building" is used to refer to a dimension of prototyping [16]: "[the essence of engineering is] the iterative process of designing, predicting performance, building, and testing".

2.3 The context in which the Engineer of 2020 reports were written was not one in which prototyping was a valued engineering skill.

A historical context that devalued hands-on abilities of engineers and the only recent rise of low-cost, easy-to-use electrical and physical prototyping platforms may explain part of why prototyping is not more prominent in the Engineer of 2020 reports. The history of engineering education is a well-documented journey away from the hands-on shop and towards theoretical modeling [17, 18]. It is from this context that the Engineer of 2020 was written, where a vision that elevated the role of hands-on skills could be viewed as moving backwards.

Equally important to what had come before is what was yet to come when the report was written: namely the democratization of making and prototyping. Accessible, low-cost electronics prototyping was in its infancy, as marked by the Arduino platform having origins dating to 2004, Sparkfun and Adafruit opening doors in 2003 and 2005 respectively, and Make magazine being founded in 2005. Additionally, the promises of "rapid prototyping" (now called additive manufacturing) as a low-cost tool were not realized in engineering education until roughly 2009, when hobbyist-level fused deposition modeling "3d printers" such as the Makerbot Cupcake CNC were released. Physical prototyping has continued to advance as such 3d printers, desktop CNC machines, and lasercutters have become easier and less expensive to use.

The attributes in the Engineer of 2020 were shaped by these two forces—the devaluing of prototyping, making, and realization within engineering and the lack of quick, low-cost approaches for making prototypes. These two forces have changed since the writing of the Engineer of 2020. The rise of firms that integrate prototyping heavily into their approach as being leaders in innovation has begun to change the view that prototyping is "beneath" the skillset of an engineer. This can be seen by the influence of "design thinking," an approach that focuses on, among other things, quick iteration through the use of prototypes [2, 6]. And, as already outlined, quick, low-cost prototyping technologies are available today that were not in 2004; technologies that facilitate the use of prototypes earlier in a design process to explore multiple ideas.

3. Methodology: A 3-hour verbal protocol analysis study with 9 teams of engineering students

Verbal Protocol Analysis has been used extensively to study design behaviors; that said, it has not been used extensively to study prototyping. One of the most intriguing examples of this is documented in Analysing Design Activity [19]. In this set of studies, nineteen groups of researchers all analyzed the same videos of a design team and an individual designing a way to attach a backpack to a bike rack. Every researcher could choose what dimension to study and which methods to use to analyze the videos. The nineteen teams of researchers gathered at Delft University in 1994 to present their findings. The subjects were not given prototyping materials, but were given a bike, a backpack, a tape measurer, and a pad and paper. Even without a wealth of prototyping materials, the workshop organizers noted that the designers used "design objects" intensively. In one case, the design team used the tape measurer to prototype a variable length support. Despite this engagement of the subjects in prototyping, *only one of the nineteen independent analyses had any significant focus on embodiments and implementations associated with prototyping*. The subjects were given no materials with which to prototype. . . they prototyped anyway. . . and almost no one noticed.

3.1 Experimental setup and subjects

Thirty-six students grouped into nine teams, each with two systems engineering (SYS) and two electrical/ computer engineering (ECE) students, completed a 3-hour Verbal Protocol Analysis (VPA) study in which they were tasked with designing a device to count and record the time when student newspapers were taken from a distribution box [20]. In addition to a newspaper distribution box with newspapers, a whiteboard, paper, and pens, each team had access to various construction materials (tape, cardboard, scissors, paper, etc.) and several electronic sensors manufactured by Phidgets and SunSPOT. Students were also provided with four laptop computers outfitted with Microsoft Office and Integrated Development Environments to configure the electrical sensors. Subjects received $100 for their participation. The study was approved by the Institutional Review Board.

3.2 Transcript preparation

Summarized transcripts of each team were coded for team structure and design process. To create these transcripts, segment breaks were inserted when the design stage the team was functioning in changed or when there was a change in team structure (working as a single group versus in subgroups). To check the validity of the summaries written, each transcript was reviewed by a second reviewer for feedback on the following areas: the accuracy of the summary (i.e., what happened, who said it, is there context missing, etc.), the accuracy of the time recorded (within a few seconds), and the length of the segments (i.e., if subsequent statements should be combined or if long segments should be split). Following this review, a second review of the videos and transcripts was completed by the principle researcher to incorporate feedback from the second reviewer.

As a final step to transcript preparation, before applying the coding scheme, three percent of segments were selected at random to be jointly coded by the principle researcher and a second coder. After discussing the challenges identified from the second coder review, the principal researcher completed another review of all transcripts.

3.3 Coding

All segments were combined into one file and decontextualized through order randomization before applying the coding scheme. Two coding schemes were applied: one focused on the structure of the team and the other on design stage. For team structure, we coded if the team was working all together or in different subteams.

For design stage, we started by investigating the coding scheme developed by Moore, Atman, Bursic, Shuman, and Gottfried [21]. We adapted this scheme to account for prototyping and testing of a realized product and to aggregate several categories into less granular, more generalized stages of design. Figure 1 shows the adapted coding scheme.

This framework is divided into two dimensions—conceptual versus realization and diverge versus converge—which determine the location of the stages in the diagram. First, each stage is characterized by whether the actions and conversations that occur within it are either abstract/generalized thinking of the mind (conceptual) or as practical implementation of thoughts and ideas (realization) [22]. Each stage is also characterized as either a creative act (diverge) or an evaluative act (converge). Divergence is associated with activities like brainstorming, ideation, building, and prototyping. Convergence is associated with activities such as analysis, selection, evaluation, and testing.

Table 1. Coding Notation Used for Team Structure

Code	Explanation of Team Structure			
4s	All 4 students in a single group.			
31e	2 SYS and 1 ECE students in a subgroup.			1 ECE student alone.
31s	1 SYS and 2 ECE students in a subgroup.			1 SYS student alone.
22m	1 SYS and 1 ECE students in a subgroup.		1 SYS and 1 ECE students in a subgroup.	
22s	2 SYS students in a subgroup.		2 ECE students in a subgroup.	
211e	2 ECE students in a subgroup.		1 SYS student alone.	1 SYS student alone.
211m	1 SYS and 1 ECE students in a subgroup.		1 SYS student alone.	1 ECE student alone.
211s	2 SYS students in a subgroup.		1 ECE student alone.	1 ECE student alone.
1111s	1 SYS student alone.	1 SYS student alone.	1 ECE student alone.	1 ECE student alone.

SYS = systems engineering; ECE = electrical/computer engineering.

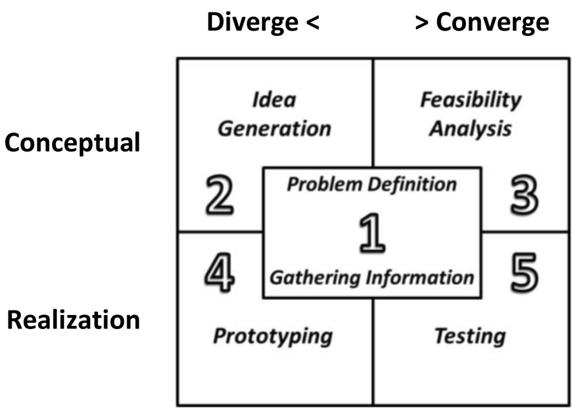

Fig. 1. Engineering Design Stages Coding Scheme.

Altogether, five stages comprise this framework. Stage 1 focuses on conversations or actions pertaining to defining requirements, project scoping, and gathering information about a particular project or the needs of stakeholders. Stage 2 focuses on conceptual conversations about new ideas for solutions or designs that pertain to the prototype, including brainstorming and other forms of idea generation (which could be applied prior to any implementation or in response to testing or implementation problems). Stage 3 focuses on conceptual conversations about the feasibility of a proposed solution, including analysis, evaluation, simulations, and multi-attribute selection of a concept. Stage 4 focuses on actions and conversations associated with the constructing of a prototype including building and software coding. Stage 5 focuses on actions or conversations associated with the testing of an implemented system or prototype.

Inter-rater reliability for design stage coding was established by having 20% of the overall number of segments coded by a second coder. Three percent of those statements were coded jointly with the second coder as described in Section 3.2. The remaining 17% were coded independently and inter-rater agreement measured by computing Cohen's kappa coefficient. An agreement was defined as both coders assigning the same code to a transcript segment. The Cohen's kappa coefficient measured after independently coding the remaining 17% was 0.81, which is considered substantial to almost perfect agreement [23]. Given these results, the remaining 80% of the segments were coded by the principal coder.

4. Results: Prototyping dominated the teams' attention and strongly influenced team structure

The prominent role of prototyping in influencing design activities and teamwork is clear in this study: no activity is more prominent based on time spent. As can be seen in Table 2, teams spent 41% of their time creating prototypes and 56% of their time creating or testing prototypes. Prototyping mainly consisted of hardware preparation/mounting and coding to control the sensors. Teams normally investigated several prototypes as a mechanism to flesh out the design and inform decisions.

Teams split up more to realize prototypes (Stage 4) than they did for any other design stage. As shown in Table 3, when creating prototypes, teams were working all four together only 19% of the time (and were split up 81% of the time). This is in contrast with problem formulation, generating ideas, evaluating ideas, and testing prototypes, where teams spent over 40% of the time working all four together (chi-square = 245, dF = 4, $p < 0.001$). The reasons team divided during prototyping appeared to be because different people on the teams knew how to do different things (e.g., write code, prototype with hardware) well and, in some cases, the scale of the prototypes did not permit more than two people to be working on

Table 2. Percent Time Spent in Different Design Stages and Phases (*rounding leads to totals for Design Phase of 99%*)

	Design Stage	% Time
1	Problem Formulation	14%
2	Conceptual: Idea Generation	11%
3	Conceptual: Feasibility	4%
4	Realization: Prototyping	41%
5	Realization: Testing	11%
Mixed	More than one stage	11%
N/A	Activities not mapped to any design stage	8%

	Design Phase	% Time
1	Problem Formulation	14%
2 or 3	Conceptual: Idea Generate and Feasibility	20%
4 or 5	Realization: Prototyping and Testing	56%
Other Mixed	Mixed not within an above pairing	1%
N/A	Activities not mapped to any design stage	8%

Table 3. Time Spent Working All Four Together in Each Design Stage

For example, 42% of the time spent in Problem Formulation was spent with all four team members working together ("4s" team structure from Table 1)

	Design Stage	% Time Working All Four Together
1	Problem Formulation	42%
2	Conceptual: Idea Generation	43%
3	Conceptual: Feasibility	78%
4	Realization: Prototyping	19%
5	Realization: Testing	47%

any one part (e.g., only one or two people could sit around a computer to write code). While the scale part may not transfer to other projects, the specialization reason could more easily transfer.

5. Discussion: While this is just one study using one design prompt, the finding that prototyping can be an important design activity for many projects extends beyond this case.

A central thesis of this paper is that prototyping abilities are important for engineers in 2020 even though prototyping is not prominent in the Engineer of 2020 reports. These results support the view that prototyping is a central activity in engineering design—at least with respect to the amount of time engineers spend prototyping and testing prototypes.

5.1 Prototyping was used to advance designs through evaluation

The primary metric used to assess the importance of different design activities in this paper is the amount of time spent on each activity. While we acknowledge that this is not an all-encompassing metric (e.g., perhaps there are less time-intensive activities that make larger impacts), the very fact that engineers spend significant amount of times realizing ideas infers some level of importance. When viewed in light of the literature in Section 2.1 that shows the positive impact of prototyping on product performance, it is indeed good that engineers spend significant time realizing their ideas.

Prototyping and testing were being used as an activity from which to learn and then advance a design, not a final activity only performed after significant analysis and vetting of a concept was complete. In fact, prototypes were essentially the only technique used to determine if an idea would work. Teams needed to answer questions such as the following, and analysis alone could not provide sufficient answers.

- Is the force sensor precise enough to measure when one newspaper is removed?
- Can we mount the button in a place that is not in the way while also measuring each time the door of the distribution box is open?
- Can a distance sensor detect when one paper is removed by the reduction in height of the newspaper stack?
- Can an RFID sensor detect when the distribution box is opened?
- How many false positives might we get? False negatives?

The teams were given sample code that would provide readings from the sensors, so basic connectivity between the sensors and the computer was not a major use of time. Instead, software coding prototypes focused more on exploring the sensitivity of the different sensors and customizing the code to do things like count each time a sensor was tripped and define thresholds for what counts as a sensor being "tripped."

Hardware prototyping focused on building mounts for sensors and determining sensor positioning. In both hardware and software, it was common to see teams implement a new solution and then test it for a few minutes. For example, teams using a force sensor underneath the stack of newspapers would get the sensor working and then test it by adding and removing newspapers to the stack while monitoring the sensor readings. Such testing would drive a modification of the design (or a change to a new concept) and re-testing to look for improved performance.

When compared to the other design stages coded in this study, prototyping and testing provided high value for rapid advancement of solutions. For Problem Formulation, teams read two pages about the problem, asked for extra information as appropriate, interacted with the newspaper distribution box, and synthesized all of that into a clear understanding of user needs/requirements. Idea Generation involved open discussions and creation of concepts. Feasibility Analysis involved teams discussing the projected performance of each concept on multiple dimensions. This narrowing process did require time, but the teams realized that they needed prototypes to inform the final selection of the strongest concepts.

The parallels to Thomas Edison's famous quote 'Genius is 1% inspiration, 99% perspiration,' are hard to ignore [24]. The 1% of effort required for "inspiration" in Thomas Edison's famous quote mirrors Problem Formulation (what do people need?) and the conceptual work of Idea Generation (what might we make?) and Feasibility Analysis (what should we make?). Edison hyperbolically asserts that realizing an idea (Prototyping and Testing) requires the remaining 99% of the effort. While in our study it only required 56% of the effort, the point that realization takes more time is consistent between this study and the quote.

5.2 Limitations

While this study does provide useful insights into the role of prototyping and realization in design, we must be careful not to overextend the findings.

Perhaps the most significant limitation to the external validity of the work is the design prompt and associated structure of the study itself. A system to detect when a newspaper is taken is relatively small in cost and scale, thereby making it amenable to prototyping. Prototyping would be less likely to be used as extensively for very expensive systems (e.g., airplanes), where simulation and analysis play more important roles. Or, if prototypes would require more time to realize than the allotted time of the study, we would likely see more time used in other design activities.

The structure of the activity itself (e.g., 3 hours, making prototyping supplies available), also influences how a team proceeds. Three hours was chosen as it is a common amount of time used in VPA studies. And, while giving teams prototyping supplies likely implicitly encouraged the teams to realize their ideas, prior VPA studies show prototyping being used even when such supplies are not given to the team (e.g., the previously cited study where teams used tape measurers to prototype in the bike rack study). Further, design teams outside of this study *do have prototyping supplies* available to them; it would be more artificial if we had prevented the teams from accessing prototyping supplies. And, as previously cited, quick and inexpensive prototypes are becoming more accessible with the rise of low-cost physical and electronics prototyping platforms.

The subjects also present a limitation. Senior undergraduates, not experienced engineers, formed the sample for this study. As such, we cannot assert that the behaviors exhibited by the subjects represent *good* design behaviors of experts. Further, it may be that experts with more experience could have answered many of the questions that the subjects used prototypes to answer from experience alone. Finally, the focus of this study is solely on time spent in different activities and different team structures. The quality of the final designs was not assessed. Therefore, we are not able to make any statements about the role of prototyping in developing *better* designs.

6. Closure

While hands-on abilities have been deemphasized for decades, recent advances in prototyping technologies may in fact be helping to reincarnate prototyping in engineering. The teams in this study, when given three hours to design a system to measure when newspapers were taken from a distribution box, spent over half of their time prototyping or testing prototypes to advance their designs. The reason these teams spent so much time prototyping was to evaluate the performance of concepts; prototyping was the quickest and least expensive way to achieve this goal.

The Engineer of 2020 report lists "strong analytical skills" as the first of several attributes of the engineer of 2020, linking these skills to principles of "mathematics, science, and domains of discovery and design." The report continues to define the "core analysis activities of engineering design" to be "establishing structure, planning, evaluating performance, and aligning outcomes to a desired objective" [15]. These activities are, in fact, exactly what the teams were using prototypes to achieve in this study. Instead of using their strong analytical skills, though, the teams were using their prototyping skills to evaluate performance and align outcomes with objectives. As we look to the Engineer of 2040, hands-on prototyping skills should not be ignored as important attributes of engineers.

References

1. H. Clancy, Dean Kamen wants you to make mistakes, *ZDNet*. [Online]. Available: http://www.zdnet.com/article/dean-kamen-wants-you-to-make-mistakes/ [accessed: 31 Jan. 2017].
2. T. Kelley and D. Kelley, *Creative Confidence: Unleashing the Creative Potential Within Us All*, 1 edition. New York: Crown Business, 2013.
3. *The DARPA Grand Challenge: Ten Years Later*, Mar., 2014.
4. S. Denning, Wikispeed: How A 100 mpg Car Was Developed In 3 Months, *Forbes*. [Online]. Available: http://www.forbes.com/sites/stevedenning/2012/05/10/wikispeed-how-a-100-mpg-car-was-developed-in-3-months/ [accessed: 31 Jan. 2017].
5. K. Ulrich and S. Eppinger, *Product Design and Development*, 6 edition. New York, NY: McGraw-Hill Education, 2015.
6. T. Brown, *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*, 1 edition. HarperCollins e-books, 2009.
7. S. Sheppard, *A description of engineering: An essential backdrop for interpreting engineering education*, presented at the Mudd Design Workshop IV, 2003.
8. Criteria for Accrediting Engineering Programs, 2016–2017, *Accreditation Board of Engineering and Technology*, 2016.
9. C. L. Dym, A. M. Agogino, O. Eris, D. D. Frey and L. J. Leifer, Engineering Design Thinking, Teaching, and Learning, *J. Eng. Educ.*, **94**(1), 2005, pp. 103–120.
10. G. Pahl and W. Beitz, *Engineering Design: A Systematic Approach*, 2nd ed. 1996. Corr. London; New York: Springer, 1999.
11. K. Otto and K Wood, *Product Design: Techniques in Reverse Engineering and New Product Development*, Pearson, 2001.
12. R. J. Youmans, The effects of physical prototyping and group work on the reduction of design fixation, *Des. Stud.*, **32**(2), 2011, pp. 115–138.
13. J. Jang and C. D. Schunn, Physical Design Tools Support and Hinder Innovative Engineering Design, *J. Mech. Des.*, 2012, **134**(4), 9pp.
14. M. C. Yang and D. J. Epstein, A study of prototypes, design activity, and design outcome, *Des. Stud.*, **26**(6), pp. 649–669, Nov. 2005.
15. N. A. of Engineering, *The Engineer of 2020: Visions of Engineering in the New Century*, 2004.
16. N. A. of Engineering, *Educating the Engineer of 2020: Adapting Engineering Education to the New Century*, 2005.
17. B. Seely, Patterns in the History of Engineering Education Reform: A Brief Essay, in *Educating the Engineer of 2020: Adapting Engineering Education to the New Century*, National Academy of Engineering, 2005.
18. J. S. Lamancusa, The Reincarnation of the Engineering Shop, presented at the *ASME International Design Engineering Technical Conference*, 2006, pp. 849–857.
19. N. Cross, H. Christiaans and K. Dorst, Eds., *Analysing Design Activity*, 1 edition. Chichester; New York: Wiley, 1997.
20. M. E. McFarland and R. Bailey, Investigating Pattern in Design Performance of Interdisciplinary Undergraduate Engineering Student Teams, presented at the *2015 ASEE Annual Conference & Exposition*, Seattle, WA, 2015, pp. 26.1038.1–26.1038.17.
21. P. L. Moore, C. J. Atman, K. M. Bursic, L. J. Shuman and B. S. Gottfried, Do freshmen design texts adequately define the engineering design process?, presented at the *Proceedings of the 1995 Annual ASEE Conference*, Part 1 (of 2), 1995.
22. V. Kumar, *101 Design Methods: A Structured Approach for Driving Innovation in Your Organization*, 1 edition. Hoboken, N.J: Wiley, 2012.
23. J. R. Landis and G. G. Koch, The Measurement of Observer Agreement for Categorical Data, *Biometrics*, **33**(1), 1977, pp. 159–174.
24. *Thomas Edison*, [Online]. Available: https://en.wikiquote.org/wiki/Thomas_Edison [accessed: 22 April 2017].

Reid Bailey is an Associate Professor in the Department of Systems and Information Engineering at the University of Virginia. His primary interests are engineering design, engineering education, and the environment. He earned a BSE from Duke University, an MSME from the Georgia Institute of Technology, and a PhD from the Georgia Institute of Technology, all in Mechanical Engineering.

Matthew E. McFarland is a Product Manager at Alarm.com located in the Washington, DC area. He attended the University of Virginia from 2011–2014, where he conducted research on undergraduate students and engineering education. He graduated from there with a MS in Systems Engineering in 2014 and from North Carolina State University with a BS in Mechanical Engineering in 2010.