Bendit_I/O: A System for Extending Mediated and Networked Performance Techniques to Circuit-Bent Devices

Anthony Thomas Marasco
Louisiana State University and Agricultural and Mechanical College

Follow this and additional works at: https://digitalcommons.lsu.edu/gradschool_dissertations

Part of the Composition Commons, and the Other Music Commons

Recommended Citation
https://digitalcommons.lsu.edu/gradschool_dissertations/5343

This Dissertation is brought to you for free and open access by the Graduate School at LSU Digital Commons. It has been accepted for inclusion in LSU Doctoral Dissertations by an authorized graduate school editor of LSU Digital Commons. For more information, please contact gradetd@lsu.edu.
BENDIT_I/O: A SYSTEM FOR EXTENDING MEDIATED AND NETWORKED PERFORMANCE TECHNIQUES TO CIRCUIT-BENT DEVICES

A Dissertation

Submitted to the Graduate Faculty of the Louisiana State University and Agricultural and Mechanical College in partial fulfillment of the requirements for the degree of Doctor of Philosophy in

The School of Music

by

Anthony Thomas Marasco
B.A., Lebanon Valley College, 2008
M.M., Towson University, 2010
August 2020
ACKNOWLEDGEMENTS

On the eve of completing my doctoral studies, I would be remiss in not expressing my sincere gratitude to the friends and family members who have supported me along this journey and throughout my life. The research undertaken here would not be possible without them, because it is they who are responsible for making me the man that I am today.

Thank you to the members of my committee—Dr. Hye Yeon Nam, Dr. Feng Chen, Dr. Stephen Beck, Dr. Edgar Berdahl, and Dr. Jesse Allison—for your skillful guidance through this project and your insightful comments and input on shaping this research into its final state.

Thank you to Dr. Jesse Allison, who took a chance on me when a dozen other schools had turned me down, and offered me a position of a life time in a program that could not have been a more perfect fit. Your constant support, encouragement, guidance, and friendship has made my time here in Louisiana one of the highlights of my life. I look forward to more collaborations and our growing friendship in the future as I take the things you’ve taught me out into the world.

Thank you to my friends spread far and wide from the Pennsylvania mountains to the Louisiana bayous, who have lifted me up in my worst moments and celebrated with at my best. To those who I’ve shared the journey with at LSU—Eric, Chase, Anna, Andrew, Bethany, Tate, Jess, Will, Matt, Katlin, Thomas, Kaci, Vanessa, and Dr. Frank—and those who have been by side for almost 20 years—Todd, Tom, Charlie, Mike, Emily, Dara, Erica, and Mary—I would not have gotten to this finish line without you.

To every leaf on my family tree—the Marascos, Sousas, Cloreys, and more—thank you for always showing me how to be strong enough and resilient enough to tackle the road ahead, and for asking me "So what kind of music do you write again?" enough times that explaining it in detail by now is a walk in the park.
Thank you to my sisters and my mother, for being fearless and brave each and every day and for teaching me how to be kinder and funnier than I ever thought possible.

Most of all, thank you to my wife Monica, who came into my life by surprise, shaped my life into something fantastic, and made my future bright. This paper would not exist without your numerous suggestions, edits, and feedback. Your love, compassion, strength, and joy have no limits.

Also, thank you to my dog Simon, who truly is the best boy.
## CONTENTS

ACKNOWLEDGEMENTS ........................................... iii

LIST OF FIGURES .............................................. vii

LIST OF CODE EXAMPLES .................................... viii

ABSTRACT ......................................................... ix

INTRODUCTION ................................................... 1

CHAPTER 1. AN OVERVIEW OF CIRCUIT BENDING ............... 3
  1.1. Definition and Practices ................................ 3
  1.2. Connections to Artistic Philosophies ..................... 8
  1.3. Notable Works featuring Circuit-Bent Devices ......... 13

CHAPTER 2. AN OVERVIEW OF NETWORKED MUSICAL PERFORMANCES .................. 24
  2.1. Definition and Practices ................................ 24
  2.2. Collaborative and Distributed Works through Networked Performance Technologies ............................................. 35

CHAPTER 3. A CASE FOR NETWORKING CIRCUIT-BENT DEVICES IN PERFORMANCE .... 42
  3.1. Networking the Circuit-Bent Readymade .................. 45
  3.2. Media Archaeology and Hardware Democratization through Networked Performance of Circuit-Bent Devices ..................... 49

CHAPTER 4. BENDIT_I/O ......................................... 52
  4.1. Design Criteria .......................................... 52
  4.2. System Components — Hardware .......................... 54
  4.3. System Components — Software .......................... 60
  4.4. System Performance ....................................... 66

CHAPTER 5. COMPOSING WITH BENDIT_I/O .......................... 70
  5.1. The Spinning Earth Shall Spread Before You ............ 70
  5.2. gravity|density ........................................ 77
  5.3. Camera Studies, Vol. 1 ................................... 84

CONCLUSION ......................................................... 93

APPENDIX A. PERFORMANCE INTERFACE CODE .......................... 96

APPENDIX B. PERMISSIONS ....................................... 103
LIST OF FIGURES

1.1 A performer circuit bending a Furby toy. .............................................. 6
1.2 A circuit-bent Casio SK1. ................................................................. 8
1.3 A comparison of a Duchamp readymade and a circuit-bent device. .......... 11
1.4 The circuit-bent CD player and remote control used in *Broken Light*. .... 17
1.5 A circuit-bent digital camera used in *Year of the Glitch* and *DCP series*. 20
1.6 Two forms of a glitch generated and captured through a circuit-bent digital camera. 22
2.1 Interaction topology graphs designed by Matuszewski et al. ................. 30
3.1 Performative readymades in *Chorus for Untrained Operator*. ............. 43
3.2 Discarded objects archived in *The Bureau of Suspended Objects*. ........ 50
4.1 A v1.0 model Bendit board. ............................................................. 54
4.2 Example of a Bendit board circuit-bending a digital camera. ............... 55
4.3 A v2.0 model Bendit board. ............................................................. 58
5.1 The control data network for *The Spinning Earth Shall Spread Before You*. .... 72
5.2 Modifications made to a Sony Discman D-E301 CD player. ................. 75
5.3 Circuit-bent Bendit_I/O-enabled CD player placed in a custom enclosure. .... 75
5.4 Web performance interfaces for *The Spinning Earth Shall Spread Before You*. .... 77
5.5 The control data and audio-routing network for *gravity/density*. .......... 78
5.6 On-stage performers for *gravity/density*. ......................................... 81
5.7 Bendit_I/O-enabled hardware for *gravity/density*. ............................ 83
5.8 Audience participant and performer web interfaces used in *gravity/density*. .... 83
5.9 Images from a performance of *Camera Studies, Vol. 1*. ..................... 86
5.10 The control data network for *Camera Studies, Vol. 1*. ...................... 89
5.11 Bendit_I/O-enabled hardware for *Camera Studies, Vol. 1*. ................ 91
LIST OF CODE EXAMPLES

4.1 Example Bendit I/O Server Code with benditHub.js Library . . . . . . . . . 62
4.2 Defining a new BenditDevice Object with the benditBrowser.js Library . . . . 64
4.3 Addressing Bendit Board i/o Channels with benditBrowser.js . . . . . . . . . 65
A.1 Web Performance Interface for The Spinning Earth Shall Spread Before You (2020) 96
ABSTRACT

Circuit bending—the act of modifying a consumer device’s internal circuitry in search of new, previously-unintended responses—provides artists with a chance to subvert expectations for how a certain piece of hardware should be utilized, asking them to view everyday objects as complex electronic instruments. Along with the ability to create avant-garde instruments from unique and nostalgic sound sources, the practice of circuit bending serves as a methodology for exploring the histories of discarded objects through activism, democratization, and creative resurrection. While a rich history of circuit bending continues to inspire artists today, the recent advent of smart musical instruments and the growing number of hybrid tools available for creating connective musical experiences through networks asks us to reconsider the ways in which repurposed devices can continue to play a role in modern sonic art.

Bendit_I/O serves as a synthesis of the technologies and aesthetics of the circuit bending and Networked Musical Performance (NMP) practices. The framework extends techniques native to the practices of telematic and network art to hacked hardware so that artists can design collaborative and mediated experiences that incorporate old devices into new realities. Consisting of user-friendly hardware and software components, Bendit_I/O aims to be an entry point for novice artists into both of the creative realms it brings together.

This document presents details on the components of the Bendit_I/O framework along with an analysis of their use in three new compositions. Additional research serves to place the framework in historical context through literature reviews of previous work undertaken in the circuit bending and networked musical performance practices. Additionally, a case is made for performing hacked consumer hardware across a wireless network, emphasizing how extensions to current circuit bending and NMP practices provide the ability to probe our relationships with hardware through collaborative, mediated, and multimodal methods.
INTRODUCTION

In *Semiconducting: Making Music after the Transistor*, the composer and hardware hacker Nicholas Collins expounds on the differences between hardware and software tools as they apply to the acts of music performance and composition. Discussing what he deems are the most important differences between how these tools suit different creative tasks, Collins underlines an intriguing duality in the hardware’s value to musicians:

> Before its replacement by a superior device, qualities of a tool that don’t serve its main purpose can be seen as weaknesses, defects, or failures: ticks and pops, oscillators drifting out of tune, tape saturation and distortion. But when a technology is no longer relied upon for its original purpose, these same qualities can become interesting in and of themselves. The return to “outmoded” hardware is not always a question of nostalgia, but often an indication that the scales have dropped off our ears.\(^1\)

Circuit benders have aligned with Collins’ perspective but have shifted beyond his focus on media and recording devices and into a realm that includes embracing the imperfections of toys, appliances, and home video equipment. While software facsimiles of these devices will always offer the average consumer more features and higher fidelity, the practice of circuit bending goes beyond a need for replication or perfection, opting instead for resurrection and remediation. The creation of the Bendit\_I/O framework is a modern approach to capitalizing on the underlying ethos of the circuit bending practice and Collins’ statement, namely, that the inherently fallible traits of hardware should be embraced by artists and exploited for creative purposes. The creation of Bendit\_I/O addresses methods for building on the technological, performative, and philosophical traits of circuit bending for growing it into a practice that uses software for the purpose of bringing collaborative and mediated performance techniques to repurposed devices. In this manner, the irreplaceable traits of electronic devices rewired through chance and exploration are not replicated by software but

instead extended through it, bring them into circles of collaboration and interaction never possible in their first or second lives.

The practice of circuit bending does not need saving. Its core values do not need to be restructured or reconfigured; its message is evergreen. But the landscape around hobbyists and hackers is changing, and in response, they have used the internet as a new concert space for their creations and community workshops around the world as spaces for hardware exploration. What else can web tools and wireless connections bring to the circuit bending experience, and can they be used to change the way we interact with remediated creations? Spare-room labs and backyard shed workstations are no longer the only places where the sounds of spastic oscillations born from a once docile consumer device ring out, but there’s still room to grow. How can circuit bending as a performative practice move from a singular experience and into a shared, reactive, communal environment?

I built my first circuit-bent instrument in a rock-floor basement in my childhood Northeastern Pennsylvania home. The experience of culling unexpected sounds never anticipated from a device that I thought I understood was an experience that has and will continue to be transformative to me with each new device I open. Bendit_I/O does not aim to replace those moments, nor can it. As benders continue to push forgotten devices beyond their limit, embracing the ticks and pops and new sounds yet to be discovered, it’s my hope that collaborative, interactive, and communal abilities of networked musical performance that Bendit_I/O brings to their workshop tables will serve as a means for expanding the walls around them while they focus ever inward.
CHAPTER 1. AN OVERVIEW OF CIRCUIT BENDING

1.1. Definition and Practices

Writing on his process of exploratory sound design through the manipulation of electronic hardware, instrument designer Qubais Reed Ghazala first coined the term "circuit bending" in the September 1992 issue of Experimental Musical Instruments magazine at last putting a name to his decades-long practice of creating new connections between the internal circuitry of an electronic device for the purpose of evoking unexpected sonic results. Ghazala conceived the term as a means of differentiating between what he saw as two complementary avenues of his craft: the creation of carefully designed electronic instruments intended to produce a predetermined set of musical operations and the act of commandeering, subverting, and remediating existing devices into service as generators of unpredictable sonic textures.

Musically, circuit bending results in the creation of avant-garde instruments, customized by their creators after exploring the sonic and visual reactions that come from connecting disparate points on a device’s circuit board(s) through the addition of new, permanent wiring or through an external interfacing element that can temporarily make and break the connection when required. The practice draws a connection to the Do-It-Yourself (or DIY) ethos of hardware modification and development prevalent among post-World War II era. Experimental musicians such as Pierre Schaeffer, David Tudor, Gordon Mumma, and John Cage created technological experiments ranging from the modification of tape players and phonographs to create delay lines and endlessly-looping audio tracks, to the amplification of

---


found objects for the use of their unique sonic timbres, to the creation of original electronic systems for live signal processing in performance.⁴

The work of Louis and Bebe Barron serves as another layer of the foundation for the technical and philosophical practices that Ghazala would later build his practice upon. As part of their joint composition efforts, they created original sound generators through the cobbling together of variable resistors, capacitors, and vacuum tubes to create unpredictable soundscapes. Their methodology sat somewhere between the extension-through-planned-modification approach taken by the aforementioned composers and the anarchic aleatoric explorations at the heart of circuit bending. In their search for uncharted sonic territory through circuitry, the Barrons’ creations were not intended to emulate pre-existing instruments and were purposely designed to react in more chaotic and “organic” ways than the professionally-built equipment of the time.⁵ The devices’ unique electrical designs subverted the intended operation of the vacuum tubes at their heart, overloading them with voltages higher than they were ideally rated for in order to cull new sound sources by putting them through stress. “You have to be free to abuse the circuit,” said Louis, a statement in the spirit of what would later become a core aspect of the circuit bending ethos.⁶

Since the inception and public unveiling of the term, circuit bending has risen in prominence as an artistic practice, outlet of aesthetic expression, and research topic, running parallel to (and often intersecting with) the broader DIY, Upcycling, and Hacktivist cultural movements.⁷ To understand how Bendit_I/O replicates and embraces circuit bending, it is important to highlight the core performative and philosophical aspects of the practice.


1.1.1. Source Devices

The source devices commonly remediated into circuit-bent instruments are low-voltage, consumer electronic devices. The focus on using battery-powered toys, media players (such as portable cassette players, CD players, DVD players, etc.), and tabletop electronic instruments as circuit bending material stems from two important tenets of the art form’s practice. Firstly, circuit bending is rooted in the act of exploration; practitioners are encouraged to seek out novel and wild audiovisual results by opening up their chosen device and rerouting the intended flow of electricity by making new connections (with either wire or with their flesh) between unconnected points along the circuit board\(^8\) (an example of this process in action can be seen in Figure [1.1]). By limiting their selection of source devices to those that use batteries as a power-source and that operate under nine volts, the user minimizes the risk of inflicting a fatal shock or other bodily harm by accidentally sending too high of a voltage into sensitive components, reversing a component’s polarity, or by creating a short between a positively-charged power rail and ground using their own body. Secondly, the use of an existing, working device capable of creating audio or visual material provides circuit benders with a familiar foundation from which they can build their new instrument. Connecting points together on a preexisting toy allows the user to draw an immediate comparison between the newly-created sonic abilities of their hacked device and those possible in its normal state of operation. Put simply, starting from a place of familiarity allows benders to easily notice the impact of their work. These core aspects of circuit bending provide a relatively low-risk entry point into the larger world of electronic instrument design and construction.

---

Figure 1.1. A performer circuit bending a Furby toy. Wire is used to route the flow of electricity through previously-unconnected points on the device’s circuit board. Screen capture from a video essay by The Verge, 2018. From The Verge website, https://www.theverge.com/2018/9/14/17844906/circuit-bending-hacking-a-furby-in-the-name-of-music. Reprinted by permission.

to amateur electronicists and experimental artists, avoiding the learning curve that comes with building new systems or performance tools from scratch.

1.1.2. Methods of Performing

After discovering new connections to make across a device’s internal circuitry, circuit benders often solder wires to these locations, adding adjustable performance interface elements in-between so that connections (or "bends") can be made or broken at a moment’s notice.

9The use of the term "amateur" in research related to circuit bending can be misleading, as it is typically used to denote a practitioner of a craft who has limited knowledge of the subject or one who does not engage in the work for professional purposes. While this may be the case for many who practice circuit bending, the art form’s long history and alluring sonic prospects, coupled with the wealth of openly-shared and available guides and resources for creating circuit-bent devices (including a number of publications and academic textbooks), has brought a number of professional electronicists, instrument designers, and dedicated hobbyists into the fold. The extent of circuit bending’s reach makes it difficult to justify applying such a term to seasoned electrical engineers and electronic instrument luthiers, even if they have minimal experience with the specific tenets of the practice. For the sake of this research, the use of the classifiers "amateur" and "non-expert" will therefore be used to denote a person who does not consider themselves to be a professional electrical engineer (working independently or as a repair technician for a manufacturer of consumer electronic devices) by trade.
The addition of these interface elements provides artists with the ability to trigger specific bends at their discretion, bringing their chaotic sound or visual generator into the realm of a performative instrument. The interface elements chosen by a circuit bender for use on their instrument vary based on the required action that needs to be exerted upon the bend points that it connects. Some common interface elements and their corresponding patching/manipulation methods are:

- **Switches**: used to create a direct, unmodified connection between bend points that can be latched and held for long periods of time.
- **Momentary push buttons and metal pads/strips**: used to create a momentary, unmodified connection between bend points for short periods of time.
- **Potentiometers/Rheostats**: used to generate variable levels of voltage or resistance between bend points.
- **Photoresistors**: used to translate changes in light into adjustable resistance/voltage levels between bend points.
- **Joysticks/Foot pedals**: larger, more robust interfaces for generating resistance/voltage changes between bend points through expressive gestures.
- **Patch bays**: used to connect multiple bend points together through external cords or cables. These are used to create semi-permanent bends that can be duplicated between multiple performances (analogous to patching between points of a modular synthesizer).
- **Flesh**: connections between bend points are made through direct touching of the exposed circuitry with the performer’s body.

Circuit-bent instruments that utilize more musically-oriented interfaces or forgo physically-tactile interfaces exist, but these design choices require more laborious soldering or the addition of voltage-regulating circuitry. Examples of instruments with more complex interface

---

Figure 1.2. A circuit-bent Casio SK1. The added patch bay, switches, and potentiometers are used to connect bend points or adjust resistance resistance in the circuit during performance. Photograph by Paul Shovel, 2017. From CircuitBenders web archive, https://www.circuitbenders.co.uk/synthmod/SK1.html. Reprinted by permission.

elements include the piano keyboard driven Sega Megadrive Synth\textsuperscript{11} and Furby Organ\textsuperscript{12} and the control-voltage modulated parameters of the Gameboy Triple Oscillator synthesizer module\textsuperscript{13} all of which are creations of UK-based artist Sam Battle\textsuperscript{14}

1.2. Connections to Artistic Philosophies

For decades, the practice of circuit bending has served as a method of creating new audiovisual material through the the act of pushing consumer hardware beyond its intended use. The process of recontextualizing common objects into artistic tools and subject matter


draws inspiration from artistic movements throughout the 20th century, firmly connecting the artistic principals of circuit bending to those of previous movements.

1.2.1. Readymades in Visual and Sonic Art

In 1913, Marcel Duchamp’s Bicycle Wheel—a single wheel taken from a bicycle and mounted upside-down onto a stool—served as the first example of his readymades, ordinary and often functional objects that were conscripted into new lives as artistic subject matter through minimal modification. Duchamp continued the evolution of the readymade with his 1917 work Fountain, which consists of a urinal displayed on its back, inscribed with a signature (see Figure 1.3). The process of choosing quotidian objects to be reassigned from their functional purposes into artistic critique and adulation draws comparisons to the early exploratory steps taken in the creation of a circuit-bent instrument; many seek out particular devices to bend due to their recognizable original purpose as well as their inherent sonic or visual generation abilities. The Speak ’n’ Spell—a toy developed by Texas Instruments in 1978 designed to teach phonics and spelling techniques to children—and its related models are highly-sought after source devices for circuit benders in part due to their complex formant synthesis circuitry, allowing users to generate chaotic music out of re-pitched and glitched vocal timbres. Furthermore, Duchamp’s process of exhibiting his readymades in galleries and exhibitions while prescribing them with new titles (abandoning their original labels) also served to highlight the once-ordinary object as a work of art. This same technique has been adopted by circuit benders such as Qubias Reed Ghazala (most notably with his Incantor series of hacked talking toys), A.J. Gannon with his hybrid guitar video game controller/Speak ’n’ Spell instrument the Speak ’n’ Spellbinder (see Figure 1.3), and Nicholas Collins with the creation of his reassembled, hand-spun CD player the Sled Dog. Some elements of the readymade definition show variance with the act of circuit bending. Duchamp’s modifi-

---

cations to his chosen objects were often minimally impactful on their physical construction, leaving the newly claimed artworks to maintain a visible similarity to their original forms. Additionally, the conversion into readymade often cancelled out its useful functionality, allowing it to only exist as an object of art. These two traits in particular do not universally map onto the circuit bending practice; the choice to allow a newly-circuit-bent instrument to retain its original functionality and the degree to which its form is physically modified vary from one bender to the next.

Coupled with their connection to the visual readymade, origins of circuit bending can be found in the practice of experimental musicians and composers who, in the mid-20th century, incorporated readymades in a sonic context. In research centered on the use of readymades as musical tools in the works of John Cage, Charlotte Moorman, and others, authors Peter Bussigel, Stephan Moore, and Scott Smallwood define the context of readymades in performance, stating that "in the context of sonic art, the readymade is either 1) an ordinary object with previously unexplored sonic properties, elevated through an artist’s re-deployment of the object in a way that reveals those hidden capabilities, or 2) an ordinary object whose familiar sonic characteristics, usually regarded as a by-product of its function, are brought into focus through its use." Both versions of the authors’ definition show how circuit bending as an instrument design process results in the creation of sonic readymades. Cage’s exploration of a tea kettle’s resonant properties in *Water Walk* draws connections to Ghazala’s original conception of what he dubbed chance electronics, the evocation of the unrealized sonic potential of consumer hardware through a probing of its internal circuitry.

---


17Ghazala, “The Folk Music of Chance Electronics: Circuit-bending the Modern Coconut.”
Figure 1.3. A comparison of two readymades. Duchamp and Gannon’s modifications to a urinal (a) and children’s toys (b), respectively, transition the items from functional objects to works of art. Art credit to the Marcel Duchamp estate, photograph by sp-Duchamp (a), 2005, and A.J. Gannon, 2009 (b). From Wikimedia Commons website, (a, https://commons.wikimedia.org/wiki/File:Duchamp_Fountain_(28365079).jpg) and file personally provided by A.J. Gannon. Reprinted by permission.
1.2.2. Hacking, Democratization, and Media Archaeology

The vast majority of consumer devices are designed to hide their inner workings from their users. The limited set of buttons, dials, or sensors provided to consumers portray the proper methods of engaging with a device as has been predetermined by its creators. The practice of circuit bending, however, embraces an anarchic approach towards a manufacturer’s wishes; the process of forcibly forging new paths for electricity to flow through, and the provocation and embrace of noise, errors, and glitches stands in direct opposition to the typical relationship between a machine and its user. From the creation of simple, temporary bends to more radical cosmetic and operational modifications, circuit bending’s central conceit aligns with artistic movements that strive to disrupt the power balance between expert technologist and amateur artists by unlocking that which was previously locked, from the inner workings of an object’s electronic components to its historical legacy and role in society.

To embrace circuit bending is to embrace hacking, a defiant act of social activism that serves to expose information or materials that have been purposely kept from the general public. Closed-off and encased in plastic shells, consumer devices present themselves to their users as veritable safes filled with components and information kept out of reach and locked-away by screws and glue. Media theorist Garnet Hertz describes the process of creating technology meant to be utilized but not understood as blackboxing, a purposeful move by corporations to produce technology whose inner workings are to be forever sealed away from the non-expert, causing them to become obsolete over a planned amount of time. This practice has only become more prominent over time, as each passing years sees our daily tasks more and more confined to mobile devices made nearly impossible to open and physically explore.

Breaking through the shell of a machine reconfigures the role of the circuit bender from casual user to hacker as the amateur enters a realm meant only for the expert, liberating the
literal and figurative material hidden within. Furthermore, the process of unblackboxing a device extends beyond the opening of its casing and into the preparation and performing of the now circuit-bent instrument. As the user hunts for bend points with the purpose of forcing glitches in a system or culls bursts of noise from hacked hardware during performance, they expose new audiovisual materials and realize the potential of its circuitry to serve as an tool of avant-garde art, freeing the possible synthesis methods once locked away and walled off by designers and engineers.

The unblackboxing of consumer devices also impacts the fields of technology education and archaeology. As the ethos of creating art through chance electronics spread beyond the works of Reed Ghazala, prominent scholars of the DIY electronic music movement such as Nicholas Collins began to encourage amateur electronics enthusiasts and experimental artists to learn the craft of building new instruments through the hands-on study and disassembly of closed-off hardware. As DIY communities form around the world and the internet continues to serve as a rich source of information for new creators, circuit bending’s prominence at workshops, maker faires, and in academic research allows the art form to serve as form of hardware democratization, creating open-source documentation of once-closed consumer hardware. Schematic drawings and step-by-step instructions illustrating how to replicate popular effects or sources of sonic chaos from well-traversed source devices such as Speak ’n’ Spells and Casio keyboards stand as invaluable resources to first-time benders and provide the only available information about the inner workings of the aforementioned devices outside of company-issued service manuals. Eschewing the need to find physical copies of commonly-bent source devices, research in the field of virtual modeling and creations such as the TR-808

---


Cymbal by Kurt James Werner, Jonathan S. Able, and Julius O. Smith present musicians with new options for recreating the sonic palettes of circuit-bent instruments through easily accessible software.21

Historically, the revitalization of discarded hardware allows circuit benders and audience members of works that utilize their creations to discover hidden narratives, intentions, and ideas surrounding these resurrected devices and their previous purposes. Known as media archaeology,22 this methodology allows for a deeper understanding of the impact dead media and forgotten devices once made on the cultures that created them. To illustrate the ways in which circuit bending shines an investigative light on discarded objects, we return to Garnet Hertz, who, along with Jussi Parikka, points to the circuit bending practice as vital to the transitioning of media archaeology "from a textual method into a material methodology that takes into account the political economy of contemporary culture."23 Through this lens, Hertz and Parikka consider circuit-bent creations as more than instruments designed through exploration of existing circuity, but as zombie media: items once considered obsolete that allow us to form a better understanding of the development of media culture through their resurrection.24

1.3. Notable Works featuring Circuit-Bent Devices

Circuit-bent devices and instruments have been featured in a wide array of artistic mediums in the years immediately following (and in some cases, before) the public introduction of the practice by Ghazala in 1992. To showcase the ways in which circuit-bent devices and their use in performance art have inspired the technological system at the center of this project,

24Ibid.
two representative works—one from the world of electroacoustic concert music and another
from the multi-format visual art realm—will be analyzed. A focus will be placed on the
artist’s choice of source device and its impact on the programmatic nature of the work, their
modification of the device and process for enacting circuit bends in performance, and any
methods in which the work serves to portray the circuit bending practice to the audience in
a manner that extends beyond its typical definition and formats.

1.3.1. Broken Light

*Broken Light* (1992) is a piece for string quartet and circuit-bent CD player, written by
instrument designer and composer Nicholas Collins. In the piece, a string quartet performs
improvised musical gestures (based on graphical shapes and instructions indicated in the
score) in tandem with and in reaction to material generated by a skipping, glitching CD
player reading through a disc of Baroque concerto grossi. Premiering on the 1992 album
*It Was a Dark and Stormy Night*, the piece takes inspiration from the game-centric im-
provisational works of John Zorn and the sample-heavy hip-hop of turntablists such as DJ
Grandmaster Flash.\(^{25}\) Collins previously experimented with using reclaimed hardware as a
sound source in his 1985 piece *Devil’s Music*, in which he uses a system comprised of custom
circuitry, looping and delay pedals, and streaming audio input from a radio to generate com-
plex patchworks of short, asynchronously-repeating samples captured in real time.\(^ {26}\)
Seeking to extend this technique of building sonic collage out of preexisting media through live per-
formance, Collins turned his focus to portable CD players, which he saw as a powerful and
natural replacement to the bulky collection of turntables and records required by DJs at the
time.\(^ {27}\)

---


The hacks undertaken on the Sony D2 Discman CD player in *Broken Light* stem from two goals Collins sought to achieve in order to make the media player more conducive for live performance: a superseding of its small on-board playback control buttons (to allow for easy control of the device by the string quartet) and any modifications that would allow the device to emulate the rhythmic looping, track cueing, and chaotic scratching techniques of live turntable performance. The first goal was achieved by modifying a remote control for the CD player, rewiring its buttons to be replaced by foot switches, which could be more easily engaged by string players while they continue to perform their instruments. Multiple switches were used to replicate the Play/Pause, Forward Seek, and Track Cue buttons on the player in order to create looping patterns, advance from one loop to the next, and to cue up specific tracks on the disc (see Figure 1.4). The second goal required more trial-and-error exploration; seeking to discover what occurred internally when the CD player was paused, Collins procured a service manual for his specific model and tracked down pins on the circuit specified as audio muting pins, used to shut off audio output from the digital to analog converter (or DAC) to the amplifier when the player was paused. By severing the connection between this muting pin and the integrated circuit (or IC) that it controlled, the CD player continued to generate sound from the disc as it read and re-read a small collection of microsamples from the same location on the disc. Collins described the looping material generated from the now unmuted player as being different in rhythmic accuracy from that of a turntable, stating that “unlike the familiar metronomic repetition of skipping vinyl, the paused CD ‘swings’, interrupting its default quarter-note pattern with occasional eighthnote [sic] accents that impart a distinctly ‘musical’ feel to the resulting rhythm.”

The choice of CD player as source device provides us with an insight into Collins’ view of hacked hardware as not solely a means to generate wild and chaotic sound sources, but as a method of creating compositions where the core harmonic structure and sonic palette of the piece are directly dependant upon and stem from the chosen circuit-bent instrument. Collins’

---

28Collins, “Hacking the CD Player.”
bending of the CD player and the resulting musical actions it is capable of (working within the confines of the musical material on the disc) dictated the progression of tonal centers and improvisatory gestures he later composed for the acoustic ensemble. Here, a circuit-bent instrument does not constitute the entirety of the musical material being generated in a performance, nor does it simply serve a supporting role as a unique timbral and tonal sound source; it is equal parts instrument and composer. Furthermore, *Broken Light* represents a shift away from a popular trait of the circuit bending practice: the role of the bender as both progenitor and performer of their custom creation. The journey through chance electronics is often a personal one, leading circuit benders to become the authority on their own creations and the manners in which they can and should be performed. But as part of a larger process to begin removing himself from performances of his own work,[29] Collins relinquished control over the CD player to one member of the acoustic quartet, placing them in the dual role of live bender and leader through the loosely structured formal progression of the piece. By ceding control of the circuit-bent instrument, Collins opens up the door for

---

[29]Duguid. *Interview - Nicolas Collins*
a hacked device to be explored and controlled by a will external from his own, allowing for
a certain degree of collaborative music-making to occur between bent machine and multiple
performers even though broad instructions on when to start, advance, and end loops are
provided in the score.

1.3.2. Glitch Textiles, Year of the Glitch, and DCP Series

*Glitch Textiles* (2012-ongoing), *Year of the Glitch* (2012), and *DCP Series* (2010-ongoing) are
three interconnected projects undertaken by visual artist and musician Phillip Stearns, who
uses circuit-bent visual devices in tandem with additional digital fabrication and manipulation
tools to create his work. In *Year of the Glitch*, Stearns cataloged a new digital image—created
through various processes including circuit bending—representing the aesthetic, technical,
and philosophical concept of glitches and hosted them to a web archive. In his *DCP Series*,
Stearns focused on generating glitch images entirely through the process of circuit bending
digital cameras. Finally, his *Glitch Textile* series took the process of generating images
through glitch into a multimodal realm by using computer-controlled looms to convert his
captured digital pictures into tangible textiles in the form of throw blankets and, through a
later collaboration with Christian Dior, clothing.[30]

An aspect of Stearns’ work centers on finding new ways to embrace, contextualize, and
exploit the concept of glitches—a term commonly used to describe errors, mistakes, or ar-
tifacts made apparent by an improperly operating digital system[31]—in multiple formats of
visual and sonic art. Contextualizing a our modern viewpoint on what a glitch is, Stearns
redefines the word as not simply a mistake or a representation of failure, but as a “slip” in a

---

[29]Images of Stearns’ glitch textile clothing by Dior can be viewed at https://glitchtextiles.tumblr.com/post/127639392194/glitch-couture-is-real-glitch-textiles-for-dior


system, attempted to be hidden from the user but now made apparent through the agency of the artist:

Going back to the Yiddish root of the word, glitsh or glitshn, meaning ‘to slip’ we can sharpen our understanding to refer to some form of slippage or break with the expected. I find that at this point there is a tendency to focus on failure, though personally it isn’t something I find productive because the notion of failure hinges upon similar factors giving rise to the perception of glitches. Slippage obviates a rift between a perceived or anticipated reality and the wild forces latent within world, the very world from which we have fashioned our systems and within which they exist. The rupture was always there, the glitch is that moment which reveals the scope and scale of the rift should we choose to see it.32

Glitches occur quickly and disappear just as fast, existing in a fleeting moment before the systems that created them work to correct what they consider to be a mistake. Stearns feels that through circuit bending, artists can cull glitches from the electrical and algorithmic systems built into consumer electronic devices and observe them not simply as intriguing patterns of color and shapes, but as a manifestation of the cognitive dissonance that occurs when we witness something that does not match our preconceived understanding of reality. To accomplish this, all three aforementioned works feature the use of circuit-bent digital cameras to allow for manual provocation of glitch moments and to capture a record of the momentary artifact that it created.

In hacking digital cameras, Stearns created images from glitches by directly connecting and/or modifying the resistance levels between bend points on the device’s image sensor module. The majority of the resulting images did not involve light taken in through the camera’s optical sensor, and are generated entirely from human manipulation of the device’s ICs and passive electrical components which in turn forced malfunctions and error-correction

executions within the microprocessor’s firmware. Breaking out the camera’s solder points to a breadboard, Stearns created new interface options for connecting and modifying his chosen bend points on the fly and was not constrained by the limited chassis space available on which to affix permanent control elements (see Figure 1.5). Furthermore, the addition of additional passive components and logic gate ICs between his interface elements and the camera’s existing circuitry allowed him to add an additional layer of chance and aleatory into his interactions with the device, building smaller electronic systems that negotiated between his input actions and the actual resulting output events.

Stearns’ works create new avenues within the practice of circuit bending that obfuscate our previously discussed concept of hacked electronic readymade and, at the same time, provide opportunities for media created with circuit-bent devices to exist outside of an instantaneous sound or image. The modified camera resulting from Stearns’ circuit bending experiments, as it is used in these three projects, does not satisfy our understood definition

---

34Zurko, *Glitches Get Stitches: An Interview with Artist Phillip Stearns*. 20
of a readymade in the same way that most circuit-bent musical instruments or live video processing/generating devices would. While it could be used in a live video art performance setting, Stearns places it in the role of a studio tool. By choosing to catalog the images it generates as art in a virtual gallery space, Stearns presents the images as art, but does not extend the same label to the modified device itself.

The resulting representations of glitches that he creates with this camera travel an interesting path to becoming quasi-readymades in their own right: the images that represent glitches occurring inside the hacked camera serve to turn a common error of functional code and circuitry into works of art. The textiles woven from these images have gone on to serve roles both artistic (as displayed tapestries and runway fashion) and functional (as throw blankets and scarves), placing the result of his circuit-bent camera in a nebulous space between all points in transition from object to readymade (see Figure 1.6). Additionally, the media Stearns generates through circuit bending has been presented to and experienced by viewers in more forms than is typically possible with circuit-bent devices. Unlike the music generated by hacked sonic toys or the video projected from hacked visual gear—both of which can only exist in a fleeting live performance form or as an eternal recorded form—the multiformat representations of glitches in Stearns’ series can be displayed online as videos and images, displayed on someone’s wall or couch, and worn on their body, allowing the artist to envision new, multitemporal manners of presenting art made through hacked machinery.

The works by Collins and Stearns discussed here both employ efforts to push the practice of circuit bending out of its comfort zone in technical, performative, and philosophical manners. The choices made by these artists serve to extend the reach of circuit bending as a tool for creation and as an art form to new audiences. By designing collaborative performance opportunities where the uninitiated can control and influence the actions of a hacked device and creating new formats for disseminating media from circuit-bent instruments through
Figure 1.6. Two forms of a glitch generated by and captured through Stearns’ work with a circuit-bent digital camera. Photographs by Phillip Stearns (a, b), 2013. From Phillip Stearns’ personal website (a, b, https://phillipstearns.wordpress.com/projects). Reprinted by permission.
physical and virtual spaces, Collins and Stearns' works both emulate artistic techniques practiced by networked musical performance artists. Using their work as an inspiration, this project aims to create a method in which traditional circuit bending practices can be extended to include the same collaborative, mediated, and web-based interactivity elements native to networked musical performances.
CHAPTER 2. AN OVERVIEW OF NETWORKED MUSICAL PERFORMANCES

2.1. Definition and Practices

The term Networked Musical Performance is contentious. If we consider the first word at face value, then the term could be used to describe any genre, style, or practice of creating music through a collaborative trading of information for the purpose of influencing the actions of one another. This definition certainly encompasses the sonic and aesthetic elements of installations and compositions created in the modern day that label themselves as networked musical performance (or NMP), but it does not highlight the unique use of technological tools that modern NMP artists feature as a means to create interconnected strands of collaborative influence on a collective sonic environment. The inclusion of technology and its importance in multiple stages of NMP composition into our previous definition of the practice opens the door for many new and overlapping definitions to take hold. Looking at the history of connecting computers and/or electronic instruments and/or devices together for the purpose of influencing elements of each other’s performance, the definition becomes increasingly complex as we try to decide if we should begin the definitive timeline connecting to modern NMP practices with the linked microprocessor instruments of the League of Automatic Composers, the participatory radio broadcast experiments of Max Neuhaus, or the Telharmonium performances of Thaddeus Cahill.

In recent research, Eric Lemmon dives headfirst into crafting a definition that acknowledges the long and varied history of networked musical performances. In his paper com-

---


3 Manning, Electronic and Computer Music
paring NMP and the closely-related field of telematic music, Lemmon lands on a definition that comes closest to what the practice currently encompasses, proposing networked musical performance be defined as “a socially constructed term for a musical performance that is conducted through the mobilization of telecommunication, electronic, and electromagnetic technologies to transmit musical signals across networks with more than one node.”

Lemmon’s definition covers significant ground for creating NMP artworks in today’s version of the practice, including nearly all possible technological tools and communication methods. To highlight the elements that are most applicable to this project’s purpose of extending the performative and philosophical aspects of circuit bending, we will focus on highlighting NMP practices and previous works that a) utilize connections between participants both telematically and co-located, b) follow a client-server architecture for handling device connections and negotiating the transmission of data between them, and c) allow for multidirectional and reconfigurable layouts regarding the technical (e.g. the routing of audio streams or control signals) and interaction topologies used to create collaborative works of shared musical influence.

2.1.1. Frameworks for Creating Client-Server Architecture in Networked Musical Performances

A common feature of networked musical performances sees performers, audience participants, and unattended hardware devices linked together through a client-server architecture. In this configuration, a central server (run locally on a computer located in the performance space or hosted remotely on the internet) hosts and manages resources necessary for the performance and distributes them to client devices such as audience participant mobile devices, network-enabled digital instruments and on-stage human or robotic performers when requested.

---


server acts as a unified central node in network topologies, negotiating the paths of control signal data or audiovisual material between client nodes during a performance. The terms of this negotiation are configured in the server code during the composition stage of an NMP work, creating all possible paths of transmission based on the artist’s intended interaction topologies for the collaborative or distributed experience.  

The design of client-server architectures for NMP works has been made easier due to the creation of software frameworks and libraries that use JavaScript to streamline the process of prototyping, refining, and deploying servers and client-side web performance interfaces. Rhizome, created by Sébastien Piquemal, is a framework that supports the building of servers for bidirectional passage of files and control signals between connected client nodes. This framework gives artists the flexibility of communicating between nodes on the network through multiple protocols such as Open Sound Control and WebSockets, enabling for performance interfaces to be built as web pages or with software such as Pure Data and MaxMSP. Rhizome also handles file transfers through a tool that can be run in tandem with the main server, providing a user-friendly method for audience participants to create audiovisual material on their own devices and share with one another through the network. This process is illustrated in Piquemal and Miki Brunou’s work *New Weave* (2015), which sees audience members use the microphones on their mobile phones to record short audio clips, sending them to the on-stage performers for further processing.

The Nexus suite of distributed and networked musical frameworks (NexusUI and NexusHub) by Jesse Allison et al. center on rapid development of web performance interfaces for art works. The NexusUI framework contains universal graphical user interface (GUI) elements

---


such as buttons and sliders along with music-focused options like piano keyboards and tactile audio file waveform scrubbers.\textsuperscript{10} Focusing on communication between interfaces, the NexusHub framework aims to simplify the process of creating control signal channels on both server files and client-side web pages through a practical coding syntax.\textsuperscript{11} In general, the Nexus suite features extensive documentation along with reusable and configurable web page templates, considerations that give first-time coders a means for learning the process of designing NMP architectures and virtual instruments.\textsuperscript{12}

### 2.1.2. Interaction Topologies in Networked Musical Performances

In Roy Ascott’s “Is There Love in the Telematic Embrace?”, the cybernetic artist speaks about the importance of producing art collaboratively through networked connections, focusing on the role of the network itself in shaping and influencing the product produced through it:

In a telematic art, meaning is not created by the artist, distributed through the network and received by the observer. Meaning is the product of interaction between the observer and the system, the context of which is in a state of flux, or endless change and transformation. In this condition of uncertainty and instability, not simply because of the crisscrossing interactions of users of the network but because content is embodied in data that is itself immaterial, it is purely and electronic difference, until it has been reconstituted at the interface as image, text, or sound.\textsuperscript{13}


Ascott puts into focus the crucial role of how, what, when, and with whom we interact with along the many nodes that form performative networks. We can view Ascott’s methodology as a means for not just discussing what types of control data or audiovisual material are being sent across the network’s nodes, but the directions of interactivity being established by the artist when creating possible channels of interactivity and collaboration provided to those participating in a work.

To establish terminology useful for describing flows of interaction and intent in an NMP work, Benjamin Matuszewski, Norbert Schnell, and Frederic Bevilacqua establish a series of interaction topologies, which they describe as “networks of relations between entities (e.g., human, technical artifacts) without any a priori hierarchy on their agencies” and represented as graphical maps indicating the directional flow of actions, intentions, and audiovisual focus/influence. Matuszewski et al. design their graphs and topology layouts to represent the possible methods of cooperative interactivity that exist between nodes in a given NMP work. Assuming the server’s role of as a central negotiator and distributor of anything being transferred, they choose to exclude it as a node from their models and stress that an interaction topology does not always correlate to the network’s technological topology, (i.e. how data or audiovisual material travels through the server). This is done for the sake of better showcasing the intended and resulting interaction from one node to another by focusing on the transmission direction and/or grouping of and eventual recipient(s) of any intentional performance action. For example, a sequenced transmission of an audio file recorded by and then sent from one participant to the next as they stand in a circle would be represented in Matuszewski et al.’s interaction topology graphs with a series of arrows, each one emanating from one node to the next in either a unidirectional or bidirectional fashion (see graphs (b) and (c) in Figure 2.1). These illustrations show the resulting travel of a collaborative action corresponding to the transmission of the file from participant to participant even though

---

it may not be technologically factual; in reality, the technical network topology of a clientserver system would indicate arrows connecting from each node to a server node located in the center of the graph, as each new audio file would need to be uploaded to the server in order to be passed on to the next participant.

Matuszewski et al.’s interaction topologies are as follows:

- **Disconnected graph**: nodes act independently.
- **Unidirectional/Bidirectional graph**: actions cause reciprocal reactions on other nodes in a preset order, progressing through that order in one or more directions.
- **Centrifugal star graph**: actions by many nodes result in reactions from one node.
- **Centripetal star graph**: actions by one node result in reactions from many nodes.
- **Forest graph**: groups of nodes are encouraged to interact together. Their actions can cause reactions from nodes within their specific group according to any of the previous topology graphs.

While these graphs are presented as individual models, one or more of them can be combined to form a hybrid interaction topology. Due to their usefulness describing multilayered pathways for participant actions and collaborations across a network, we will use the interaction topology types designed by Matuszewski et al. in our interrogation of preexisting NMP works and, in later chapters, new works created with Bendit_I/O.

### 2.1.3. Devices for Networked Musical Performances

Networked musical performances typically utilize portable mobile devices such as laptop computers, smartphones, and tablets as performance instruments or receivers of distributed audiovisual materials. These devices have become nearly ubiquitous tools in the NMP practice due to their built-in wireless radios, plethora of on-board environmental and gestural sensors, embedded multi-point touchscreens, microphones, and speakers useful for tactile input and audiovisual feedback, and ever-increasing computational power. Their proliferation
throughout global culture also affords artists the security of knowing that a majority of audience members will have one of these devices on them while attending a performance and they will be familiar with performative actions such as tapping the screen or rotating to match gestural instructions, allowing for large-scale collaborations to occur between performers and novice participants. Featuring mobile devices as audience performance or reception tools allows NMP artists to highlight their prevalent role in our lives in a programmatic manner. In works such as Benjamin Taylor’s *The Last Cloud* (2016), a solo performer creates sonic and visual collages by composing with pieces of web media such as QuickTime videos, streaming MP3 files, and cascading patterns built out of multiple web browser windows. Taylor invites audience members to use their mobile devices to receive a real-time localized stream of his web medium collages, positioning the audience to simulate the act of browsing the web on their personal devices which have now been turned into receptors of high art made with everyday internet media.

Over the past two decades, the creation of network-enabled instruments and controllers has become a prominent subject in digital lutherie. Through inclusion of embedded microcontrollers or palm-sized computers, instrument designers develop brand new tools for interactive and collaborative musical performance across local networks and through the internet that precisely match the performance needs of their creators. These custom devices are designed to communicate with and influence the actions of other like devices or to connect with mobile devices and other web-enabled controllers, allowing for cooperative performances between consumer mobile devices and smart musical instruments, a term coined by researcher Luca Turchet. Taking inspiration from networked consumer hardware, par-

---


allel research fields such as the Internet of Musical Things (IoMusT)\textsuperscript{18} and the Orchestra of Things\textsuperscript{19} have formulated, sharing a focus on studying and creating new devices designed to control, receive, or generate audiovisual material and performance data collaboratively over local networks and the internet. Examples of smart musical instruments (or hybrid systems that utilize them) that fall within these two fields are a system of networked speakers, software instruments, and performance interfaces designed by Stephen Beck and Chris Branton for use in digital music ensembles\textsuperscript{20} the Smart Cajón\textsuperscript{21} of Luca Turchet, Andrew McPherson, and Mathieu Barthet, an electroacoustic percussion instrument that can transmit and receive data between mobile phones in performance; and the Happy Brackets framework\textsuperscript{22} of Oliver Bown et al., a hybrid software/hardware framework designed to program and coordinate the network topologies of Raspberry-Pi-enabled smart musical instruments.

2.1.4. Technological Mediation of Performance Actions within Musical Networks

By stepping back and taking a broader look at networks as a concept—considering the expansive collection of different software and hardware tools that make interconnectivity possible as one large collection with a unified purpose—we can view the existence of telematic and locally-connected links as a technological mediation tools that aid the practice of music performance at large. In works that connect multiple geographically-displaced performers together for the purpose of performing in a shared, virtual concert space, the client-server


\textsuperscript{20}Ibid.


architecture and toolkits used to create collaborative musical networks can be considered a mediation system, viewed as an additional technology helping to negotiate between multiple audiovisual streams in order to make synchronized performances between isolated musicians a reality.\textsuperscript{23,24} Using the definition of networked musical performance that we set above, however, we see the existence of a network not as an additional facilitation tool but as a required environment that encompasses all interactive and aesthetic elements of composition. Within this framing, technological mediation systems can still exist and be stationed between any of the nodes within the web of connected users, negotiating between the performative actions of participants (both human or machine) and any audiovisual reactions.

To explore this concept of intra-network mediation systems, let us look at an analysis on the role of mediation systems in NMP works undertaken by Felipe Hickmann and Rui Chaves.\textsuperscript{25} The authors describe mediation systems placed between paths of material or interactivity as being akin to a window set in the wall of a building. The shape, size, opacity, and location of the window all work to determine what elements of the world outside the building can be seen by those looking through it from the inside.\textsuperscript{26} In this manner, Hickmann and Chaves’ perception frames the purpose of mediation systems embedded within networks as serving “to mediate different levels of access and communication”\textsuperscript{27} by determining the state of any transmitted media or data allowed to pass through the "window" between one node and the next. Graphing their metaphor onto an artistic example, the authors discuss Rob King’s and Pierre Proske’s \textit{Packet Loss} (2010), a work by for two networked Disklaviers.


\textsuperscript{26}Ibid.

\textsuperscript{27}Ibid.
In this piece, a live performer’s musical improvisation on one instrument is attempted to be replicated note-for-note by the other, which is robotically performed by MIDI note and velocity data it receives across the network from the live performance. The attempt at synchronized performance between instruments human and robotic is purposely subverted by the inclusion of a mediation system within the network, taking the form of a software gate that only allows notes played at a velocity stronger than a predetermined threshold to pass through. The stream of MIDI data is therefore filtered, and the unattended Disklavier only reproduces a fraction of the notes performed by the live performer. Using the authors’ window analogy, we can perceive the velocity-gated mediation system as a partially-opaque window, the stream of live MIDI data as our intended view of the outside world, and the impaired performance of the robotic instrument as the manifestation of our impaired view due to the parameters of the window.

Another impact of intra-network mediation systems can be seen by considering the entirety of the network—including its technical and interaction topologies—as an instrument in its own right, one that can be performed collaboratively by all connected participants, performers, and unattended hardware devices. Our intentions of what we would like to accomplish together with and through the network may be stifled by the inherent intentions of the mediated-network-as-instrument itself, a theory proposed by instrument designer Tom Davis in his research on musical instruments and their role as performance mediation systems. Davis builds his theory from the belief of theorist Peter Paul Verbeek, who felt that technological systems and devices are imbued with their own inherent intentions, born from their design and operating process. When interacting with technology, humans are negotiating between the intentions of the device they use and their own intentions of what it should do. Davis expands this point through the lens of Verbeek’s concept of composite intentionally, created when both the performer and the instrument have their own sense of

---


agency and intent. "In this conception," says Davis, “the instrument is not transparent, it does not disappear in use, the performer is not trying to master it; they are collaborating with it; adjusting and shaping themselves in relation to it. This wider conception of musical instrumentality relates to concepts of mutable boundaries between performer and instrument." Viewing "instrument" as synonymous with our description of a network with embedded mediation systems, Davis’ description of an instrument as a collaborative force with which performers are in a continuous negotiation with embraces the collaborative and transformational aspects of mediated interactions possible through networked musical performances. As the technologies used to create systems for mediation become more powerful, artistically aimed, and easier to access through existing NMP technologies artists will continue to be presented with creative possibilities for designing systems that modulate intent and interactions.

2.2. Collaborative and Distributed Works through Networked Performance Technologies

To showcase the ways in which the aesthetic and technological aspects of networked musical performances have inspired the system at the center of this project, representative works that address the art form’s flexible approach to building collaborative and mediated environments for linking performers together will be analyzed. The artist’s choice of interaction topologies, mediation systems, networked hardware, and audiovisual creation through collaboration will be highlighted for each work discussed.

---


31Examples of technologies that meet these goals are web-based machine learning tools such as those created in the MIMIC Project (https://mimicproject.com/about) and Sentiment Analysis tools as featured in IBM Watson’s Tone Analyzer service (https://www.ibm.com/watson/services/tone-analyzer/), both of which can be integrated into web performance interfaces through JavaScript libraries and are accessible through the internet.
2.2.1. Imaginary Berlin

Anna Xambó’s *Imaginary Berlin* (2018) is an interactive work for mobile phones, streaming audio, computer, and audience participation. The piece is inspired by John Cage’s 1951 work *Imaginary Landscape No. 4* in name, sonic context, and process for collaborative music making. In Xambó’s piece, a solo on-stage performer creates twelve channels of audio comprised of field recordings and radio station samples recorded in Berlin, the site of the piece’s premiere performance. Audience members are encouraged to use their mobile phones to pick from one of the twelve audio channels, which are being streamed wirelessly to client-side performance interfaces running in their web browsers. To choose a channel, participants move their phones horizontally through the space in front of them. The on-stage performer has designated invisible, segmented spots in this empty space as points where localized playback of each audio stream will occur on the participant’s phone; a full sweep from left to right allows the participant to shift through all possible audio streams as if changing the channel on a radio, alternating between whichever ones they prefer. In addition to the sounds emanating from every audience phone, the solo performer’s laptop feeds every audio stream into a web-enabled software mixer and out to the concert hall sound system. Through vertical movement of their phones, each user can contribute to the overall presence of their selected audio stream in this master mix.\(^{32}\)

Just as in Cage’s piece, the core focus of interactivity in *Imaginary Berlin* stems from a sense of individual activity that overlaps into group participation and influence. Xambó provides no score to her audience performers, so all actions taken represent their individual intents. Sounds are not distributed to one phone at a time in a pre-determined sequence, and it is up to each audience member to choose which audio stream they will listen to and for how long they listen before moving onto the next. Forgoing a pre-sequenced order of sound distribution by giving complete performative action to the audience matches Matuszewski

et al.’s disconnected interaction topology (see Figure 2.1), with the overlapping layers of sound from one phone to the next unmodulated by and independent from the actions of their neighbors. We can also overlay both the centrifugal and centripetal graphs on to this topology by changing our perspective as to who in the network is the central node. Looking at the interactions from the on-stage performer distributing a set of invisible audio stream options for each disconnected user to choose from, and, from the perspective of the web-enabled audio mixer, the interactions from each phone meant to adjust the presence of selected audio streams.

Xambó’s network topology offers a mediation system in the form of the on-stage performer’s placement of the stream shift points along the space in front of the participants. These points are invisible, and the audience has to work to explore the space in front of them to find a stream they enjoy, and to remember where to find it later if they would like to hear it again. There is an interesting balance of chance at play in Imaginary Berlin that is different from the Cage work that inspired it: the predetermined audio streams chosen by the solo performer remove the opportunity for participants to find audio elements that are completely presented to them at random, a possibility when using a transistor radio to pull from a wider band of uncontrollable public radio sound sources. In this manner, the use of mobile devices trades the expressive physical input gestures and high fidelity of local playback possible with modern smart devices acting as radios over the complete randomness achieved with actual radios.

2.2.2. Diamonds in Dystopia

Diamonds in Dystopia (2015) is a networked exercise in building a collaborative poem with audio visual accompaniment, created by Derick Ostrenko, Jesse Allison, and Vincent Cellucci. The work is scored for two performers, poet, live electronics and visuals, and audience

\[^{33}\text{Matuszewski, Schnell, and Bevilacqua, “Interaction Topologies in Mobile-Based Situated Networked Music Systems”}^3\]
participation through mobile devices. Throughout the piece, a bed of live audio is performed while a poet reads from a pre-written "seed" poem inspired by transcripts of TED conference talks. As the performance progresses, the seed poem is presented to the audience participants through their web performance interface and on a projector screen in the front of the hall. Participants tap on words of their choice, causing a localized audio file of their selection to be "spoken" from their phone’s speakers. This results in a real-time change to the seed poem for all parties, including the poet. A mediation system that relies on Markov chains uses the audience-chosen words as search terms to riffle through a database comprised of thousands of previous TED Talk transcripts, collecting lines of text that match the terms and joining them together into new stanzas. A group of newly-generated stanzas are then sent to the poet, who in turn chooses which of the options they would prefer to read and selects it as their new seed poem. This process continues until the piece’s end, with the original seed poem eventually lost entirely through the collaboratively-generated writing process.\(^{34}\)

At first glance, the interaction topologies of *Diamonds in Dystopia* match those at work in Xambó’s *Imaginary Berlin*: localized audio playback triggered by tapping text results in unsequenced, disconnected interactions between audience participants; text choices sent to the mediation system and new stanzas sent back to the audience match the centripetal and centrifugal star respectively. One pattern of interaction that can be added to this overlay is revealed when we look at the work’s action from the perspective of the poem. Starting at the poet and audience simultaneously, the seed poem is altered when the mediation system sends new stanzas for their choosing. Once chosen by the poet, the new seed poem is sent back to the audience, and the process is repeated. This continuous progression from poet/audience, to mediation system, then back to poet matches the unidirectional circular topology, representing a closed loop of continuous evolution and total of three distinct interaction topologies at play in the work simultaneously.

\(^{34}\)Jesse Allison, Vincent A Cellucci, and Derick Ostrenko, “Creative Data Mining Diamonds in Dystopia: An Interactive Poetry Web Application,” *Journal of the New Media Caucus| ISSN*, 1942, 017X.
2.2.3. Patchwerk

In Brian Mayton et al.’s Patchwerk broadcast (first streamed 2012), audience participants are able to log onto a server from anywhere on the planet and collaboratively perform the Paradiso Analog Modular Synthesizer (PAMS), a vintage 1970s synthesizer created by Joseph Paradiso.\textsuperscript{35} Using a web page performance interface, small groups of geographically displaced users collaboratively perform the PAMS (located at MIT Labs in Massachusetts) by engaging with HTML5 buttons, potentiometers, and switches that emulate physical controls found on the physical synthesizer. Manipulation of their web interfaces generates control signals routed through a remotely-hosted server and on to the Patchwerk, a custom-made module that translates the incoming actions into control voltage for use in performance of the PAMS.

The server used in the Patchwerk broadcasts mediates interactions with the PAMS by distributing control of the modules’ inputs and outputs to only a handful of users at a time. Upon logging on to the server, users enter a queue to wait for their turn to perform. After a set amount of time has elapsed, a new group of performers are allowed to take control. To share the resulting audio of performances created with their networked system, the Patchwerk module includes internal hardware for capturing and converting live audio streams into Ogg Vorbis format, preparing it to be streamed through the internet to any connected listeners and the performers.\textsuperscript{36}

Performances with the Patchwerk/PAMS system follow one of a handful of Matuszewski et al. interaction topologies, some becoming a better fit than others based on the perspectives we choose to look at the interactions from. If we consider the performance from the perspective of the audience participants, the work follows a centripetal star topology, with each performer node sending intended interactions to the Patchwerk module, acting as a


\textsuperscript{36}Ibid.
central node receiving those interactions. But, if we consider the work from the perspective of the Patchwerk, the centrifugal star topology would best fit the process of streaming audio data to each listener’s device as an aural feedback measure of their performative actions with the PAMS. Another possible topology would be an overlay of the forest, centripetal, and centrifugal graphs, which best describes the occurring interactions when the performance is looked at with an all-encompassing perspective: since the server only allows for small groups to perform the PAMS at a time, the sub-groupings shown in the forest graph would best represent a limited set of nodes transmitting their actions to the center Patchwerk module, which in turn sends audio feedback out to every node, whether they are performing or not.

Mayton et al.’s creation of a networked performance environment with the Patchwerk module displays a unique approach to using smart musical instruments that straddle the line between the past and the present. Patchwerk is a contemporary creation made to look like and interact with a vintage piece of music technology, making it an IoMusT-centered remake of a device that looks as if it was repurposed. In a way, the device explores the history of modular synthesizers through an aesthetic approach to media archaeology instead of through the practice-based approach that Hertz and Parikka attach to the act of circuit bending. While it was made to interface with a piece of vintage hardware, neither the Patchwerk module, nor the PAMS, would classify as an example of zombie media since they are not reanimated or repurposed devices. They do however open the door to using NMP and mediated performance techniques for the purpose of hardware democratization by providing free, global access to an expensive and sought-after electronic musical instrument.

The options for interactivity and collaboration made possible by networked musical performance techniques provide a road map towards bringing the same abilities to circuit-bent devices. Modeling the technologies and architectures discussed earlier in this chapter in the design of Bendit_I/O’s software components, a synthesis of both artistic practices will al-

---

low artists to create new experiences between performers, audience members, and hacked consumer devices.
CHAPTER 3. A CASE FOR NETWORKING CIRCUIT-BENT DEVICES IN PERFORMANCE

In *Chorus for Untrained Operator* (2016)—an interactive installation by Peter Bussigel and Stephan Moore—participants enter a room where dozens of everyday objects such as oscillating fans, acoustic toy instruments, mechanical alarm clocks, 8mm film projectors, disco lights, and antique telephones lie placed around the space. Each object is connected to a large mid-century telephone operator switchboard positioned in the center of the room and connected to the aforementioned objects through a web of long cables. This ever-evolving collection of common objects (refreshed and reorchestrated with different items between each installation) serves as a mechanical chorus, waiting for participants to bring to life various combinations of ringing, clicking, whirling, and sputtering voices by picking up a patch cable and using it to connect various points on the antique switchboard. When a connection is patched, an object bursts into action from a once dormant state, and further patching reveals new outbursts of reanimated reaction from the once discarded items, collected by the artists from local thrift shops and junkyards (see Figure 3.1). As they explore, participants can fill in blank scoresheets to notate interesting patches they have discovered, or collectively improvise new voicing by working together in groups. Using the limited tools provided to them, participants can create layers of hardware-produced sonic material, crafting a new rendition of the musical work through networked performance of readymade objects with each improvisatory interaction.\(^1\)

While *Chorus for Untrained Operator* does not fall squarely within the realms of circuit bending or networked musical performance, it does lie somewhere in-between. While some devices have been internally modified in order to adjust their pitch, no drastic subversion of an object’s operating process on the level of traditional circuit bending occurs. Physical actions are undertaken in order to perform each piece of hardware, but the unlabeled

\(^1\)Bussigel, Moore, and Smallwood, “Reanimating the Readymade.”
Figure 3.1. Performative readymades in Peter Bussigel and Stephan Moore’s *Chorus for Untrained Operator*. Photograph by Peter Bussigel, 2016. From Peter Bussigel’s personal website, https://www.triangleline.com/chorus. Reprinted by permission.
switchboard presents something closer to a mediation system than the explicitly mapped performance interfaces typically found on circuit-bent instruments. Without a diagram from a previous performance, participants must discover which connections bring which machine to life, negotiating with a system that mediates their actions with a level of ambiguity and confusion. Most interestingly, while the piece suggests an established network built between devices, there is none. The piece does not provide ways for one readymade to influence another. Electrical signals generated by a sewing machine, for instance, do not affect the musical output or operation of a toy xylophone through either a sharing of data or electrical pulses through the patch bay connections or through exertion of outside mechanical force. The artifice of machine-to-machine interaction is in reality a network of influence existing between and created through the actions of the audience. In reaction to the audiovisual output of the devices, the participants build channels for shared control data for the purpose of performing the room-wide readymade chorus between themselves, transmitted along channels made of verbal communication, written notes, and drawings.

This ability for readymades to influence the actions of performers and of performers to control readymades through collaborative and network-influenced methods is what distinguishes Bussigel and Moore’s work as an example of how the practices of circuit bending and NMP can and should be synthesized together. Through the use of the Bendit_I/O framework, artists can investigate the questions involved in the intersection between these two practices. How can the philosophical and audiovisual elements of circuit bending be further explored when hacked devices are able to be performed through performance interfaces that replace directly-mapped buttons and dials for those that can be ambiguous, intangible, or exist in multiple mediums and formats? What aspects of the pre-established relationships between performers, makers, and witnesses of circuit-bent instruments can be enriched and recontextualized when these instruments have the ability to influence the performative intentions of one another?
Pieces such as *Chorus for Untrained Operator* and those discussed in previous chapters, along with the potential for creating new works that address the questions posed here serve to make the case for networking circuit bent devices in performance. Before presenting the inner workings of the Bendit_I/O framework and the compositions created through its use, a further look at its potential impacts on elements of NMP and circuit bending practices will be undertaken, highlighting existing works that have made use of similar technological and methodological concepts related to this research.

### 3.1. Networking the Circuit-Bent Readymade

#### 3.1.1. Expanding Performance Interaction Options

Following traditional circuit bending practices, a majority of performance interface elements added to circuit-bent devices require tactile engagement from the performer. All of these interface elements require the performer to either be within arms reach of the instrument or to run cabling between the instrument’s bend points and a separate control box. Interface elements are explicitly mapped and typically done so in a one-to-one relationship where each parameter or action is controlled by a single interface element. Examples of this mapping approach would be the temporary opening and closing of a single switch or button that connects two bend points together to toggle between a clean and distorted tone color, or a single potentiometer whose variable resistance level is used to modulate the speed of rhythmic noise bursts.

Modifying a traditional circuit-bent instrument to be controlled across a wireless network would extend the options of possible performance interfaces available for engaging with the instrument as well as remove the requirement to be nearby or physically-tethered to the de-

---


vice. Network-enabling a circuit-bent device in this manner opens the door for performance through the use of sophisticated sensors, such as multi-axis accelerometers, remotely-located atmospheric sensors, and audiovisual-reactive triggers; multi-touch interfaces such as mobile phone or tablet screens; streams of data culled from social media, open APIs or databases; and gestural and biometric controllers such as the Myo armband. The inclusion of additional scaling software or mediation systems in the control chain between sensor/performance interface and networked circuit-bent device allows for performers to design more complex mapping relationships between interface element/data source and the device’s audiovisual generation parameters.

3.1.2. Expanding the NMP Device Set

Networking circuit-bent devices allows for an expansion of the traditional device set used in works featuring networked performances systems. Integrating the often unpredictable and device-specific synthesis processes of circuit-bent readymades into the realm of NMP allows artists to adopt the distinct sonic properties of vintage toys, analog media players, and common household appliances into primarily digital soundscapes. Artists can use familiar NMP systems to distribute or share performance data between client participants, digital synthesis engines built with tools such as the Web Audio API, and unique circuit-bent instruments in a collaborative setting.

Networking circuit-bent readymades not only expands the sound palettes available to NMP practitioners, it shines a light on our increasing move away from interactions with tangible objects towards their disembodied software facsimiles or internet-controlled doppelgangers. This migration to advanced versions of everyday objects can be addressed in works through a blending of digital tools with distinct or nostalgic analog hardware fluctuating

---

between disembodied and embodied portrayals of a sound-making process. An example of this concept realized would see audiences controlling a group of unattended circuit-bent tape players through a web page performance interface, dragging their fingers across a waveform UI element to move forward or backward through the audio on each physical cassette tape. While the control interfaces are software-based and ethereal, the sound engines are physical, serving as embodied representations of their disembodied digital interfaces.

3.1.3. Mediating our Interactions with Circuit-Bent Devices

The wide range of software tools available for building intra-network mediated systems can also be employed in performances of networked circuit-bent instruments, negotiating between the performer’s input actions and the resulting engagement or modification of a device’s bend points. Circuit bending artists such as Phillips Stearns incorporate hardware-based mediated systems in their use of circuit bent devices, designing logic gate circuits that sit between tactile performance interface elements and the bend points of their devices. Here, added elements of programmable gating result in an instrument that responds differently to Stearns’ actions based on whether or not those actions meet the circuit’s conditions. By moving conditional barriers and logical gates into the software realm, performers of networked circuit-bent devices can include more complex and adaptive performance analysis procedures through neural networks or supervised machine learning models into their control chain. Through the use of frameworks such as Google’s Magenta or Rebecca Fiebrink’s

---


mediated systems that allow for a procedural mapping process can be incorporated in performance of circuit-bent instruments.

3.1.4. Collaboration between Multiple Circuit-Bent Devices

Using a client-server architecture, the ability to create collaborative performance environments amongst participants can also be extended to networked circuit-bent devices. Audiovisual material and control data can be both received and generated by a circuit-bent instrument, as well as passed along to an additional smart musical instrument or mobile device. Following the topology graphs designed by Matuszewski et al., a server can be used to design paths of interaction between circuit-bent devices, routing incoming data from a pre-determined group of instruments organized by source device type or timbral similarities. By choosing which individual device or additional group of devices that data ultimately gets passed to, the server can allow all instruments in that sub-group to trigger the same bend simultaneously or to act independently from one another, randomly choosing how to react based on the shared message. These suggested use cases illustrate the point that through network-enabling, the role of active interaction coming from performers and audience participants in a piece utilizing circuit-bent devices can be scaled and mitigated to various degrees, even to the degree of removing their influence entirely. Orchestras of independently acting circuit-bent toys, driven by their own agency, or a trio of collaborative circuit-bent toasters creating a generative composition based on patterns they each generate and share between each other, are examples of works possible through the networking of circuit-bent devices.

---


9Matuszewski, Schnell, and Bevilacqua. Interaction Topologies in Mobile-Based Situated Networked Music Systems.
3.2. Media Archaeology and Hardware Democratization through Net-worked Performance of Circuit-Bent Devices

With the use of web interfaces and internet data sources, interactions between users and circuit-bent devices can be enhanced by presenting a device’s historical background through new audiovisual means. A previous work that investigates the pasts of found objects is *The Bureau of Suspended Objects*, an ongoing residency and installation by media artist Jenny Odell. In Odell’s work, a large collection of reclaimed consumer devices are collected in a gallery space and labeled with QR codes. When scanned with a smart device, each code presents audience participants with a collection of historical information about the device, including a list of materials used in its assembly, a location of the factory of its origin, and archival videos and commercials from its initial release.

Supporting Hertz and Parikka’s idea of circuit bending as a practice-based approach to media archaeology, the networking of circuit-bent devices can result in new methods for dispersing information about the readymades used in a work of art to audience performance interfaces. Through real-time generative web art based on historical data or sonic, textual, or visual archival quotes presented through custom apps, media archeological actions could be integrated into works that further unblackbox circuit-bent readymades through the use of notoriously blackboxed smartphones and tablets, themselves now repurposed as tools for media culture exploration. Synthesising concepts from Odell’s *Bureau* and Bussigel and Moore’s *Chorus*, participants could use their smart devices to bring circuit-bent readymades back to life and usher them through an accelerated timeline of their operational lifespan from consumer object to circuit-bent instrument; as a user drags their finger along a graphical timeline from the devices conception to its point of media zombification, pairs of bend points inside the device can be gradually connected, resulting in more and more chaotic audiovisual generation.

---

Figure 3.2. Discarded objects archived in Jenny Odell’s *The Bureau of Suspended Objects*. QR codes link to historical and archival audiovisual material catalogued items. Photographs by Jenny Odell (a, b), 2015 and screenshot by author (c), 2020. From Jenny Odell’s personal website (a, b, http://www.jennyodell.com/bso.html) and BOSO online archive (c, https://www.suspended-objects.org/). Reprinted by permission.
Circuit bending as a practice, a methodology, and a performance medium can be democratized in ways that stretch beyond existing steps taken by the open-source distribution of techniques and schematics or the organization of communal festivals. While telematic technology has allowed for globally-hosted workshops aimed at teaching circuit-bending techniques, the network-enabling of hacked instruments could allow for collaborative performances across the world. The democratization of hardware through circuit-bending practice can now exist not just through the sharing of educational info and experiences, but through real-time circuit bending preparation, exploration, and performance. Teams of benders can share control of a single device: while one participant probes disparate points on a device’s circuit board, a remote participant can trigger the connection of those points from afar, working together in a performance while physically separated by oceans. Those without soldering skills and those with limited physical mobility could still explore the art of chance electronics by utilizing performance interaction methods or environments—such as augmented or virtual reality spaces—that are more accessible than buttons, switches, dials, or tools for electronics modification. Furthermore, these techniques can be used to provide outlets for those wishing to explore older hardware but unable to due to limited or no access to specific source devices for circuit bending. In areas of the world where older consumer devices from Western cultures are scarce or expensive, a system that enables networked circuit bending opens the door for cooperative experiences between practitioners through outlets for exploring the artistic practice in hybrid realities.

11Hertz, “Art after New Media: Exploring Black Boxes, Tactics and Archaeologies”
CHAPTER 4. BENDIT_I/O

Bendit_I/O is a framework that extends mediated and networked performance techniques to circuit-bent devices in order to enhance the means in which performers and audience participants interact with hacked hardware in audiovisual artworks. Composed of both hardware and software components, the framework translates control signals sent over a wireless network into connections and adjustments of resistance between bend points on a circuit-bent instrument. This replaces physical interactions between a performer and their circuit-bent device, opening the door for mapping complex physical gestures, data sources, or artificial intelligence systems to assume control over a hacked readymade.

Bendit_I/O is designed to embrace the various interaction topologies of networked musical performances. Once properly augmented with the framework’s hardware component, a circuit-bent device can be treated as a node along an network, effectively turning any consumer device into a smart musical instrument. By merging the aesthetic values of circuit-bending and NMP together, it stands as a system aimed at helping artists from either practice design works that examine our human connection to the devices that permeate our past and present, facilitating connections between electronic readymades, human users, and digital tools. The framework’s software tools can be downloaded at [www.benditio.com](http://www.benditio.com).

4.1. Design Criteria

Early in the process of developing Bendit_I/O, the following design criteria were established in order to create a framework that practitioners of various skill levels in the fields of hardware-hacking and NMP could utilize:

- All components of the Bendit_I/O framework (both hardware and software) would be designed with modularity in mind; both the software and hardware components should include “basic” versions that include a core set of functionality as well as provide users
with the ability to expand that functionality through the addition of extra hardware or additional code libraries.

- The framework should encourage new creative explorations through a merging of the circuit bending and networked/distributed art practices without purposely limiting or diminishing key aesthetic elements of either practice.

- To facilitate ease of use and promote a user-friendly methodology, a clearly documented set of user guides—including a software library reference page and application programming interface (API)—and examples of practical use cases for the Bendit_I/O framework would be developed and posted online for users to access.

- To model the wide variety of platforms/devices used by both hardware hackers and network artists, the Bendit_I/O framework should be compatible with preexisting audiovisual software environments in order to encourage integration of hacked hardware devices into artists’ current workflow.

In order to complete these objectives, the following steps were undertaken:

- Examination of previous works in the fields of circuit bending and NMP in order to determine crucial design criteria and to best understand, replicate, and facilitate through the Bendit_I/O framework the methods undertaken by practitioners during their creative process.

- The development of the framework’s hardware component, “plug-and-play” circuit that adds networked capabilities to attached circuit bent devices.

- The development of the framework’s software components, which consist of user-friendly JavaScript libraries that facilitate the creation of Node.js servers (for networking devices, performers, and audience participants) and client-side, web-browser-based performance interfaces.

- The writing of an API to serve as reference documentation explaining the core functionality of the Bendit_I/O framework’s software tools, providing users with information on how to use both the client-side and server-side libraries.
• The creation of three compositions, each of which utilize hacked hardware devices, web-based performance interfaces or interfaces not commonly used to perform circuit-bent instruments, and NMP interaction topologies, in order to exhibit Bendit_I/O’s ability to extend mediated and networked performance techniques to circuit-bent devices.

4.2. System Components — Hardware

To enable a circuit-bent device to be performed through the Bendit_I/O framework, users augment their device with a Bendit board (see Figure 4.1). Once the ideal bend points on a device have been chosen, users solder the bare ends of ribbon cables to these points instead of connecting them to push buttons, switches, or dials.

Using two separate ribbon cables—one for enacting direct connections between bend points and a second for points connected through variable resistance levels— a user then connects their hacked device to a Bendit board’s multi-pin header terminals (see Figure 4.2). Once wired, the circuit-bent instrument is performed through interaction with the Bendit_I/O software tools.
To replicate the action of push buttons and switches, Bendit boards contain a series of reed relays for making or breaking an electrical connection between bend points. An digital potentiometer IC generates electrical resistance in the range of 0 to 100k ohms between bend points and is used to replace physical rheostats or potentiometers. For driving a device’s built-in motors or engaging objects that must be moved physically (such as the pushing or contracting of a tape head on a VCR or tape player), Bendit boards also contain a dual-channel motor driver IC capable of controlling direct-current motors, solenoid actuators, or stepper motors. A multicolor LED is used to indicate important status changes, such as when the board is searching for a Wi-Fi signal or when the board has connected to the Bendit_I/O server.

To communicate across a wireless network, Bendit boards contain an ESP32 microcontroller unit, which includes onboard Wi-Fi and Bluetooth radios, multiplexing abilities on the majority of general purpose input/output (GPIO) pins, an 8-bit resolution multichannel

---

digital to analog converter (DAC), and a 12-bit resolution multichannel analog to digital converter (ADC). Support for ESP32 chipset is well documented in the physical computing and Internet of Things communities and code for the module can be programmed through an Arduino IDE tool chain. Each Bendit board’s ESP32 module is mounted onto a smaller development board manufactured by a third party, which is in turn connected to the Bendit board’s base PCB through male and female headers. The decision to use commercially-available ESP32 development boards instead of the bare ESP32 module was made in order to maximize assembly time (due to the inclusion of necessary components such as voltage regulators, a USB-to-UART serial bridge IC, a micro-USB port, and useful power conditioning and signal filtering circuits that are included on these prefab boards) and to provide users with a quick way of replacing the most complex component of their Bendit boards if the need may arise. The inclusion of serial-data-converter circuitry built into ESP32 development boards provides users with an easy method of flashing updated firmware to their Bendit_I/O-enabled creations.

4.2.1. Hardware Revisions and Expandability

The first prototype of the Bendit board was created in November of 2018, followed by the design and manufacture of version 1.0 in January of 2019 after rigorous testing. The version 1.0 hardware is designed to fit a specific ESP32 development board (the Adafruit Feather HUZZAH model) and it includes six reed relays, six digital potentiometer channels, and two motor driving channels. Following months of performances with the framework, work began on a new design for the Bendit board, focusing on expanding the board’s feature set and optimizing its use of the ESP32’s GPIO pins (see Figure 4.3). Significant hardware changes in this newest revision include:

---


• Reconfiguring the board design around a more generic, less expensive model of ESP32 Development board (the DOIT ESP32 Dev Kit) that provides access to more of the ESP32 module’s GPIO pins.

• Replacing the analog RGB LED with a digitally-addressable WS2812b LED.

• Incorporating a shift register IC into the design in order to drive the onboard reed relays.

• Increasing the number of on-board reed relays and digital potentiometer channels to eight.

• Providing users with multiple voltage options for the digital potentiometer IC’s power supply lines in order to replicate voltage divider circuits on a wider range of circuit-bent devices.

• Adding a second screw terminal for easier access to the second motor channel.

• The inclusion of additional “breakout” terminals providing optional access to the board’s DAC, ADC, and I²C channels, along with unused GPIO pins.

These changes result in a Bendit board that better satisfies the design criterion of an expandable, flexible, and modular approach to all components of the framework. Changes made to the reed relay and digital potentiometer portions of the board allow users to extend the number of possible bend points they can manipulate on a circuit-bent device; by using the Switch and Pot extension terminals, additional relays and digital potentiometers can be added to and controlled by a single Bendit board. Access to additional i/o types such as the DAC provides users with the ability to control a circuit-bent device through voltage changes generated from the attached Bendit. Using the I2C breakout section, an external ADC chipset can be added, providing higher-resolution readings of analog signals generated by a circuit-bent device. Sending or receiving this data to and from the server allows it to be treated as control data and provides new methods interacting with a hacked device in performance.
Figure 4.3. A v2.0 model Bendit board.
4.2.2. Bendit Board Firmware

The Bendit board firmware is written in the Arduino-compatible hybrid C++/C language, utilizing Espressif’s Arduino Core as a development toolchain. The firmware code makes use of the ESP32_AnalogWrite[4] Ticker[5] and Socket.io-Client libraries[6]. Before using their board in performance, users are required to configure the board’s profile and connectivity information by modifying the firmware code, designating the network name and password of their performance Wi-Fi network, the address and port of the Bendit_I/O server, and profile information consisting of a board number (used to match with the virtually-created BenditDevice object created in their performance interface web page), an LED color, and a device nickname (used to refer to the Bendit board by name matching the circuit-bent device it is attached to, e.g. “CD_Player”, “Casio_Keyboard”, etc.). This data is then written to a text file that is stored on the ESP32’s flash memory.

Upon power up, each Bendit board searches for and connects to the designated Wi-Fi network and server. On successfully connecting to the network, the boards open a socket connection and emit a "handshake" event listener, passing along their assigned board profile data to be stored on the server and associated with their unique socket. As this handshake event channel is unique to Bendit boards, it allows the server to differentiate between connections made by web interface clients and Bendit boards and to catalog them accordingly. Users can read and rewrite a board’s original profile data from their connected web performance interface for easy reconfiguration before and during performance.

---


4.3. System Components — Software

The software components of the Bendit_I/O framework consist of JavaScript libraries designed for building servers and client-side interactive and reactive performance interfaces. Both libraries utilize Socket.io—a framework built on top of the standard WebSocket protocol that creates bi-directional data communication between client web pages and a server. This communication is event driven, and on connection, each client and Bendit board are assigned a collection of event listeners that are triggered or reacted to based on the control data being received or sent.

4.3.1. The Server

Bendit_I/O servers are built using the benditHub.js library, which was written with Node.js, Socket.io, Express.js, and NexusHub package. A Bendit_I/O server can be run locally on a performer’s machine or hosted remotely via a Node.js hosting service such as Glitch, Heroku, or Amazon Web Services.

In performance, the server creates a new Socket.io socket each time a web client or a Bendit board connects. The server is designed to differentiate between connections made by client web performance interface users and those made by Bendit boards; on connection, Bendit boards emit a unique handshake event that web client users do not. The server responds accordingly based on the type of connection it receives, collecting the socket ID for all client web performance interfaces and generating a packet of Assigned Board Data—comprising of a color choice for the board’s LED and a Board Number generated sequentially by order of connection—for every connected Bendit board. Lists of the aforementioned client and Bendit

---

9“NexusHub: Nexus Distributed Performance Framework.”
board data are maintained by the server and accessible by performers using the client-side Bendit_I/O library discussed later in this chapter.

The core of the bendHub.js library consists of a single class named *BenditHub*. This object includes the following methods:

- **initServer**: starts the server by initializing a new instance of NexusHub, Socket.io, and Express, hosts all webpages and media files required for the performance, and mounts the Socket.io instance to the internal Node.js HTTP server. This function also creates the core series of event listeners on each newly opened socket when a user of Bendit board connects.

- **postStats**: logs information about the server and its connected users and Bendit boards to the console based on the argument passed in when called. Possible information that can be logged includes the server’s port number, the version of the benditHub.js software library used to create the server, and the total number of currently-connected Bendit boards.

- **setAdditionalChannels**: extends the core set of Bendit_I/O event listeners created for each socket on connection. This is only available for event listeners with actions pertaining to webpage interfaces or reactive media and cannot be used to add additional event listeners to the firmware of connected Bendit boards.

The server library strives to simplify the process of designing networking applications by reducing the amount of code one needs to write. This allows artists to create a standard server file template that provides the core functionality of the framework and can be reused without significant modification for each new Bendit_I/O composition. If additional server functionality is needed—such as the creation of new, performance-specific event listener channels—it can be added to the standard server template through a single function call after the server has been initialized. Listing 4.1 shows the code required to create a simple Bendit_I/O server file using the benditHub.js library.
4.3.2. Client-side Library

Bendit_I/O has a client-side JavaScript library that is used to create interactive interfaces for performers and reactive media for audience members with a web browser. As with the benditHub.js server-side library, this client-side toolkit—named benditBrowser.js—is designed to aid artists in communicating with networked circuit-bent devices and connected participants in a performance using the Bendit_I/O framework. To simplify the coding process, the library’s syntax aims to be user-friendly and models function names after traditional, tactile actions of circuit-bending performances (e.g. flipping a switch, setting a motor speed, setting the position of a potentiometer). To keep track of which circuit-bent device is being controlled, artists assign variable names to the virtual representations of their hardware. This allows for easy and straightforward addressing of each hardware device in performance.

To create a web-based Bendit_I/O performance interface, the benditBrowser.js library must first be referenced in the head of the web page’s HTML file. The benditBrowser.js library consists of two primary object classes: Bendit and BenditDevice. When the page loads, a global instance of the Bendit class is defined, establishing a connection with the Ben-
dit_I/O server. Internally, the constructor achieves this by pointing the library’s Socket.io instance to the web address and port of the server hosting it, bypassing the need for the user to update their code if they decide to host their server remotely after initially hosting it on a local machine, or in the case that they move their server to be hosted at a URL different from one used in a past performance. The global Bendit instance includes three arrays: one containing the socket IDs of connected web interface clients, another containing a collection of board data for all connected Bendit boards, and a third array containing all virtual BenditDevices that have been created and assigned to a corresponding Bendit board. Additionally, the Bendit instance includes the following methods that collect, list, and/or create the aforementioned data:

- **getConnectedBenditBoards**: pings the server and refreshes the list of assigned board data objects held in the Bendit.availableBoards array.
- **getConnectedUsers**: pings the server and refreshes the list of socket IDs belonging to users/performers/audience member’s connected to the server through web performance interfaces.
- **addDevice**: creates a new instance of the BenditDevice class. This returns an object that serves as a virtual representation of a connected Bendit board/circuit-bent device pair in the physical world. Newly created BenditDevices are automatically added to the global Bendit.devices array when this method is called.

Once an instance of the Bendit class is created, performers must declare variable names representing the circuit-bent devices they wish to communicate with. Each variable must then be defined as a new BenditDevice instance, which serves as a virtual representation of a Bendit_I/O-enabled circuit-bent device. This is done by calling the addDevice method on the global Bendit instance and passing in arguments that stipulate the required control parameters (i.e. how many switch, potentiometers (pots), and motor channels will be needed to control elements of the circuit-bent device) and the assigned board number of the Bendit board they want to address. These arguments can be passed into the function as either...
individual numbers or as a JavaScript object (see Listing 4.2). When called, the `addDevice` method builds three arrays, each containing local instances of the library’s Switch, Pot, and Motor classes. The length of each array is defined by the arguments passed in when the `BenditDevice` object constructor is run. Finally, the newly-created `BenditDevice` object is added to the global `Bendit.devices` array.

Listing 4.2 Defining a new `BenditDevice` Object with the `benditBrowser.js` Library

```javascript
let Bendit = require('bendit'); // load benditBrowser.js library as modules
let bendit = new Bendit.Browser(); // define global instance of Bendit class

/*
Create a `BenditDevice` instance to control a DVD player attached to Bendit board 1
*/
let dvd = bendit.addDevice(6,3,0,1); // 6 switches, 3 pots, 0 motors, board 1

/*
Create a `BenditDevice` instance to control a toy attached to Bendit board 2
*/
let toy = bendit.addDevice({"switches": 3, "pots": 8, "motors": 2, "boardNumber": 5});

// log all Devices
console.log(bendit.devices); // -> [BenditDevice{}, BenditDevice{}]
```

The Switch, Pot, and Motor classes created when a `BenditDevice` is initialized all contain methods unique to how these elements are used to perform circuit-bent devices. To engage an i/o channel on a Bendit board, performers execute lines of code formatted in a Dot-Notation syntax. Each line first addresses the device variable name, then the specific Switch, Pot, or Motor object from their individual arrays, followed by a method of that object (a complete list of available methods for each object and summary of their actions can be found in Table
Users can also define variable names for each channel to better match the specific action that results from engaging a particular bend attached to this channel, further tying the syntax of their code to the intended hardware action (see Listing 4.3).

---

**Listing 4.3 Addressing Bendit Board i/o Channels with benditBrowser.js**

```javascript
1 //Set switch 4 on DVD player to "close"
2 dvd.switches[3].setSwitch("close");

4 //Flip state of switch
5 dvd.switches[3].flipSwitch(); //--> switch state: "open"

7 //Set value of pot 5 to circa 50k ohms
8 dvd.pots[4].setPot(127);

10 //Define purpose of specific I/O channels
11 let playPause = dvd.switches[3];
12 let horizontalScrollGlitch = dvd.pots[4];

14 //Engage specific actions/bend by descriptive name
15 playPause.flipSwitch();

17 horizontalScrollGlitch.rampPot(0, 127, 1000);
```

---

The syntax used in benditBrowser.js is modeled after common JavaScript and Object-Oriented Programming practices, and specific design criteria were implemented to ease first-time coders and artists more familiar with electronics and hardware hacking into the process of building web page performance interfaces. The methods for all classes in the benditBrowser.js library use terminology familiar to circuit-bending practitioners, helping to directly associate these new digital performance actions with traditional physical ones. These methods simplify the nomenclature of Socket.io method calls by removing confusion around when to use the library’s "emit", "broadcast", and "rooms" features when passing information through the network. The benditHub.js server library handles the process of creating individual rooms for each new Bendit board that connects, and creating a new Device with
Table 4.1. Available methods for the Switch, Pot, and Motor classes in the benditBrowser.js library.

<table>
<thead>
<tr>
<th>Class</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switch</td>
<td><strong>setSwitch</strong>: set state of switch (&quot;open&quot; or &quot;closed&quot;)</td>
</tr>
<tr>
<td></td>
<td><strong>flipSwitch</strong>: reverse current state of switch</td>
</tr>
<tr>
<td></td>
<td><strong>toggleSwitch</strong>: combine two flips (reverse state, wait, return to previous state)</td>
</tr>
<tr>
<td>Pot</td>
<td><strong>setPot</strong>: set value of pot immediately (0-255)</td>
</tr>
<tr>
<td></td>
<td><strong>rampPot</strong>: ramp between two values over a set amount of time</td>
</tr>
<tr>
<td>Motor</td>
<td><strong>run</strong>: start motor, specify speed and direction</td>
</tr>
<tr>
<td></td>
<td><strong>stop</strong>: stop motor</td>
</tr>
<tr>
<td></td>
<td><strong>flipDirection</strong>: reverse direction, keep current speed</td>
</tr>
</tbody>
</table>

benditBrowser.js sets that Device to only communicate to that room. Additionally, benditBrowser.js creates a single Socket.io socket when a new instance of the Bendit class is initialized and passes that socket into each new member of the BenditDevice, Pot, Switch, and Motor classes as a protected, read-only parameter upon their creation. This prevents the user to from accidentally re-writing or forgetting to include a reference to that socket when receiving data from or sending data to a Device/Bendit board pair.

### 4.4. System Performance

As of this publication, the complete Bendit_I/O framework has been used to create three compositions (all of which will be discussed in the next chapter) that merge networked interaction, technologically-mediated systems, and circuit-bent devices. While both the software and hardware components perform as expected, it is important to note that there is room for improvement in some areas and more rigorous testing in others in order to continue providing artists with a flexible and robust framework for making networked art with circuit-bent devices.

The current Bendit_I/O server is designed to limit connections to a group of 100 Bendit boards/client performance interfaces. This has purposely been done to limit the amount of traffic the server would need to handle and protect against increased latency between the triggering of an event from a participant’s performance interface and the actuation of that
event on a Bendit_I/O-enabled device. Socket.io and Node.js servers have been shown to be capable of handling significant amounts of network traffic, however, the ESP32 microcontrollers at the heart of each Bendit board are limited to communicating over a 2.4GHz wireless frequency, which makes them prone to potential issues with transmission speed and latency. Early tests with a single Bendit board and single web page client communicating through the server on a closed Wi-Fi network showed an average latency time of 25 milliseconds between the tap of an HTML button and the toggling of a board’s switch channel relay. While this latency has not resulted in any significant performance hindrance, a more rigorous stress test on a Bendit_I/O server needs to be performed in order to understand its true limits and get a better sense of the amount of latency that may present itself when dozens of attached clients and Bendit_I/O-enabled devices are connected simultaneously.

Regarding the hardware components of the framework, the use of digital potentiometer ICs on both the v1.0 and v2.0 revisions of the Bendit board present some limitations and potential performance issues. The Bendit board potentiometer channels produce a linear output taper, with no software option currently available to toggle between that and a logarithmic taper. This is due to the internal limitations of the selected IC, the Analog Devices AD5206 (for v1.0 boards) and the AD5204 (for the v.2.0 boards). While these ICs will hopefully prove useful for the needs of most circuit benders in their artwork, the downside is that they do not provide easy control of a hacked device’s volume output level due to their non-logarithmic taper. Additionally, digital potentiometers are prone to producing zipper noise—a form of distortion caused by rapid changes in volume level across a large range—when used to attenuate audio signals. This noise is often considered a detriment to using digitally-controlled volume attenuation in high-quality audio systems, and may create


an issue for those who wish to mediate the control of a hacked device’s volume level in
performance.\textsuperscript{16}

Thankfully, there are a number of relatively easy modifications that can be made in order
to turn a Bendit board’s potentiometer output into an approximated logarithmic taper and
to reduce the intensity of unwanted zipper noise. The addition of additional “pad” resistors
between a channel’s A, B, and center wiper pins, can produce a quasi-logarithmic taper.\textsuperscript{17}
Furthermore, dedicated volume attenuation IC’s exist, a number of which can be controlled
through the same Serial Peripheral Interface protocol used to control a Bendit board’s digital
potentiometer channels, making them an easy add-on option for those who wish to extend
this functionality to their hacked device.\textsuperscript{18} While the production of zipper-noise from a digital
potentiometer is unavoidable, the intensity of the effect can be reduced in a number of ways.
Adding in-line capacitors would serve and a quick and easy approach to smoothing the signal
of a digital potentiometer output channel, although larger values of capacitance would result
delay along the line. A more robust—but more complex—option is the inclusion of circuitry
that stops the IC from updating its internal registers until the audio signal has reached a
zero-crossing window. This noise-reducing approach would require the addition of multiple
operational amplifier, logic gate, and passive filtering components to the output channels of
a Bendit board, compromising its compact form factor and increasing the cost of production
for each unit.\textsuperscript{19}

To provide solutions for controlling volume levels on hacked devices without modifying
the cost and design of the current v2.0 Bendit boards, future research and development will
be directed towards the creation of special expander boards that provide users with zipper-


\textsuperscript{18}PGA2310 /-15V Stereo Audio Volume Control with Low Gain Error (0.05) |, https://www.ti.com/product/PGA2310.

\textsuperscript{19}Li, “Digital Volume Control Eliminates Zipper Noise.”
noise-reducing and dedicated volume-adjusting control options for their projects. These expander boards will be able to connect to a Bendit board’s Pot output channel headers and to the Pot Expansion output respectively. This will allow users to add these features to their projects if and when the need arises without increasing the price and design complexity of a standard Bendit board.
CHAPTER 5. COMPOSING WITH BENDIT_I/O

To demonstrate the artistic capabilities of networking circuit-bent devices, three compositions were created with the use of the Bendit_I/O framework. Varying in form and structure, these works showcase the audiovisual potential of hacked consumer devices and the ways in which mediated and networked performance techniques can be used to interact with them in performance. All three works highlight various interaction topologies and include different configurations of performers, audience participants, and circuit-bent devices. Details pertaining to the design of performance interfaces and control data/audio routing network schema will be discussed for each composition, and accompanying ancillary documents can be found in the appendices of this document where noted and online at http://atmarasco.com/ensemble.

5.1. The Spinning Earth Shall Spread Before You

*The Spinning Earth Shall Spread Before You* (2018, second revision 2020) is a composition for one or two performers and two networked circuit-bent CD players. In performance, each CD player reads from a disc containing the same musical material pre-written by the composer, and the structure of the piece is dictated by the actions taken by the performer(s) to synchronize, phase, and overlap the material on both CD players in an improvisatory manner. The performer(s) move through the material on each CD by pressing virtual buttons and toggle switches on custom HTML performance interfaces that engage the Play, Pause, Forward Seek, and Electronic Shock Protection functions of each player. The web-enabled performance interfaces also allow performers to engage bend points that bridge the data-sampling pins on the CD players’ Digital Read-Access Memory IC and the disconnection of the internal ADAC’s mute function. These modifications allow for unconventional exploration of a CD player’s audio generating capabilities. By disengaging the device’s standard
process of blocking audio output when it is paused, percussive and looping patterns are generated as the laser reads and rereads the same short audio sample at a single point on the disc, corrupting the incoming audio samples that fill the device’s internal audio buffers when Electronic Shock Protection mode is engaged. This results in chaotic, data-mangled sonic outbursts that override the material currently being read from the disc.

5.1.1. Concept

The initial inspiration for The Spinning Earth Shall Spread Before You came from Nicolas Collins’ Broken Light as well as a desire to experiment with turning portable CD players—a consumer music device that played a prominent role in my life throughout adolescence—into hardware granular synthesizers. Hacks were made to both CD players that replicated the unmuted DAC modifications made by Collins, along with the addition of bend points on their internal DRAM ICs, a technique inspired by circuit bender r20029’s 2012 creation The Discbitch. After experimenting with the musical results of these hacks, original audio material was composed and burned onto a CD, serving as the source of melodic and harmonic material to be manipulated throughout the piece.

As is the nature of working with circuit-bent instruments, composing for precise, repeatable musical events can be difficult. With this in mind, the decision was made for the piece to be improvisatory, yet still be built around a through-composed formal structure. This balance between the two compositional approaches was accomplished by choosing the order in which audio tracks would be burnt to the CD and restricting the performer to a forward-only progression through the tracks. With two CD players capable of performing the same material, the initial incarnation of the piece was on a solo performance focused on building gradually evolving musical patterns and counterpoint through the synchronization and phasing of simultaneous audio tracks. By staggering their start times, each CD

\footnote{CD Player Circuit Bending – Hi., December 2012, \url{https://r20029.wordpress.com/2012/10/09/cd-player-circuit-bending-intro/}}
player would begin playing their first track in a round, separated by about 8 bars. When paused by the performer, this pattern of temporal separation is broken; one CD continues to progress forward through its material and the other generates repeated, glitching pulses as it reads and re-reads the same microsample continuously until it is unpaused, creating a new temporally-distanced layering between the two audio sources that results in an entirely new and unpredictable melodic and rhythmic counterpoint.

In its original incarnation, the interaction topology in *The Spinning Earth Shall Spread Before You* followed a centrifugal star graph, where, from the perspective of the performer,
interactions resulted in the playback and circuit bending of both Bendit_I/O-enabled CD players. After multiple performances in this configuration, a new approach was devised which would allow for more chance-based and complicated interaction to occur between the performer and both circuit-bent devices. The 2020 version of the piece allows for the addition of a second performer, and the interaction topology was modified to allow for bidirectional communication between the performers and the hacked devices and from hacked device to hacked device. Now capable of exhibiting agency, the CD players periodically send an event message to a performer, temporarily disabling a randomly-chosen UI element on their interface. Both CD players can pass event messages between one another as well, allowing them to engage and disengage each other’s playback controls and bend-point patches.

Conceptually, these alterations re-imagine the purpose of the piece, now no longer a work only focused on exploring the myriad possibilities of sonic chaos through digital sample manipulation and melodic/rhythmic complexity possible through phasing. In it’s reworked configuration, the piece follows a combination of forest and bidirectional graphs, where small groups are formed (e.g. group with first performer and second CD player, group with both CD players, etc.) and interactions can be initiated by either member of the group. While control signals are not passed between both performers, interaction between them occurs through performative reactions in response to the other’s musical choices. This new hybrid topology allows for a more ensemble-like approach to improvisation and interaction between humans and machines, drawing connections to the approach taken by Collins when scoring his work for string quartet and performer-controlled circuit-bent CD player. Since every human and machine member of the ensemble is capable of making decisions and generating control signals along the network, they can all undertake actions that influence the sonic output and playback actions of another. CD players can block the anticipated-performance actions of the human performers before they can be triggered. Human performers, meanwhile, are forced to adjust their plans for the musical ideas they were hoping to generate, and are asked to embrace not only the sonic chaos that comes with patching bend points but the structural
chaos that comes with performing as one member of a freely-improvising ensemble, creating music out of trust, anticipation, action, and reaction.

5.1.2. Hardware

Two Sony Discman D-E301 CD players were chosen for this piece due to their inexpensive price point on the second-hand market, relative ease of disassembly and hacking, and the existence of previous research proving the ability to manipulate the DRAM data on this model. Wiring was added to the data input pins of the DRAM IC, and early experiments connecting these new bend points found multiple combinations that created intriguing effects through data corruption. Wiring was also soldered to the pads of the Play/Pause, Forward Seek, Electronic Shock Protection On/Off, and Stop buttons allowing for these playback controls to be engaged through the network. Finally, a study of the player’s service manual and schematic showed the existence of multiple audio muting circuits, but only a disconnecting of the DAC’s latching pin allowed for audio data read from the CD to still be sent to the headphone and line out amplifiers while the player was paused. A trace running from the main microprocessor unit to the DAC latch pin was cut, and wiring was added in-between these newly-disconnected points, allowing for the "rhythmic-skipping" unmuting function to be toggled on and off easily in between performances. After all bend points were discovered and internal modification was completed, the newly-added wiring was soldered to a flat ribbon cable to allow for easy connection to a Bendit board (see Figure 5.2). 3D-printed stands were created hold the players upright during performance and avoid unintentional bumping and skipping of the CDs in performance. A CNC-milled wooden enclosure was created to house the attached Bendit board and hide as much of the wiring as possible (see Figure 5.3), allowing the circuit-bent CD players to exude the look of stand-alone electronic instrument.
Figure 5.2. Modifications made to a Sony Discman D-E301 CD player. On the left, wiring is soldered to the playback controls. On the right, traces are cut to stop the DAC latching process, allowing the feature to be toggled on and off by a Bendit board.

Figure 5.3. Circuit-bent Bendit _I/O-enabled CD player placed in a custom enclosure.
5.1.3. Software

Software components for *The Spinning Earth Shall Spread Before You* consist of a web page performance interface and a Bendit_I/O server. The performance interface contains button and toggle user-interface elements from the NexusUI library. Two different web page interfaces exist; the solo-performer web page contains two sets of user interface controls (one set for each CD player) while the dual-performer web pages each contain a single set of controls that each control a single CD player. These interfaces are designed to be run on a touch-screen device such as a smartphone or tablet (see Figure 5.4). Using the benditBrowser.js library, control data is sent to a CD player when a NexusUI interface element is tapped or held. The Bendit_I/O server is run locally on a computer in the performance space and a closed Wi-Fi network is used to connect the smart devices and Bendit_I/O-enabled CD players. The server code is written using the benditHub.js library.

Additional event listener channels are added to the standard server code template for this piece in order to facilitate CD player-to-CD player communication, expanding the server’s core functionality. One new event listener keeps a count of the number of control messages being sent through the server from any performance interface to any CD player. When the server is launched, a randomly-generated Trigger Threshold value (an integer between 3 and 7) is generated within this event listener; if the total number of sent messages is congruent mod the trigger threshold value, the server instructs a random CD player to generate a control message and send it to either a human performer or the other CD player. The resulting control message sent from the CD player is also randomly chosen from an array of messages, some of which are CD player-specific performance actions while others are meant to alter the performance interface actions by temporarily disabling a UI button or toggle switch. If a CD player or a performance interface receives a message that can only be enacted by the other, that message is ignored.

---

5.2. gravity\textit{|}density

\textit{gravity/density} (2019) is a piece for two performers, three circuit-bent devices (a guitar distortion pedal and two CD players), mobile devices, and audience participants. Written as a collaboration with composer Jesse Allison, the piece merges performance of networked, circuit-bent devices with distributed, localized-audio generation on mobile phones through the use of the Web Audio API. One performer controls the actions of the circuit-bent devices while the other distributes audio files and control signals for manipulating HTML performance interfaces to the mobile devices of audience members. The audience participates in the piece by logging onto a web page and controlling the playback of distributed audio files through various tactile interaction methods. Audience members take turns playing both passive and active contribution roles throughout the performance. In certain sections, audio is sent to an audience member’s device and remotely triggered for playback by one of the on-stage performers, while other sections provide more interactive opportunities for audience members to choose when to sample audio being generated from the circuit-bent devices, which portion of an assigned audio file they would like to playback, and when that
Figure 5.5. A diagram of the control data and audio-routing network for *gravity/density*. Dashed lines represent audio routing channels/distributed audio to user devices. Solid lines represent control data sent to three Bendit I/O-enabled devices.
playback should occur. Figure 5.5 shows the control data and audio signal routing used in the networking schema for gravity/density.

5.2.1. Concept

The concept of gravity/density grew out of collaborative research with Jesse Allison on expanding the possible device set used for networked and distributed performances with Web Audio performances to include repurposed and circuit-bent hardware\(^4\) Complimenting recent advancements made in the IoMusT field\(^5\) gravity/density showcases the sonic potentials of bringing discarded musical hardware into the fold as cyber-hacked smart musical instruments, enabling them to interact with users and other custom-made hardware through a network.

From its earliest conception, gravity/density was designed to alleviate potential problems that could stem from providing a large crowd of audience participants with control over a small set of circuit-bent devices.

The decision was made to not allow the audience to have direct control of the circuit-bent devices through their networked interactions and to instead carry over the same unidirectional interaction topology from the original incarnation of The Spinning Earth Shall Spread Before You. Performance of the circuit-bent devices would be driven by one of the two on-stage performers, with audio from the CD players being fed into the second performer’s computer for real-time sampling. The second performer manages a software system (consisting of a Max patch and a web page performance interface built with NexusHub\(^6\) and the Web Audio API) that controls when and how groups of audience members can participate in the performance to add to the overall sonic texture of the piece.


\(^5\) Turchet et al., “Internet of Musical Things.”

\(^6\) NexusHub: Nexus Distributed Performance Framework.”
The piece begins with the two on-stage performers manipulating fixed-audio sources through the performance of two circuit-bent CD players. The audio on the CDs is comprised of original music, mixing synthesized melodic and chordal patterns with sampled archival text and field recordings from the NASA Apollo Mission archives. As the piece progresses, the mangled audio from the circuit-bent CD players are sampled by the second performer’s computer at the request of audience participants who choose when they would like sampling to begin by tapping a UI button on their interfaces. These recorded samples are then distributed to the audience’s mobile devices in both passive and interactive manners. Passive distributions allow the performers to use participant mobile devices as a massive speaker array, creating intricately-spatialized rhythmic interplay between the glitching CD players and the blanket of overlapping samples dispersed throughout the networked audience. Active distributions allow the audience to join in and contribute to the performance by a) tapping their screen to playback their assigned audio file locally and b) choosing small portions of the audio sent to them and sending these selected samples back to the performers. These chosen samples can then be collected, stitched together, and fed into a Bendit_I/O-enabled distortion pedal, allowing this stream of collaboratively-generated audio to be processed by networked hardware in real-time before being sent back to the audience for more manipulation.

There are multiple interaction topologies at play in gravity/density, resulting in overlapping cycles of control and audio generation between performer, audience, and machine. Active distribution sections can best be represented by a hybrid centrifugal/centripetal star and disconnected topology graph, which is applicable when viewed from the perspectives of both the audience and the second performer. In passive distribution sections, the topology changes to a centripetal star and disconnected graph with the flow of interaction stemming from the second performer out to the audience members.

---

7 Matuszewski, Schnell, and Bevilacqua, “Interaction Topologies in Mobile-Based Situated Networked Music Systems.”
The routing of control signals and audio diffusion, along with the multiple forms of audience participation used in gravity/density showcase the ways in which Bendit_I/O can be used to create unique musical experiences between networked circuit-bent hardware and many human participants when coupled with preexisting systems such as the Web Audio API. While audience members do not have direct control over the networked hardware in this piece, the audio that they manipulate with their mobile devices comes from—and is further processed by—the hardware featured on stage. The audience's choice of samples and decision on when to play them back can influence the performance of both on-stage performers, creating a communal channel of improvisatory sonic interaction between both parties. Furthermore, the performative options provided by their web page interfaces allows each audience participant to emulate the same audio-generation techniques as the circuit-bent CD players. Through manipulation of a graphical index window, audience members can select a short selection of their assigned audio file and loop through it, creating glitching
pitch bursts and rhythmic patterns through granular synthesis, a computer music process that can simulate the malfunction of hardware digital media players. Through these actions, the audience members perform a disembodied virtual avatar of the physical hacked hardware featured on stage.

5.2.2. Hardware

Two circuit-bent Sony Discman D-E301 CD players (internally modified in the same manner and capable of creating sonic similar as they are for The Spinning Earth Shall Spread before You) and a DiMarzio Very Metal distortion pedal are used as the networked hardware devices in gravity/density. Each device is connected to a separate Bendit board, enabling them for control along the wireless network. The two CD players retain the same performance controls as they do from The Spinning Earth Shall Spread before You, while the distortion amount is the sole controllable parameter of the pedal. This was achieved by severing two terminals of the pedal’s on-board potentiometer and replacing them with wire connected to the Wiper and B output terminals of a Bendit board’s digital potentiometer channel. To avoid possible control issues caused by mismatched impedance level between the two pieces of hardware, the grounds of both the Bendit board and the distortion pedal are also joined. Figure 5.7 shows all three Bendit I/O-enabled devices ready for performance.

5.2.3. Software

The software components used in gravity/density consist of three web performance interfaces, a Bendit I/O server, and a Max patch used to adjust the levels of all incoming and outgoing streams in the performance and create a 7-channel mix sent to the house sound system. Two web page interfaces are used for controlling audio: one interface is a client-side page used for audio sampling, manipulation, and playback of distributed audio files by audience members.


participants. Another is used by the on-stage performers to record audio coming from the CD players when instructed by the audience participants, upload the resulting audio samples to a server for distribution to client mobile devices, and process audio files remotely through the use Web-Audio-based digital signal processing on each participant’s local device. The interface used by the second performer includes a 2D-map of the concert hall displaying the location of all connected client mobile devices and additional UI elements used for distributing, processing, and triggering audio on them remotely (see Figure 5.8). A third web page interface is used to perform the networked circuit-bent devices and is built using NexusUI button, toggle, and dial elements and the benditBrowser.js library.

The Bendit_I/O server for gravity/density is comprised of two separate servers run simultaneously in performance: one built with the benditHub.js library and used to manage control data between an on-stage performer and the Bendit_I/O-enabled devices, and a second server (built with NexusHub) dedicated to managing control data and audio-file distri-

\footnote{Allison, Oh, and Taylor, “NEXUS: Collaborative Performance for the Masses, Handling Instrument Interface Distribution through the Web.”}
Figure 5.8. Two web performance interfaces used in *gravity/density*. On the left, the client-side interface used by audience participants to select and playback portions of an assigned audio file; on the right, a second interface (displaying the location of all connected client interfaces in the concert hall) used by an on-stage performer to record audio from the CD players, distribute it to audience mobile devices.

bution between the audience participants and the second on-stage performer. The decision to spread the workload across two servers was made to avoid overtaxing and potentially crashing the server mid-performance and to keep the latency between a performer’s control messages and the Bendit devices to a minimum.

5.3. Camera Studies, Vol. 1

*Camera Studies, Vol. 1* (2020) is the first in a planned series of pieces for networked circuit-bent visual equipment and live electronics. The first volume is scored for a single performer, circuit-bent digital handicam, CRT television, synthesizer, and a web-enabled four-track looper and effects patch built in Max. The piece begins with repeating loops of static noise recorded live from the speakers of the television repeating in asynchronous patterns. A digital handicam is positioned so that it is aimed at both the blank television screen and the live performer, displaying a live video feed of its view onto its built-in LCD screen. During the
performance, improvised synthesizer melodies are performed over four looped tracks, as their start, end, and current playhead position points altered as the piece progresses. Real-time digital signal processing in the form of granular pitch shifting and pitched delay also occurs on all four audio tracks, and the effect parameters are modified directly by the performer through gestural motions captured on the synthesizer’s MIDI keyboard interface. A second, unmodified camera displays a close-up view of the handicam’s LCD monitor to the audience.

Throughout the piece, the performer has no direct interaction or control over the circuit-bent handicam. Instead, the web-enabled looping and effects software includes a subsection that communicates with the networked circuit-bent handicam, triggering bends inside the modified camera in reaction to musical events. For each loop track, a random trigger point number—representing a position in time within that track’s length—is generated and stored once audio is first recorded into its corresponding buffer. When a loop’s playhead reaches the position in the buffer that matches its assigned trigger point, an event message is sent to the Bendit_I/O-enabled handicam, resulting in the patching of up to four bend points. These bends result in a live processing of the handicam’s video feed, modifying the image of the performer. In the final section of the piece, the handicam’s video feed is switched out to be projected through the TV, creating an ouroboros of visual feedback. At this point, new bends within the handicam are engaged by the loops as their playback positions become randomly set, turning the once-live video feed into a frozen image that periodically glitches and degrades. The intensity of this visual processing is modified though resistance changes across the handicam’s bend points, generated from the same gestural MIDI performance data used to modulate audio effect parameters earlier in the piece. As the camera’s feed continues to glitch, the performer fades out all audio tracks and leaves the stage.

5.3.1. Concept

The central concept of Camera Studies, Vol. 1 stems from my interest in exploring the visual possibilities available through exploratory hacking of consumer-grade video and television
Figure 5.9. Images from a performance of *Camera Studies, Vol. 1.*
equipment. Conversion of this equipment into readymades serves as means of subverting the important role these devices play in capturing memories and providing entertainment, information, and other forms of visual media to the masses. In addition to the work of Phillip Stearns discussed earlier, work probing these topics has been done by artists such as Nam June Paik (creator of the Wobbulator, a visual synthesizer built from a modified CRT television[^11], and teams such as Eric Souther, Laura McGough, and Jason Bernagozzi, who produced avant-garde visuals through modification of digital-to-analog television converter boxes[^12]. Using these works as a starting point, I contemplated the idea of creating circuit-bent live streams as the world moved into lock-down status at the height of the global COVID-19 pandemic in March 2020 and social gatherings ranging from work meetings to choir concerts moved into the virtual world of internet meeting spaces. Imagining how the visual elements of a typical telematic concert could be purposely distorted in an effort to embrace and celebrate glitches commonly caused by disrupted internet connections and coupling this with the desire to explore a remediation of personal home video equipment, *Camera Studies, Vol. 1* aims to bring unpredictable video distortion into the realm of a controlled, performative environment.

The interaction topology designed for *Camera Studies, Vol. 1* is significantly simpler that those at play in *gravity/density* and *The Spinning Earth Shall Spread Before You*, resulting in a different relationship between performer and hacked hardware. The network consists of only two client devices hosted on a server—the Bendit_I/O-enabled handicam and the performer’s web-enabled stochastic looper patch—and only a unidirectional flow of control.


data is established. The handicam reacts to messages sent from the stochastic looper, but
does not generate any control messages of its own, nor does it initiate any communication
with the patch (see Figure 5.10). Control messages to the digital handicam’s are generated
by the patch’s audio-reactive stochastic triggering system as opposed to being generated by
direct interaction with performance interface elements such as HTML buttons, toggles, and
dials or by another networked circuit-bent device. Since the performer is not made aware of
the trigger point numbers assigned by the stochastic looper patch during the performance,
their primary focus during performance is on generating audio and not on manipulating
the bend points on the circuit-bent handicam. The stochastic looper patch mediates the
performer’s interactions with a hacked device resulting in a ambiguous balance between
intention, actions, and results. While the performer is aware that their actions are being
used to in generate visual material from the handicam, they are unable to determine which
action resulted in which visual texture and at which point in time. This shifting of agency
and control away from the performer and onto a technological mediation system serves
to highlight the ways in which Bendit_I/O can lead to more complicated, unpredictable
interactions with circuit-bent devices.

5.3.2. Hardware

The sole circuit-bent device in Camera Studies, Vol. 1 is an Aiptek 8 megapixel digital hand-
icam. Unlike the aforementioned Sony D-E301 Discman CD players, the Aiptek handicam
condenses a number of crucial operations into a single microprocessor unit, most notably
the system control functions (such as enabling recording and playback of footage) and the
video codec. Simply put, extended circuit-bending on this particular IC could lead to costly

---

13 Using the graphs established by Matuszewski et al., the interaction topology for this piece is a bit odd
due to the limited paths for actions and reactions to travel. It could either be considered a unidirectional
circle (action moves from the stochastic looper node to the handicam node) or a very scaled-down centrifugal
star, since the the interaction events from looper to camera are not strictly "sequenced". With only two
nodes on the network and one path for interaction to travel, a more accurate graph might simply be a
straight line connecting one node to the next, illustrated in Figure 5.10.
Figure 5.10. The control data network for Camera Studies, Vol. 1.
mistakes; damaging portions of this IC would render the camera unusable by shutting down its video processing functionality. The decision was made to divide the bend points across two different ICs with four located on the data output pins of the LCD screen driver IC, and four located on the central microprocessor unit IC. This resulted in the rainbow saturation and horizontal white scroll effects only being visible through the LCD screen, while the doubled-image, digital artifact effect (made possible by connecting multiple pins on the microprocessor unit) can be displayed on an external monitor through the handicam’s audiovisual output ports (see Figure 5.10). The LCD screen bends can be engaged without freezing the video processing unit, allowing them to be featured in the first half of the piece, while the remaining bends (which result in a halt in the video codec’s frame refresh process) require a complete restart of the device to return to normal operation and are only enacted at the piece’s end.

A Bendit board was connected to the chosen bend points via two ribbon cables. Four switch output channels are used to patch the aforementioned bend-point pairs (see Figure 5.11). Additionally, a single pot output channel is connected in series between the bend points on the microprocessor unit. Subtle changes in the level of resistance between these connected points result in varying modifications to the projected image, including different horizontal offset amounts between the frozen image captured in the codec’s buffer and its "ghosted" duplicate image and a change in the amount green tint added into the final image.

5.3.3. Software

The software components of Camera Studies, Vol. 1 consist of a Bendit_I/O server and a performance patch created in Max. The performance patch’s core functionality is split amongst three sub-patches: a four-track audio looper, a multi-channel digital signal processor utilizing a pitch-shifting delay line and granular synthesis engine, and a web-enabled stochastic trigger system used to engage and disengage the Bendit_I/O-enabled digital hand-
icam. Connection between the patch and server is handled by an instance of the node.script object, which runs a local Node instance, connects to the server as a client web interface, and executes event commands to the handicam-connected Bendit board upon receiving messages from other sub-patches. The performer interacts with the patch through the computer’s keyboard and trackpad, as well as through a MIDI keyboard.

Serving as a mediation system between musician and hacked handicam, the patch negotiates between the performer’s actions and the resulting bends. Before the start of the piece, four audio samples of white noise emanating from the CRT television’s speakers are recorded into the patch’s buffers. Once the recording process has ended, the stochastic triggering portion of the patch assess the total length in seconds of each sample, randomly chooses a number between zero and the total sample duration, and stores that number as the corresponding loop’s trigger point. As the performer seeks through each loop or modifies their end-of-cycle/beginning-of-cycle points, a message is sent to the handicam’s Bendit board instructing it to engage or disengage a bend point. Changing values generated from pressure data captured from the MIDI controller’s keys (sent to both the hardware synthesizer to modulate pitch and to the software performance patch to change effects parameters)
is written to a dictionary object in the patch during the first half of the piece. Once the video output from the handicam changes from the LCD screen to the CRT television in the final section, values from that dictionary are read back randomly and used to adjust the resistance level across the handicam’s microprocessor bend points, tying the current modulation of the frozen image to the performer’s previous musical actions.
CONCLUSION

Before looking forward to the future, let us return once more to the beginning. In his 2004 retrospective article covering the origins of his circuit-bending practice, Qubais Reed Ghazala writes:

I suppose the greatest value I see in circuit-bending, beyond the new palette, is how the art encourages fresh musical thought. It is these two aspects, the art’s sound and its ideas, that have kept me at the bench and in the studio for decades.\footnote{Ghazala, “The Folk Music of Chance Electronics: Circuit-bending the Modern Coconut.”}

Ghazala’s summary of the personal practice he formally named and shared with the world nearly three decades ago boils down the layers of aesthetic theory, philosophical analysis, and anticonsumerist activism surrounding the core conceits of circuit bending as a practice: an exploration in search of new sounds and new perspectives. Those same conceptual seeds lie at the center of networked musical performance, a practice which, through a backward tracing of its evolving software and hardware components, its advancements in data transmission protocols and telematic reach, and its ever-adapting commentary on how ubiquitous computing has reshaped our lives, we can see stemming from goals not dissimilar from Ghazala’s: a desire to search out new perspectives on communication and influence through collective music-making processes.

It is here now, at its own core, where Bendit_I/O stands and looks outward towards the layers of aesthetic theory, philosophical discussion, and technological advancements of its core components that will hopefully grow and envelop with the passage of time. As the framework makes its way into the hands of artists, aspects of its software and hardware components will inevitably be pushed to and beyond their limitations in the same manner as the source devices of circuit-bent instruments themselves. While a complete and working system at its current scale and design, there are will always be new avenues for improvement.
and advancement to be made as the circuit bending and networked musical performance practices it serves to synthesize grow and evolve in their own right. In the immediate future, beta testing endeavors will begin and workshops focuses on leading users through the process of getting started with the framework will be held. Through hands-on experiences observing how users approach Bendit_I/O as a tool for enhancing their current practices, new ideas on features to add into the system as well as unforeseen faults or limitations in its current state will become apparent. These insights will help to form a road map for future development goals and feature sets aimed at enhancing the system’s performance possibilities and easy of use.

As of this publication, the first new updates planned to the software side of Bendit_I/O include the creation of a board registration and configuration front-end application, to be written for use on Mac and Windows operating systems. This application will supplant the need for users to modify their Bendit board firmware code from within the Arduino IDE upon first use, providing them with user-friendly GUI options such as text boxes and drop-down menus instead. The application will also give users the ability to select, launch, and monitor feedback from a Bendit_I/O server file of their choosing, avoiding the need for these steps to be undertaken in a terminal window or from a code editor IDE. Additionally, the creation of a Bendit-Max Node module will soon be undertaken in order to bring the function calls and circuit-bending-focused syntax for the benditBrowser.js library to the Max platform, making use of its Node-for-Max components. This will allow for easier development of audiovisually-focused software patches to be developed for collaboration with and control of Bendit_I/O-enabled circuit bent devices in performance.

Focusing on the hardware element of the framework, rigorous testing of the v2.0 Bendit board design will occur as the beta test phase begins. New hardware add-on components that make use of the v2.0 board’s Switch Extension, Pot Extension, I^2C, DAC, and ADC breakout ports will be developed, giving users the ability to add more Switch and Pot channels to a single board in addition to add electrical signal sampling and generating functionality.
Providing adventurous users with new options for designing bidirectional data channels with a networked circuit-bent device, these new hardware features—existing as plug-and-play add-on circuits—will allow users to add more i/o options to their Bendit boards as the need arises without raising the price of the framework’s standard-issue hardware components.

Finally, new feature sets and changes to the software and hardware elements of the framework will spur on changes in the online documentation and educational materials provided to Bendit_I/O users. After workshops are held and the beta testing period closes, updated versions of the benditBrowser.js, benditHub.js, upcoming Bendit-Max API, and reference documentation will be published online at www.benditio.com. Eventually, all components of the framework will become open source, allowing users to build their own boards and to design new software features for each library, helping to suit their specific artistic needs.

As time marches forward, the consumer electronic objects most commonly chosen as source devices for circuit-bent instruments will eventually be lost to us forever. The last remaining Speak ‘n’ Spell will, one day, break irreparably with no matching replacement ICs available to resurrect its unique voices both hidden and obvious. When the time comes, more advanced devices—once cutting edge, but now relegated to obsolescence—will replace the ones that came before them as the circuit-bending practice shoulders on with a new territory of electronic landscapes now available for exploration. Along with the technical and collaborative elements at the heart of networked musical performance, its my hope that the research undertaken here and the existence of Bendit_I/O helps to reshape and reconfigure the sense of what is possible when it comes to the performance, networking, and exploration of everyday consumer devices and how their life (and after-life) affects our own in perpetuity.
APPENDIX A. PERFORMANCE INTERFACE CODE

Listing A.1 Web Performance Interface for *The Spinning Earth Shall Spread Before You* (2020)

```javascript
/*
The Spinning Earth Shall Spread Before You (2020)
Performance Interface
Anthony T. Marasco - 2020
*/

// Init Bendit and BenditDevices
const Bendit = require('bendit');
const bendit = new Bendit.Browser();
let cd1 = bendit.addDevice(6, 0, 0, 1);
let cd2 = bendit.addDevice({
  "switches": 6,
  "pots": 0,
  "motors": 0,
  "boardNumber": 2
});

// CD Player 1 Controls
let pb1 = new Nexus.TextButton('#toggleButton1', {
  'size': [150, 50],
  'state': false,
  'text': 'Play',
  'alternateText': 'Pause'
});

let sb1 = new Nexus.TextButton('#toggleButton2', {
  'size': [150, 50],
  'state': false,
  'text': 'Stop'
});
```
let ffb1 = new Nexus.TextButton('#toggleButton3', {
' size': [150, 50],
'state': false,
'text': 'Forward'
});

let espb1 = new Nexus.TextButton('#toggleButton4', {
'size': [150, 50],
'state': false,
'text': 'ESP on',
'alternateText': 'ESP off'
});

let trigMute_1 = new Nexus.Toggle('#toggleButton5', {
'size': [150, 50],
'state': false
});

let trigBend_1 = new Nexus.Toggle('#toggleButton6', {
'size': [150, 50],
'state': false
});

//CD Player 2 Controls
Nexus.colors.accent = "#FFA07A";

let pb1_2 = new Nexus.TextButton('#2_toggleButton1', {
'size': [150, 50],
'state': false,
'text': 'Play',
'alternate': 'Pause'
});
let sb1_2 = new Nexus.TextButton('#2_toggleButton2', {
  'size': [150, 50],
  'state': false,
  'text': 'Stop'
});

let ffb1_2 = new Nexus.TextButton('#2_toggleButton3', {
  'size': [150, 50],
  'state': false,
  'text': 'Forward'
});

let espbl_2 = new Nexus.TextButton('#2_toggleButton4', {
  'size': [150, 50],
  'state': false,
  'text': 'ESP off',
  'alternate': 'ESP on'
});

let trigMute_2 = new Nexus.Toggle('#2_toggleButton5', {
  'size': [150, 50],
  'state': false
});

let trigBend_2 = new Nexus.Toggle('#2_toggleButton6', {
  'size': [150, 50],
  'state': false
});

//React to Messages from Bendit Boards

switch (bendit.messageFromBoard.boardNumber) {
  case 1:
    //Disable Performer Controls
    if (bendit.messageFromBoard.content === "performer_2_disable_play") {
      pb1.size = [0, 0];
    } else if (bendit.messageFromBoard.content === "performer_2_disable_stop") {

sbl.size = [0, 0];
}
} else if (bendit.messageFromBoard.content === "performer_2_disable_forward") {
  ffb1.size = [0, 0];
}
} else if (bendit.messageFromBoard.content === "performer_2_disable_esp") {
  espb1.size = [0, 0];
}
} else if (bendit.messageFromBoard.content === "performer_2_disable_DRAMBend") {
  trigBend_1.size = [0, 0];
}

//Enable Performer Controls
if (bendit.messageFromBoard.content === "performer_2_enable_play") {
  pbl1.size = [150, 50];
}
} else if (bendit.messageFromBoard.content === "performer_2_enable_stop") {
  sb1.size = [150, 50];
}
} else if (bendit.messageFromBoard.content === "performer_2_enable_forward") {
  ffb1.size = [150, 50];
}
} else if (bendit.messageFromBoard.content === "performer_2_enable_esp") {
  espb1.size = [150, 50];
}
} else if (bendit.messageFromBoard.content === "performer_2_enable_DRAMBend") {
  trigBend_1.size = [150, 50];
}
}

break;

//Disable Performer Controls
if (bendit.messageFromBoard.content === "performer_1_disable_play") {
  pbl1_2.size = [0, 0];
}
} else if (bendit.messageFromBoard.content === "performer_1_disable_stop") {
  sb1_2.size = [0, 0];
}
} else if (bendit.messageFromBoard.content === "performer_1_disable_forward") {
  ffb1_2.size = [0, 0];
}
} else if (bendit.messageFromBoard.content === "performer_1_disable_esp") {
  espb1_2.size = [0, 0];
}

trigBend_1.size = [150, 50];
} else if (bendit.messageFromBoard.content === "performer_1_disable_esp") {
    espb1_2.size = [0, 0];
} else if (bendit.messageFromBoard.content === "performer_1_disable_DRAMBend") {
    trigBend_2.size = [0, 0];
}

//Enable Performer Controls
if (bendit.messageFromBoard.content === "performer_1_enable_play") {
    pb1_2.size = [150, 50];
} else if (bendit.messageFromBoard.content === "performer_1_enable_stop") {
    sb1_2.size = [150, 50];
} else if (bendit.messageFromBoard.content === "performer_1_enable_forward") {
    ffb1_2.size = [150, 50];
} else if (bendit.messageFromBoard.content === "performer_1_enable_esp") {
    espb1_2.size = [150, 50];
} else if (bendit.messageFromBoard.content === "performer_1_enable_DRAMBend") {
    trigBend_2.size = [150, 50];
}

break;

//CD 1 Controls

//Play Button
pb1.on('click', function () {
    cd1.switches[0].setSwitch("close");
});

pb1.on('release', function () {
    cd1.switches[0].setSwitch("open");
});

//Stop Button
sb1.on('click', function () {
    cd1.switches[1].setSwitch("close");
});
sb1.on('release', function () {
    cd1.switches[1].setSwitch("open");
});

//Forward Seek Button
ffb1.on('click', function () {
    cd1.switches[2].setSwitch("close");
});

ffb1.on('release', function () {
    cd1.switches[2].setSwitch("open");
});

//ESP Toggle - needs .toggleSwitch to avoid holding system pin high for too long
espb1.on('change', function () {
    cd1.switches[3].toggleSwitch(200);
});

//Mute Latch
trigMute_1.on('change', function () {
    cd1.switches[4].flipSwitch();
});

//DRAM Bend
trigBend_1.on('change', function (v) {
    cd1.switches[5].flipSwitch();
});

//CD Player 2 Controls

//Play Button
pb1_2.on('click', function (v) {
    cd1.switches[0].setSwitch("close");
});

pb1_2.on('release', function (v) {
    cd1.switches[0].setSwitch("open");
});
//Stop Button
sb1_2.on('click', function (v) {
    cd1.switches[1].setSwitch("close");
});

sb1_2.on('release', function (v) {
    cd1.switches[1].setSwitch("open");
});

//Forward Seek Button
ffb1_2.on('click', function (v) {
    cd1.switches[2].setSwitch("close");
});
ffb1_2.on('release', function (v) {
    cd1.switches[2].setSwitch("open");
})

//ESP Toggle - needs .toggleSwitch to avoid holding system pin high for too long
espb1_2.on('change', function () {
    cd2.switches[4].toggleSwitch(200);
});

//Mute Latch
trigMute_2.on('change', function (v) {
    cd2.switches[3].flipSwitch();
});

//DRAM Bend
trigBend_2.on('change', function (v) {
    cd2.switches[5].flipSwitch();
});
APPENDIX B. PERMISSIONS

Figure 1.1

Olivia Lloyd <olivia.lloyd@voxmedia.com>
Mon 6/22/2020 12:19 PM
To: Anthony Marasco <amarasco@lsu.edu>

Thank you for your email. We can approve the requested screenshot, please proceed (we don’t need a license for this use).

Olivia Lloyd | Rights Specialist

85 Broad Street, 15th Floor
New York, NY 10004
olivia.lloyd@voxmedia.com

Figure 1.2

Circuitbenders <info@circuitbenders.co.uk>
Tue 6/23/2020 3:15 PM
To: Anthony Marasco <amarasco@lsu.edu>

Hi,
Yes, that would be fine.

Best Regards,
Paul
Figure 1.3

Part A is reprinted here under a Creative Commons Attribution 2.0 Generic license. The license can be viewed at [https://creativecommons.org/licenses/by/2.0/deed.en](https://creativecommons.org/licenses/by/2.0/deed.en). The image has not been altered. Part B is reprinted here by permission of A.J. Gannon:

aj@mygannon.com

Wed 6/24/2020 7:07 PM
To: Anthony Marasco <amarasco@lsu.edu>

Hello there! This is all pretty awesome! You can totally use this image for your dissertation, I’m honored.
Thanks so much! Aj

Figure 1.4

Nicolas Collins <ncollins@saic.edu>

Mon 6/22/2020 9:10 AM
To: Anthony Marasco <amarasco@lsu.edu>

Hello Mr. Marasco:
I’m happy to grant permission for you to reproduce the image.
Good luck.
Nic

Figures 1.5 and 1.6

phillip stearns <phil@phillipstearns.com>

Wed 6/24/2020 3:46 PM
To: Anthony Marasco <amarasco@lsu.edu>

Hi Anthony,

You have my full permission to use the requested images in your dissertation.

Best,

Phillip David Stearns

Figure 2.1

Benjamin Matuszewski <Benjamin.Matuszewski@ircam.fr>
Mon 6/22/2020 2:34 PM
To: Anthony Marasco <amarasco@lsu.edu>

Hi Anthony,

Yes, of course and thanks for asking!

Bests, Benjamin

Figure 3.1

pbussigel@ecuad.ca <pbussigel@ecuad.ca>
Wed 6/24/2020 6:24 PM
To: Anthony Marasco <amarasco@lsu.edu>

Hey Anthony,

Congratulations on finishing your Ph.D. and feel free to use that image.

All the best, Peter

Figure 3.2

Jenny Odell <jodell@stanford.edu>
Tue 6/23/2020 11:46 AM
To: Anthony Marasco <amarasco@lsu.edu>

Hi Anthony,
Thank you for reaching out – yes, you have my permission to use those images. Wishing you the best with your work!

Cheers, Jenny

**Figures 5.1, 5.5, and 5.10**

Diagrams created by the author. All icons (except for the Bendit board icon) used in these figures were made by Freepik from www.flaticon.com and used in accordance to their free-use and attribution policy.

**Figure 5.6**

From: Alessia Milo <a.milo@qmul.ac.uk>
Tue 6/23/2020 1:13 PM
To: Anthony Marasco <amarasco@lsu.edu>

Dear Anthony,

Thank you for contacting me! Please go ahead with using the image, it’s a good one.

Best wishes,

Alessia
BIBLIOGRAPHY


Allison, Jesse, Vincent A Cellucci, and Derick Ostrenko. “Creative Data Mining Diamonds in Dystopia: An Interactive Poetry Web Application.” *Journal of the New Media Caucus/ISSN*, 1942, 017X.


Mayton, Brian D., Gershon Dublon, Nicholas Joliat, and Joseph A. Paradiso. “Patchwork: Multi-User Network Control of a Massive Modular Synthesizer.” In *Proceedings of the


PGA2310 /-15V Stereo Audio Volume Control with Low Gain Error (0.05) /. https://www.ti.com/product/PGA2310.


VITA

Anthony T. Marasco is a composer and sound artist who takes influence from the aesthetics of today’s Digimodernist culture, exploring the relationships between the eccentric and the everyday, the strict and the indeterminate, and the retro and the contemporary. These explorations result in a wide variety of works written for electroacoustic ensembles, interactive computer performance systems, and multimedia installations.

An internationally-recognized artist, his music and installations have been presented across the United States as well as in Norway, Italy, Brazil, Denmark, and Canada. He has received commissions from performers and institutions such as WIRED Magazine, Phyllis Chen, the American Composers Forum Philadelphia, Quince Contemporary Vocal Ensemble, Toy Piano Composers, the Rhymes With Opera New Chamber Music Workshop, Data Garden, Maureen Batt, and the soundSCAPE International Composition Exchange. Marasco was the grand-prize winner of the UnCaged Toy Piano Festival’s Call for Scores, a resident artist at Signal Culture Experimental Media Labs, and a grant winner for the American Composers Forum’s “If You Could Hear These Walls” project. His works and research have been featured at festivals such as New Interfaces for Musical Expression (NIME), the Web Audio Conference, the Toronto International Electroacoustic Symposium, the Society for Electro-Acoustic Music in the U.S. (SEAMUS), Electroacoustic Barn Dance, New York City Electroacoustic Music Festival, the International Computer Music Conference (ICMC), the National Student Electronic Music Event (NSEME), Mise-En Festival, Montreal Contemporary Music Lab, Electric LaTex, and Omaha Under the Radar.

Marasco intends to graduate with a Ph.D. in Experimental Music Digital Media at Louisiana State University. His research centers on the creation of new software and interfaces for digital art performance and installation. While at LSU he served as an Extern Instructor of Digital Design and Emergent Media for LSU, and a Program Coordinator for the Digital
Design and Emergent Media LSU STEM Certification Pathway. He previously served on the faculties of the University of Scranton and the Pennsylvania State University.