Abstract. This paper presents a mixed-method research design investigation that integrates a Hybrid Digital-Analog Software-Hardware protocol referred to as the No Keyboard, No Mouse (NK-NM) platform. The NK-NM process uses both theoretical and applied research mechanisms to measure its influence on architectural design decision-making, knowledge exchange, student learning, aesthetics, and user experience in the context of an undergraduate architectural design studio. Observing a recognized gap in the current digital architectural design environments this paper details how the NK-NM protocol bridges this gap through an instructed hierarchical design process, customizable physical interface, and iterative simulation-based feedback loop.

Keywords. Digital Hardware; Digital Design; Pedagogy; Human-computer Interaction; Physical computation.

1. Introduction:
A literature review revealed that tools and methods for making and engaging with design, engineering, and architecture are becoming both more pervasive, integrated, and accessible. Combined with rapid prototyping methods that are becoming more ubiquitous and affordable there is an opportunity to develop a formative framework of computationalism and experiential movement. As a consequence of this “freedom,” there is an increased necessity to question conventional methods for design, which is particularly true for design studio environments infused with design research modalities that challenge students to produce performative results regardless of their personal level of knowledge or design acumen. When it comes to design students with limited design expertise or non-design students with no design background, understanding how to make “valid” design decisions that leverage the vast possibilities of design software result in two extremes - either the hesitancy to make informed decisions or the increased facility to make uninformed decisions.
The generative idea behind the No Keyboard/No Mouse (NK-NM) research contributes to the established culture of “instruction” in the design studio, where a framework of guidelines, limitations, and programmatic definitions inform the design solutions as both 2-, 3-, and 4-dimensional digital and fabricated outcomes. The method proposed in this paper embeds the traditional vehicle for instruction between the instructor and students in the design software/hardware platform as a new design tool/device: NK-NM. The NK-NM research project improves upon earlier investigations where design-researchers attempted to develop innovative design platforms that facilitated creative interactions between the designer and non-design oriented user groups. These previous studies deployed only enhanced digital design processes limited to software and algorithmic developments. The NK-NM research project, however, is not constrained by the software or digital design environment. Instead, it leverages existing design software and programming platforms to heighten user experience by formatively situating the user and the design platform within an interactive context that harnesses the “master-designer’s” knowledge and then connects it with qualities that can be tailored to define design parameters, rules, and controls.

2. Guiding Questions:

This research examines three primary research questions:

1. How can the proposed NK-NM platform, challenge the traditional design process and decision-making by introducing an “instructed” hierarchical design-making matrix?
2. How can the design interface become easier to work with and enable more in-depth exploration through the use of actual physical interface/hardware design?
3. How can the NK-NM instructional platform make digital design protocols and rapid-prototyping processes accessible in response to today’s rapidly changing and increasingly more complex environments?

3. Informal Observations:

The traditional dialogue between architectural design platforms, game design, and the gaming environment presents a range of different approaches to operating with information from its processing to its representation. William J. Mitchell when discussing the computer processors and programming, concerning their operating timeline, says that the first computers “were strictly sequential machines executing one operation at a time; programming was a matter of specifying these operations in precise order.” He then compares that to the new computational process: “For example, computer animations of three-dimensional environment could be computed and stored for later playback, or as in today’s video game framework, they could be computed and then presented on the fly, with no perceptible time lag” (Mitchell, 2003).
Table 1. Comparing design and gaming platforms with daily devices regarding user interface.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Advanced Software Interface</th>
<th>Interactive Physical Presence</th>
<th>Input</th>
<th>Output</th>
<th>Relationship between Input/Output</th>
<th>Learning process complexity</th>
<th>Level of experience to use the platforms level of difficulty</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kitchen Accessory, e.g. Microwave</td>
<td>No</td>
<td>Yes</td>
<td>Keypad</td>
<td>Direct / Short process</td>
<td>Intuitive / Fast</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Smartphone</td>
<td>Yes</td>
<td>Yes</td>
<td>Keypad</td>
<td>Direct / Short process</td>
<td>Intuitive / Fast</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Smart TV</td>
<td>Yes</td>
<td>Yes</td>
<td>Controller</td>
<td>Direct / Short process</td>
<td>Intuitive / Fast</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>X-Box / Gaming</td>
<td>Yes</td>
<td>No</td>
<td>Controller</td>
<td>Direct / Short process</td>
<td>Intuitive / Medium</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Digital Design Platforms</td>
<td>Yes</td>
<td>No</td>
<td>Keyboard/ Mouse</td>
<td>Indirect / Long process</td>
<td>Interactive / Short</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Programming Board design platform</td>
<td>Yes</td>
<td>No</td>
<td>Keyboard/ Mouse</td>
<td>Customizable operations</td>
<td>Indirect / Long process</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

The NK-NM research identified a “qualitative” gap between the experienced designer-Instructor and/or upper-level student and the “user” or beginning student. In the theoretical framework of design critiquing in architecture studios, Yeonjoo Oh asserts that “Moore argues that instructors in architectural studios have followed ingrained conventions through generations without seriously examining the underlying pedagogy (Moore, 2001). While a practicing architect no doubt brings a great deal of experience to the studio, their teaching methods are often based only on their own learning experiences or on intuition (Grasha, 1996). They often cannot articulate what instructional method they are using, or is appropriate, for a specific condition” (Oh, Ishizaki, Gross, & Do, 2013).

Similar to the design profession, architectural design software platforms look beyond disciplinary boundaries for inspiration. These platforms include those utilized by the game industry who leverage the possibilities by the film and animation, industrial design, media arts, etc. Although the simulation/visualization gap seems to be bridged, there is still a considerable gap between the way that a gamer operates the game “software” and the way that designer/architect operates an architectural design platform. Within pre-defined boundaries, the gamer is in complete control of the game “scene.” In this case, the pre-set environment and its boundaries are helping the gamer to stay within the “scenario” of the game. It maintains the design characteristics of the game while each gamer situates their unique experiential approach to solving the game. In the architectural design platform, the “designer” is similar to the gamer. The designer has a defined toolkit and a specific set of operations. The difference, however, leaves the “designer” in a vast, open environment of possibilities. While this flexibility provides a range of opportunities to design for experienced designers that enable them to define their own “environment,” it also presents a range of challenges for those who are less experienced users with the design platform.

In the gaming environment, each “command” presents multiple digital functions combined and embedded in the core of the “software.” Jose Sanchez refers to the “command container” as a combinatorial design (Sanchez, 2016). In addition to the combinatorial design qualities, the game environment can be customized, using a controller hardware interface such as joysticks with multiple inputs and assigning each input to a task, whereas in an architectural design
platform, the designer combines multiple “commands” to complete a “function.” Moreover, since the physical controlling interface in the architectural design platform is bound to the limited number of keys on the keyboard and the movement restrictions of the mouse, there is a limited range of possibilities that inform the customization of the physical controlling interface.

Since 2014, researchers and designers such as Jose Sanchez, have tried to address these issues in a rigorous research context in his articles such as Combinatorial design: Non-parametric computational design strategies, (Sanchez, 2016). The origin of this idea, of course, has roots in the interactive gaming theories of Ian Bogost who explains the difference between unit and system in different platforms (Bogost, 2006). In Bogost’s approach towards the connection of the architectural design environment and the possibilities of a gaming environment through combinatorial design, Sanchez uses the game engine platform to define the design process and to create set of “commands” together as a cluster. This new way of thinking about design process will change the decision-making process and the hierarchy. As Sanchez explains: “What is described here as games draws concepts from gaming culture, but mainly refers to a guided simulation where the decisions, in this case, the player, start generating a branching narrative. The narrative is the geometric development and the decisions of function and performance embedded by a designer. These game simulations generate a strong feedback loop between the computational constraints and the intuition and experience of a human designer” (Sanchez, 2016).

Although the proposed method by Sanchez is tremendously valuable and bridges part of the gap between the interactivity of the game design and its possibilities as an architectural design process, it has two main challenges:

1. Most of the design process occurs either through pure coding or outside of the “native” architectural design platforms.
2. The Hardware and Physical interfaces are limited to the keyboard, mouse and gaming controllers, which limits the degree of customization.

4. Methodology:

The formulation of the hypotheses and an overview of the qualitative and quantitative research methods used to test the hypotheses and explore three intrinsically linked categories: the design process, the controlling/decision-making software platform, and the physical interactive controlling hardware device. As part of a more extensive interactive design machine exploration, this research simultaneously works across the digital and physical environments to bridge the gap caused by the lack of clear and direct communication between the digital design environment and the physical input hardware interfaces.

5. Definition of Terms:

To better understand the NK-NM process, it is necessary to know the vocabulary used in its research:
Master-Designer: The experienced designer that defines the guidelines of the design process through the script and the physical interface (hardware).
Second-Designer: The user/designer with no or minor design experience that interactively designs within the flexibilities defined by the master-designer.
Hosting Software: The primary design platform that the NK-NM protocol builds. In this paper, the hosting software is Rhinoceros 3D.
Secondary Software: The script developed by the master-designer to operate based on the hosting software as means of defined design rules.
Command Cluster: A group of commands and functions that can be controlled all at the same time with one digital or physical input.

6. Limitations and Delimitations
The main limitation of this study is the small number of students involved in the initial research as master-designers and second-designers. The constraints that have affected the precision of the outcome of this investigation are:

- The master-designers of the NK-NM protocol had differing design backgrounds. As a result, the primary two issues (NK-NM devices) studied in this paper, are limited to those design backgrounds and interests.
- The users of the NK-NM platforms as second-designers are limited to the third and fourth-year architecture students and a limited number of participating faculty members. Although the difference and similarities of the outcomes are valuable, the research could benefit from a more diverse group of users.

Despite these limitations, every possible effort was made to design the research in a way that maximizes the potential contribution of the study’s findings of how the NK-NM platform could inform design pedagogy. At the time of this writing, the NK-NM platform continues its development through a collaborative partnership between faculty at Kent State University - College of Architecture and Environmental Design (CAED) and the University of Kentucky - College of Design-School of Architecture. The goal is to maintain the platform development and then introduce it, as a rigorous and comprehensive research project, into various levels of design studios and measure its impact.

7. The No Keyboard, No Mouse (NK-NM) Protocol
NK-NM has its roots in the architectural design studio and is a direct response to the recognized lack of interactive, intuitive and organic design workflow in the teaching/learning process of digital design. NK-NM is a bridging protocol for design developed in both digital software and the physical interface/hardware. Most of the digital tools in the field of architecture are lacking the self-learning qualities that are prominent in digital game platforms. Although in a universal digital design platform “everything” is possible, based on the defined limits of the process, only some of the outcomes are valid. Based on the aforementioned gap, NK-NM is a response to the need for a more controlled design platform for the assigned tasks. With the use of customized design software (secondary software) and the hardware (physical interface) the process validates that the outcomes meet the design limitations.
The diversity of the possible scripting platforms is contingent on the hosting software; in the case of this research, the digital controlling platforms (secondary software) develops through scripting in node-based visual coding platform for Rhinoceros, Grasshopper, Python, C Sharp, C, or Java. The direct connection to the hosting design software as a design platform enables the master-designer to design the secondary software, with the features of the hosting software but as a container of command clusters.

The NK-NM protocol has two design decisions categories; Master-designer’s design decisions and the second-designer’s design decisions.

8. The Master Designer’s Design Decision process:

The work follows of the NK-NM platform contains two main steps in the production mode:

1. Secondary Software Development: Defining the function/command “cluster,” design “rules” and outcome through scripting in the digital design environment. During this step, using any scripting language the master-designer defines the functions/rules of the design environment. Similar to the game environment, the tasks and the environment/scene are flexible enough to be set at different scales, levels of restriction, environmental forces, and various operations, etc.

2. Hardware Development: Designing a corresponding controller, as the physical interface to make a more tangible connection between the act of design and the steps of the design process. The master-designer makes the use of command clusters and the secondary software even more tangible by providing the second-designer with a customized hardware/physical interface to enable the second-designer to learn the design process in a more intuitive/experimental way by exploring the direct effects of the physical interface on the digital 3d model.

Through these steps, the master-designer instructs the second-designer. Based on the interest and the goal of the design, the master-designer can also include fabrication controls as part of the NK-NM protocol.

9. The Second Designer’s Design Decision process:

Every design decision is happening within the limitations and design rules by the master-designer embedded into the NK-NM protocol. Although the rules are defined and while the second designer cannot break them, the second designer is entirely free to design within constraints as long as the outcome meets the limitation and the capabilities of the physical interface.

The workflow for the second-designer contains one main step before starting the design process; understanding the interface through interaction. Since NK-NM assigns each of the inputs on the physical interface (controller) to a command cluster, the user would immediately see a design outcome and the effect of the controlling data. This fast and intuitive learning process enables the second-user to learn the possibilities of the design tool within minutes.

As part of the NK-NM research, this workflow has been tested through two different projects looking at both surface design and massing studies.
10. NK-NM SoftFracture | Digital Surface Design

The NK-NM SoftFracture is a digital/analog software/hardware hybrid protocol that studies surface design as an instructed process. As an NK-NM protocol, the SoftFracture design process starts with the master-designer who begins by observing the hosting digital modeling software, Rhinoceros 3D. The master-designer begins the modeling process by determining an “architectural” surface in Rhinoceros 3D. Then, by recording the computer screen, analyzes each step of the modeling process and transform them into a sequence of comprehensible moments. Using the analytic information from the screen recordings and extracted modeling steps, the master-designer can create the design a flowchart/decision-making matrix for the modeling process. (Figure 1)

Figure 1. SoftFractures, Digital Surface Design Process in the Hosting modeling software, Rhinoceros 3D.

Considering the hosting-software, Grasshopper was used as the native node-based visual programming platform in the Rhinoceros 3D to develop the second-software. The master-designer develops the computational flowchart diagram of the process, aiming for reducing the steps by creating command clusters.

Using the information from the second-software development, the development of the last component of the NK-NM protocol starts; the physical interface: Controller. Using a programmable microcontroller Arduino, the master-designer studies different physical iterations and possibilities to use analog sensors to receive input data and change values from the microcontroller. Using the bridging platform Firefly—an add-on for Grasshopper developed by Andy Payne and Jason Kelly Johnson, the research team received data in the Grasshopper environment and input it to drive command clusters. (Figure 2)
By having the connection, the master-designer developed different iterations for the physical interface and the controller. The identified “favorable” option had enough input to control the surface and afford the possibility of being hand-held. The outcome of the process is the NK-NM SoftFracture. (Figure 3)

**11. NK-NM BoxCron | Digital Massing Study / 3D printing**

The design process of the NK-NM BoxCron is very similar to the NK-NM SoftFracture regarding the workflow. The significant difference, however, is the fact that the NK-NM BoxCron considers 3D printing and its restrictions as part of the design instructions embedded in the protocol.

Similar to the NK-NM SoftFracture, the NK-NM BoxCron starts with an observation of the digital modeling process of a massing study in Rhinoceros hosting software. During the development of the final second-software in Grasshopper, considering 3D printing as a desired fabrication method, the master-designer introduces another layer of guideline and instruction through the protocol. Based on the characteristics of the 3D printer used for this study the minimum thickness of the designed massing model should not be less than a millimeter and the maximum size of the outcome model could not be bigger than a cube of 15 cm. (Figure 4)

By embedding the fabrication limitations and instructions in the second-software, the master-designer starts the process of designing the physical interface. Also, similar to the NK-NM SoftFractures project, through different controller iterations, the NK-NM BoxCron master-designer had a hand-held and
wireless physical interface. Through multiple magnetic connections, the NK-NM BoxCron controller reduced the amount of wiring and created a semi-wireless controlling system. (Figure 5)

![Figure 4. BoxCron, Digital massing and 3D Printing study in the Hosting modeling software, Rhinoceros 3D.](image)

![Figure 5. Massing studies for undergraduate design, using No Keyboard, No Mouse BoxCron platform.](image)

The NK-NM platform assigns analog sensors input value to the command cluster controller and the second-software to achieve the connection between the controller, the hosting software, and the second software. As an outcome of the NK-NM BoxCron protocol, the generated models produced developmental massing through a Boolean operation using cubes, spheres, and custom geometries and a process with complete control of the transformation, deformation, and density factors. The method successfully met the limitations of the 3D printing.

12. Results

Observing the interactive and intuitive learning process/methods in the gaming process, the suggested approach in this paper produced an intuitive, self-learning process through embedded guidelines. The interactive learning process led to a faster and deeper understanding of the digital modeling or digital fabrication process. Observing the immediate design outcomes of the controller enabled the user, as the second-designer, to find the possibilities of the hosting design software more accessible through the use of command clusters.

The NK-NM research has produced four primary outcomes:

1. A new method of interaction between the designer and the digital design environment, through both digital and physical platforms, using a hierarchical
design process between master and second designer.

2. A tool for enabling designers with limited design experience to design in an instructed design platform, securing the design outcome validation as both a digital and digitally fabricated physical model.

3. The introduction of physical design interface as a method to increase the design awareness for designers with limited design background by increasing the tactual design interaction between the user and the design software.

4. The increased ability of design instructors to introduce digital design environment in earlier stages of design education through physical, tangible interface.

13. Conclusion

The NK-NM process advances research in the domain of physical computation and human-machine interaction by using both digital simulation and hands-on investigations. This paper demonstrates the potential of NK-NM as a pedagogical tool for extending and reshaping the design and design-thinking processes by utilizing real-time physical/analog inputs and feedback. The initial results of the research narrowed the gap between experienced designer and designers with minor expertise. This study moves beyond the conventional hardware and software settings to produce a real-time interactive toolkit of design parameters – a data collection system that has converted design qualities into quantitative controllable inputs. This customizable process encourages the hands-on procedures as a feedback and simulation loop from the very early stages of design.

14. Future Research

This paper outlines a platform that enables an iterative analog model making process linked to the digital design environment. One component of the future research agenda of this platform will demonstrate Patkau Architects’ physical-analog modeling process. In this examination, the NK-NM will bridge between the analog model and the analysis of inherent material properties across scales, to responsively inform a digital model. The research team anticipates that subsequent feedback loops between the digital models will allow for the generation of adaptive analog models. In this way, the NK-NM platform will directly inform the physical model iterations and ultimately influence, the final built form.

References


