

“A Computer for the Rest of You”: Human-Computer Interaction in the Eversion

by

Shaun Gordon Macpherson  
B.A., University of Victoria, 2011

A Thesis Submitted in Partial Fulfillment  
of the Requirements for the Degree of

MASTER OF ARTS

in the Department of English

© Shaun Gordon Macpherson, 2014  
University of Victoria

All rights reserved. This thesis may not be reproduced in whole or in part, by photocopy  
or other means, without the permission of the author

**Supervisory Committee**

“A Computer for the Rest of You”: Human-Computer Interaction in the Eversion

by

Shaun Gordon Macpherson  
(B.A., University of Victoria, 2011)

**Supervisory Committee**

Dr. Jentery Sayers, Department of English (CSPT)  
**Supervisor**

Dr. J. Allan Mitchell, Department of English (CSPT)  
**Departmental Member**

Dr. Arthur Kroker, Department of Political Science (CSPT)  
**Outside Reader**

## **Abstract**

### **Supervisory Committee**

Dr. Jentery Sayers, Department of English (CSPT)  
**Supervisor**

Dr. J. Allan Mitchell, Department of English (CSPT)  
**Departmental Member**

Dr. Arthur Kroker, Department of Political Science (CSPT)  
**Outside Reader**

With the increasing ubiquity of networked “smart” devices that read and gather data on the physical world, the disembodied, cognitive realm of cyberspace has become “everted,” as such technologies migrate the communications networks and data collection of the Internet into the physical world. Popular open-source “maker” practices—most notably the practice of physical computing, which networks objects with digital environments using sensors and microcontrollers—increasingly push human-computer interaction (HCI) into the physical domain. Yet such practices, as political theorists and some philosophers of technology argue, bypass the very question of subjectivity, instead lauding the socioeconomic liberation of the individual afforded by open-source hardware practices. What is missing across these discourses is a technocultural framework for studying the material ways that everted technologies articulate subjects. I argue that examining the various, contradictory forms of interface that emerge from physical computing provides such a framework. To support this claim, I focus on several case studies, drawn from popular physical computing practices and communities, and analyze the particular ways that these devices articulate subjectivity. I conclude by linking my technocultural framework with various feminist theories of boundary transgression and hybridity, and end by suggesting that, in an everted landscape, the subject is politically constituted by a proximity to present time and space.

## Table of Contents

Supervisory Committee .....	ii
Abstract .....	iii
Table of Contents .....	iv
Table of Figures .....	v
Acknowledgments .....	vi
Introduction: The Eversion of Cyberspace .....	1
“A Computer for the Rest of You”: Physical Computing and the Maker Movement .....	13
Three Interfaces .....	27
Chapter 1: Arduino, Manufacturing, and Productive Subjectivity .....	43
The Productive Subject .....	54
Chapter 2: RepRap, Handicraft, and the Expressive Subject .....	62
Chapter 3: DIY Drones and the Fiduciary Subject .....	79
Conclusion: HCI as Breached Boundary and the Presence of the Subject .....	95
The Politics of a Present Subject .....	99
Works Cited .....	104

## Table of Figures

<i>Figure 1: Representation of Project Tango technology</i> .....	1
<i>Figure 2: A bank of latch relays in a “relay room” of an early analog computer</i> .....	6
<i>Figure 3: Botanicalls sensor in a potted plant and a Twitter account for the device.</i> .....	16
<i>Figure 4: A partial list of permissions for the “free” Nike+ Running app</i> .....	18
<i>Figure 5: “The Maker Bill of Rights”</i> .....	20
<i>Figure 6: The Mac OS “desktop” for the Apple Macintosh 128K</i> .....	32
<i>Figure 7: “How the Computer Sees Us”</i> .....	34
<i>Figure 8: The Arduino UNO microcontroller board</i> .....	43
<i>Figure 9: The Busicom 141-PF printing calculator and the Intel MC4-004</i> .....	47
<i>Figure 10: A populated circuit board with an epoxy-covered integrated circuit</i> .....	49
<i>Figure 11: Textspresso, an Arduino-integrated espresso machine</i> .....	50
<i>Figure 12: A side-by-side comparison of the RepRap Mendel and the Darwin</i> .....	66
<i>Figure 13: : Screenshot of FreeCad, a WYSIWYG CAD program for RepRap.</i> .....	68
<i>Figure 14: Customizable infill patterns for fused-filament desktop fabricators</i> .....	72
<i>Figure 15: A misprinted object on a 3D print bed</i> .....	75
<i>Figure 16: The ArduCopter installed on an RC helicopter.</i> .....	83
<i>Figure 17: A Screengrab of Mission Planner flight programmer</i> .....	88
<i>Figure 18: The serial monitor readout of the ArduEye</i> .....	89
<i>Figure 19: An Afghan man has his iris scanned by a US Army officer</i> .....	102

## Acknowledgments

My sincere appreciation goes out to the members of my thesis committee. Dr. Jentery Sayers has provided immeasurable support throughout the course of this project, and his insights into both contemporary media studies discourses and technical practice were vital to the development of this topic. I am enormously grateful to Dr. J. Allan Mitchell for his willingness to participate in this committee, and for offering his attention and a fresh perspective on the topic of everted technologies. My good fortune has also afforded me the privilege of working with Dr. Arthur Kroker, whose enthusiasm and encouragement throughout the past year I am immensely thankful for. My deepest thanks also goes to Dr. Lisa M. Mitchell for agreeing to sit on my committee.

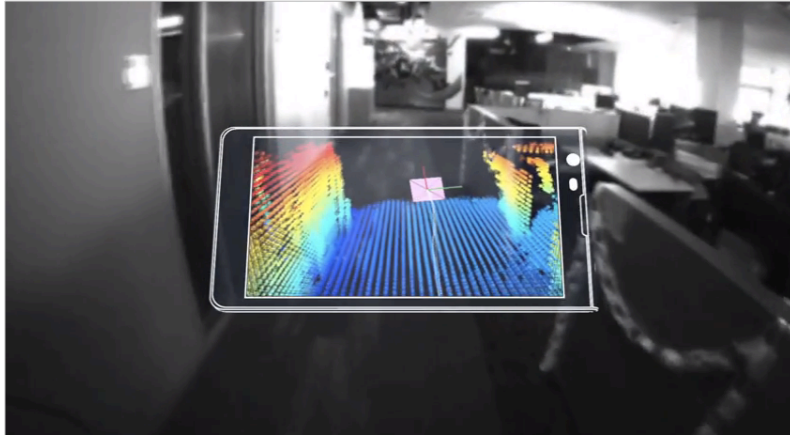
I owe a huge debt of gratitude to Dr. Nicole Shukin, for her inspiration, guidance, wisdom, and investment in my development as a critical thinker as I navigated the graduate program; to Dr. Lisa SurrIDGE, who encouraged me to pursue graduate studies; and to Marilouise Kroker for her kindness, humour, and encouraging words of support throughout the writing of this thesis. I am also indebted to Dr. William J. Turkel at Western University for his interest in my research and indispensable expertise in the history and practices of programmable technologies.

A word of thanks is needed for the vibrant, inclusive, and stimulating community of graduate students in both English and in the Cultural, Social, and Political Thought concentration. This group has made the past three years immensely fun, challenging, and intellectually exhilarating. In particular, I'd like to thank Tim Personn and Mike Smith, whose shared wisdom and input was crucial to my development as reader of theory, and to Jana Millar Usiskin and Mikka Jacobson, who read and commented on earlier drafts of this thesis.

Thank you also to my parents, Bob and Vivian Macpherson, for their unwavering support and enthusiasm for my academic development.

Finally, thank you to Nina Belojevic, whose daily support and encouragement in the face of fatigue, doubt, and my obsession with this topic was the single reason this project was completed.

## Introduction: The Eversion of Cyberspace



*Figure 1: Representation of Project Tango technology mapping indoor terrain in three-dimensional real-time.*

In February 2014, Google’s Advanced Technology and Projects Group (ATAP)<sup>1</sup> announced Project Tango, a smartphone prototype designed to “give mobile devices human-scale understanding of space and motion” (“Project Tango” n. pag.). Project Tango combines computer vision with geolocation sensors to enable a phone to track its motion in three-dimensional space while simultaneously geometrically mapping the space around it (see Figure 1).<sup>2</sup> The motivating principle behind Project Tango is to create devices that can situate themselves within their physical environment akin to the way that humans tacitly perceive and navigate space. In other words, the device combines

---

<sup>1</sup> Formerly the Motorola Advanced Technology and Projects Group.

<sup>2</sup> These sensors include a megapixel camera, two computer vision processors, an integrated depth sensor, and a motion tracking camera, as well as the accelerometer, gyroscope, magnetometer, and GPS sensors that are already ubiquitous elements of smartphones. According to the project’s website, “these sensors make over a quarter million 3D measurements every single second, updating its position and orientation of the phone in real-time, combining this information into a single 3D model of the environment” (“Project Tango”).

various discrete sensors to algorithmically construct something approximating an element of human perception. Such behaviour would open up a whole range of interactive possibilities: from rapid three-dimensional mapping of an indoor environment to new applications for video games or augmented reality apps that integrate the physical environment<sup>3</sup> to assist people with special needs. Of course, since Google is funding the project, it is also reasonable to speculate how the information gathered by these devices will be integrated into the company's grander (read: hegemonic) project of data collection—that Project Tango's slogan is “The future is awesome. We can build it faster together” hints that the company's next era of world-mapping will be increasingly crowd-sourced. In short, the device works to both simulate a mode of perception and process information algorithmically and, presumably, convey that information to larger computational networks. ATAP is currently distributing<sup>4</sup> prototypes among various tech developers who seek to integrate this technology into their applications in the coming months and years.

Project Tango is an example of what several critics have referred to as the “eversion” of the Internet. In a 2010 *New York Times* op-ed piece, William Gibson discusses the “genie-like” way that Google's algorithms constitute a participatory surveillance mechanism in that they simultaneously intuit a user's willingly supplied personality and behaviour traits and permanently store that information. He describes how emerging technologies have migrated the communications networks and data

---

<sup>3</sup> From the Project Tango website: “Imagine playing hide-and-seek in your house with your favorite game character, or transforming the hallways into a tree-lined path.”

<sup>4</sup> At the time of this writing, Project Tango is soliciting requests for its prototype developers “in the areas of indoor navigation/mapping, single/multiplayer games that use physical space, and new algorithms for processing sensor data” (“Project Tango”).



collection of the Internet into the physical world: “cyberspace, not so long ago, was a specific elsewhere, one we visited periodically, peering into it from the familiar physical world. Now cyberspace has everted. Turned itself inside out. Colonized the physical” (“Google’s Earth” n. pag.). Echoing this idea of the virtual “colonizing” the physical world, Marcos Novak likewise uses the term “eversion” to describe the “casting outward of the virtual into the space of everyday experience” (qtd. in Jones 32). As Steven E. Jones writes, Gibson and Novak are referring to the notion that the informational networks of the Internet are no longer representative of or constituted by “elsewhere”; rather, the ubiquity of networked devices means that the Internet has “everted,” or *become the condition* for everyday physical and social existence.<sup>5</sup> Jones points to “the rise of mobile computing” as the technological shift that facilitated just such an eversion (34), arguing that when banal technologies become networked, our everyday behaviours and interactions with both physical and digital entities come to exist in a blurred boundary between data and experience. Paradoxically, in this metric of eversion, the interface becomes a hyper-focused, almost intimate interaction between device and person (or thing, or environment) while at the same time retaining information captured by such exchange and communicating it among vast and largely unknown networks.

The eversion also indicates a shift in thinking about computer phenomenology. In the framework for human-computer interaction (HCI) first articulated in 1980s cyberpunk literature, cyberspace was “out there,” and, because it supplied a novel conduit for new modes of cognition and social communication, it was largely framed as a tool—albeit a

---

<sup>5</sup> To crystallize this point, Jones makes a useful comparison between Gibson and Novak’s use of “eversion” and Adam Greenfield’s term “everyware,” which describes a “‘paradigm shift’ around 2005 to ubiquitous or pervasive computing” (Jones 37).

nebulous and ineffable one—that facilitated human experience. However, with the eversion of cyberspace, the question regarding the computer’s experience of the world has come to occupy discourses in both design and criticism. James Bridle, an influential London-based artist, designer, and critic, has coined the evocative term “New Aesthetic” to describe an emerging aesthetic in which works of art and design reflect humans’ awareness of the ubiquity of computer vision and expression. As Bruce Sterling’s essay on the New Aesthetic suggests, the movement’s rhetoric is indicative of the eversion: “The New Aesthetic concerns itself with ‘an eruption of the digital into the physical.’ That eruption was inevitable. It’s been going on for a generation” (“An Essay” n. pag.).<sup>6</sup>

Of course, the eversion of cyberspace did not spontaneously happen, but rather is the outcome of numerous practices that produce and integrate networked technologies into banal artifacts and environments. In the context of Project Tango and the other mobile network technologies, this practice takes place in corporate-funded research and development labs for the mass-production and distribution of black-boxed<sup>7</sup> consumer devices. Yet these devices, while playing a fundamental role in the eversion of cyberspace (as Jones points out), nonetheless perpetuate an ocular logic— they draw our

---

<sup>6</sup> In the same essay, Bruce Sterling criticizes proponents of the New Aesthetic as having “weak aesthetic metaphysics”— that they mask a humanistic anthropomorphism of the machine under the guise of metaphor, a problem, he points out, because “computers don’t and can’t make sound aesthetic judgments” (“An Essay” n. pag.). Still, Sterling applauds the movement for inciting a conversation about design in the digital age that he is confident will lead to a substantial understanding of early twenty-first-century aesthetics in the future.

<sup>7</sup> I follow Bruno Latour’s employment of the term “black box,” which he in turn borrows from technologists, who use the term “whenever a piece of machinery or a set of commands is too complex. In its place they draw a little box about which they need to know nothing but its input and output” (Latour 2). In other words, black-boxed technology is stuff that is too complicated, too miniaturized, to be taken apart and analyzed by anyone without highly specialized knowledge and equipment.

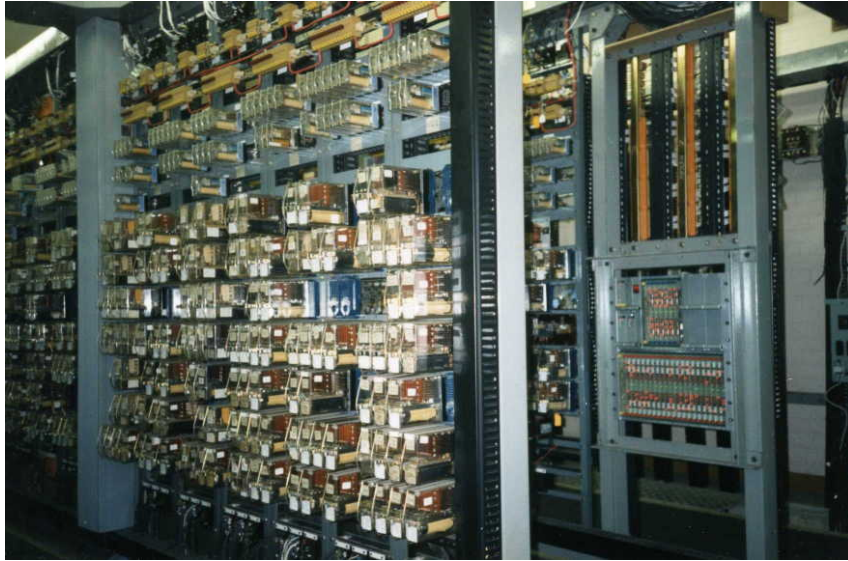
eyes to the screen, and represent their data visually. By actuating their processes through screen interfaces, they reify the screen as the privileged site of human-computer interaction (HCI). Accordingly, they still maintain a degree of tool-like instrumentality in our banal cognitive activities, though their enormous capacity to quietly gather data on operators<sup>8</sup> and environments, and their influence our daily lives both on- and offline, increasingly exceeds our understanding or awareness that such activities are taking place.

Primarily, what is overlooked by this increasing naturalization of everted technologies is the transductive material processes that comprise computation and interactive physical systems—in other words, the naturalization of everted technologies functions in similar fashion to the way that screen-based interfaces constitute computers as machines of transcendence. Transduction, broadly defined, is the conversion of one signal into another. While this term originates in the biological sciences, it can be taken up in the context of technoculture studies as a way of understanding, as Matthew Fuller puts it, the process of “how this becomes that” (85). Many electrical technologies easily demonstrate the cause-and-effect principles of transduction: a light bulb, for example, transduces electricity into light and heat, while a microphone transduces sound waves into fluctuating currents of electricity. Even early analog computers’ transductive processes could be somewhat observed and thus more easily grasped—latch relay switches were large enough that their switching mechanism could be seen or heard (the

---

<sup>8</sup> I use Alexander R. Galloway’s term “operator” in place of “user” throughout this essay, as this term more effectively conveys the manner in which “the machine and the operator work together in a cybernetic relationship to effect . . . various actions”—in this recursive relationship, “the action of the machine is just as important as the action of the operator” (“Machines and Operators” 5).

click of the switch was audible), and thus the operator was able to understand the material way a computer could perform, say, sequential logic operations (see Figure 2).



*Figure 2: A bank of latch relays in a “relay room” of an early analog computer.*

With the increasing miniaturization of digital technologies, physical transduction becomes increasingly complex and harder to observe without specialized equipment. Thus, the ways in which contemporary electronics and data processing actually work is frequently relegated to what is formally expressed on screens (Montfort n. pag.; Kirschenbaum 31). Accordingly, Matthew Kirschenbaum diagnoses the screen as a culprit in how the popular technological imagination has come to categorize computers as machines of transcendence. He points to the common perception that computing takes place on a symbolic level; this perception stems from the apparently non-inscriptive nature of electronic texts that can apparently disappear without a trace.<sup>9</sup> For instance, the

---

<sup>9</sup> Interestingly, the theory that underscores this phenomenon self-perpetuates itself through cultural conditions. Moore’s law—the hypothesis that the number of transistors

marks once visible on punch cards have receded from view, now impressed on hard disk drives installed inside the black boxes of personal computers. He argues that much of the critical discourse of the past twenty-five years has emphasized the formal aspects of materiality—that computers have come to be viewed as machines that convey a “technological sublime” (34). Transduction is collapsed into the aesthetics of output or display, as popular representations of cyberspace have dominated people’s understandings of computation during the last three decades, even if those representations mask or erroneously depict the particulars of computers. (Recall, for example, Fisher Stevens surfing through the insides of computers and networks in the 1995 film *Hackers*.) Such “screen essentialism” (Montfort n. pag.) is not entirely surprising, giving the astonishing evolution of ever-smaller, sleeker, faster machines that seem impossibly powerful and detached from any mechanical process. And with the proliferation of smart devices and touch-screens, the human-computer interface is now even closer to the screen (even the mechanics of the keyboard or mouse button have been replaced, symbolically rendered on the screen).

Kirschenbaum calls this dominance of popular representations over the material particulars of technologies a “medial ideology” (36), and he unpacks how it functions across media theory, film, and science fiction. Through a medial ideology, cyberpunk authors such as William Gibson and Neal Stephenson have focused their fiction on cybernetics—the study of communication and control in mechanical and biological feedback systems (Wiener 11–12)—as a way of exploring how a posthuman mind is

---

on an integrated circuit doubles every two years—forms the basis for long-term project planning in the technology industry. Lente and Rip refer to this phenomenon as the “self-fulfilling prophecy” of Moore’s Law (206).

constructed (or deconstructed) in relation to computational processing, networking, and memory. In cyberpunk fiction, the posthuman mind abandons “meatspace” and is uploaded to a seemingly immaterial, virtual world—a world of distributed cognition, where human and machine intelligence converges.<sup>10</sup> This trope, while often a concerted exploration of how our material bodies impact various subjectivities (Foster xix), nevertheless reifies the ephemeral realm conjured by an information-centric flattening of transduction. That said, it also sparks important conversations on the political implications of technological fetishes, as well as the recursive relationships between subjects and technologies in networked environments. Nevertheless, the lack of attention paid to the technical processes that facilitate interfaces tends to elide other questions related to the articulation of subjectivity in a materialist context: for example, when is the demarcation between subject and object, human and non-human, or operator and machine absent or messy? How are subjects produced via the particulars of a given technology’s materiality? Through, say, the intricacies of everted cyberspace?

These questions play a central role in the following pages. I argue that attention to the materiality of networked physical technologies reveals the nature of the recursive relationship between operators and machines within contemporary technoculture. With that position in mind, what is needed to reconcile cultural and political concerns with a particular awareness of the mechanisms and processes of the eversion is a technocultural framework that addresses both the conditions for the emergence of everted processes and

---

<sup>10</sup> N. Katherine Hayles points to the research in neurophysiology, anthropology, and philosophy on distributed cognition and how, “in analyzing how these extended cognitive systems work, researchers frequently draw on the cybernetic paradigm of recursive feedback loops, uniting components into dynamic and enactive systems that includes both human and non-human components” (Hayles 15).

the ways that those processes actually work to articulate subjects. Perhaps ironically, my inquiry centres on interfaces—sites where operators, machines, and various other entities detect and respond to one another—constituted by and through open-source, physical computing devices. Physical computing is a historically recent set of emerging practices and technologies that are central to “maker culture,” and the devices that emerge from these practices expand not only the computer’s range of actuation beyond traditional modes of interface but also the ways that the computer “senses” physical matter and processes. Thus, they push HCI beyond the logics of a screen.

Although the politics of how black-boxed devices (such as Project Tango) and seemingly transcendent computational processes impact the subject in an everted landscape is a pressing and important topic, I choose to focus on the HCI related to open-source hardware and physical computing because their cultural rhetorics tend to gloss the politics of the eversion and its effects on subjectivity, and instead emphasize the ludic potential of “making things” or the socioeconomic liberation of the individual from reliance on proprietary technologies. Existing discourses on the nature of interaction among things and people tend to emphasize the distinct difference or distance between humans and machines, and, in so doing, frame the interface as a dividing line, one that emphasizes the particular ontologies of humans (as in posthumanist theory) or non-humans (as in speculative realism). Yet, as mentioned above, the subject that gets articulated in and by the eversion requires a study that pays attention to the messy or blurred line between operators and machines itself.

In order to undertake this study, I turn to what Susan Leigh Star and James R. Griesemer refer to as a “boundary object.” Arising from the need for a method of

mediation among heterogenic scientific communities, boundary objects “are objects which are both plastic enough to adapt to local needs and the constraints of the several parties employing them, yet robust enough to maintain a common identity across site use. These objects may be abstract or concrete” (Star and Griesemer 393). The boundary object is a physical mechanism through which physical computing technologies help articulate this technocultural framework. The boundary object for my inquiry is Arduino, a low-cost, open-source, reprogrammable microcontroller board that is ubiquitous across physical computing communities, practices, and objects. I treat Arduino as the boundary object that, first, calls into question the posthumanist boundary—called “distinct difference”—between objects and things and, next, reveals three interfaces, which map an everted subjectivity that is legible yet ironically bound up in hybridity, non-coherence, and paradox: the visual interface, or how we see the computer; the physical interface, or how the computer sees us; and the haptic interface, or the space where the notion of “distinct” embodiments between human and non-human becomes complicated as the physical parts of machines and bodies begin to overlap and become ontologically enmeshed. I detail these three interfaces at the end of this introduction.

The three chapters that follow each analyze how these interfaces work to articulate subjectivity by focusing on case studies drawn from physical computing practices and projects. In each case, I draw from the history of the device’s development, its technocultural impact, and the way that the subjective formations that are afforded in and through these three interfaces. In chapter one, I examine how Arduino emerged from histories of programmable manufacturing and miniaturized circuits to become a boundary object widely used in physical computing devices. Arduino’s particular combination of



attributes, combined with its relatively low cost and user-friendly interface, posit it as a device that enables non-experts to engage in physical computing practices and make devices that network physical and digital environments. As such, it is implicated in the discourse of access; yet, despite its common framing in maker rhetorics as a device that fosters the socioeconomically liberated individual—one able to act, according to Chris Anderson, as both “inventor and entrepreneur”—attention to the physical and haptic interfaces reveals the articulation of productive subjects that produce data for the machine as much as for themselves.

The two subsequent chapters focus on devices that were developed using Arduino as their respective microprocessors; both were chosen because they have notably raised the profile of open-source devices in the popular technical imagination. The case study in chapter two is the Replicating Rapid Prototyper, or RepRap, a desktop fabrication device that converts digital objects into 3D-printed, plastic ones. Like Arduino, the RepRap is a product of the legacy of programmable manufacturing and production, but it is also bound up in the logics of biomimetics—the study and application of biological processes in mechanical and computing engineering. As such, the machine’s interfaces spark a consideration of the overlapping ontologies of the operator and the machine, specifically in the context of their respective relations to source and output. Here, I use Walter Benjamin’s concept of the aura in the pre-mechanically reproduced work of art to argue that the operator’s immanent relation to the output articulates an expressive subjectivity, one that behaves in a similar fashion to the machine but that remains ontologically distinct.

In chapter three, I turn to ArduPilot, a device that can convert remote-controlled (RC) vehicles into unmanned aerial vehicles (UAVs), or drones. ArduPilot represents a departure from the legacies of manufacturing and production, and instead situates itself in the discourse and history of location devices traditionally employed by governmental and military interests. Yet in the hands of individual or private interests, drones articulate subjects less as state citizens and more in the context of social relations among humans and machines. I apply Sandy Stone's concept of "warranting"—the articulation of a subject via the linking of discursivities and embodiment through location technologies—to the HCI between the human and the drone. Here, the drone itself fulfills the criteria for a kind of subjecthood, and engages in a social relation between the locatable, or "fiduciary," subject (also Stone's term, inspired by the work of William Gibson). Importantly, the context through which that subject is warranted relies largely on the intent and abilities of the drone's operator.

In the concluding chapter, I explore how these case studies provide an avenue through which to consider how interface is less a border that demarcates discrete, different entities, and more a political space that allows for various subjective formations to take shape. I argue that the technocultural framework formulated through the three interfaces aligns with feminist theories regarding the oppressive and enigmatic construction of boundaries. In particular, I cite the work of Stone, Gloria Anzaldúa, Judith Butler, and Donna Haraway. Haraway's cyborg, "a cybernetic organism . . . hybrid of machine and organism" (65) provides an especially salient example of the kind of subjectivity constituted through the blurring and transgression of borders. Haraway writes that "the relation between organism and machine has been a border war," and elucidates

the emergence of the cyborg as “an argument for *pleasure* in the confusion of boundaries and for *responsibility* in their construction” (66). Haraway’s cyborg is a feminist figure because it problematizes the boundaries between “production, reproduction, and imagination”—in other words, between source, process, and output. I end by postulating the politics of a subject invested in the spatiotemporal presence of the physical world, one who both resists the ephemerality of medial ideological interpretations of technology, and instead cultivates a situated knowledge grounded in practice and attention to the immediacy of the physical world.

Before undertaking this study of particular devices and how they illustrate particular configurations of HCI in the eversion, it is necessary to understand the historical conditions for the emergence of physical computing, its politics, and how they gesture towards a need for a technocultural framework that does not currently exist.

### **“A Computer for the Rest of You”: Physical Computing and the Maker Movement**

The increasing public availability in recent years of cheap, powerful electronics (such as microcontrollers and RFIDs<sup>11</sup>), combined with the ascent of Internet support communities, has facilitated the emergence of physical computing. Broadly defined, physical computing is a practice among artists, technologists, hobbyists, academics, and amateurs that combines do-it-yourself (DIY) hardware hacking or modding with programming in order to create networked, interactive devices. Massimo Banzi defines physical computing as the use of “electronics to prototype new materials for designers

---

<sup>11</sup> Radio-frequency identification (RFID) devices transmit radio waves using electromagnetic signals.

and artists. . . . It involves the design of interactive objects that can communicate with humans using sensors and actuators controlled by a behaviour implemented as software running inside a microcontroller” (3). The methods and technologies of physical computing align closely with the open-source movement,<sup>12</sup> which means that, commonly, the barrier to access is lower than, say, the kinds of research taking place at ATAP. As such, physical computing has become an important practice for artists, technologists, hobbyists, academics, and amateurs.

Physical computing plays a fundamental role in the eversion of cyberspace because it revolves around facilitating computer interaction with the physical world beyond the screen. Dan O’Sullivan and Tom Igoe discuss how the practice of physical computing intervenes in the notion of the expanded capabilities of computers to access the physical world—in their book *Physical Computing*, they describe how the practice is invested in making a “computer for the rest of you”—that is, a computer that interacts with the human operator<sup>13</sup> outside of the constraints of the mouse, keyboard, and screen (xvii). Accordingly, physical computing extends the concept of the eversion beyond the augmented reality interfaces and data accumulation capabilities of personal mobile devices, and includes any networked digital device or process that interacts with physical materials or processes. Neil Gershenfeld writes about how “personal fabrication will bring the programmability of the digital worlds we’ve invented to the physical world we

---

<sup>12</sup> Open-source software, or “Free Software,” is defined by Christopher M. Kelty as “a set of practices for the distributed collaborative creation of software code that is then made openly and freely available” (2).

<sup>13</sup> While some critical discourses, such as Ian Bogost’s concept of “unit operations,” theorize the ways that non-humans also operate (and thus also interface with other units, human, machinic, or otherwise), I choose to focus on the human and follow O’Sullivan and Igoe’s particular emphasis on HCI, which, I argue, is the site where the articulation of subjectivity takes place.

inhabit” (24). Personal fabrication, or desktop 3D printing, is an example of eversion that involves a networked device’s relation to physical building materials and the production of artifacts, a topic that is discussed at length later in this essay. In *Shaping Things*, Bruce Sterling uses the term “Internet of Things”<sup>14</sup> to refer to the ongoing transformation of banal, inert objects into traceable, machine-readable—and thus historical—entities that he refers to as the precursors to “spimes,” or objects that can be traced in SPace and tIME—sustainable, information-rich objects that are poised to succeed an era of “gizmo” technology. According to Sterling, the Internet of Things began with the introduction of bar-coding into consumer goods and has evolved into the RFID-based tracking technology that is integrated into everything from products to pets.<sup>15</sup> This history also necessarily implies the proliferation of networked materials into physical space—the “turning inside out” of cyberspace and its emphasis on information. Elsewhere, designer Matt Jones also points out the way that the traits and behaviours of physical artifacts are increasingly coming to resemble those of networked digital devices:

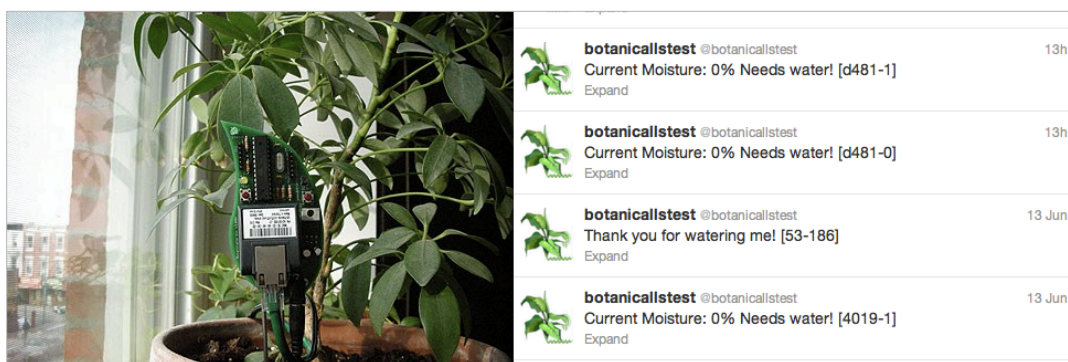
It’s getting hard to find consumer goods that *don’t* have software inside them. . . . This is the near-future where things around us start to display *behaviour*—acquiring motive and agency as they act and react to the context around them according to the software they have inside them, and increasingly the information they get from (and publish back to) the network. (“Gardens and Zoos” n. pag., original emphasis)

---

<sup>14</sup> “Internet of Things” was originally coined by Kevin Ashton in 2009.

<sup>15</sup> Sterling traces the ascent of the Internet of Things from the advent of bar-coding technology to what he calls “arphids,” a neologism for RFIDs and what he considers as “the seeds of Spimedom.” (*Shaping Things* 85–91).

Sterling and Jones both gesture towards the eversion of cyberspace in their descriptions of how the blending of physical artifacts with digital networks causes each to increasingly resemble the other. On the one hand, banal physical artifacts, when integrated with computational technologies, display something that resembles computational behaviour; they are capable of interacting with data from sensor inputs by expressing a response or reaction to that data. One such example is Botanicalls, an open-source moisture sensor that is poked into the soil of a household plant. The device communicates with wireless networks, sending updates on moisture levels and requesting water via its dedicated Twitter feed (see figure 3).

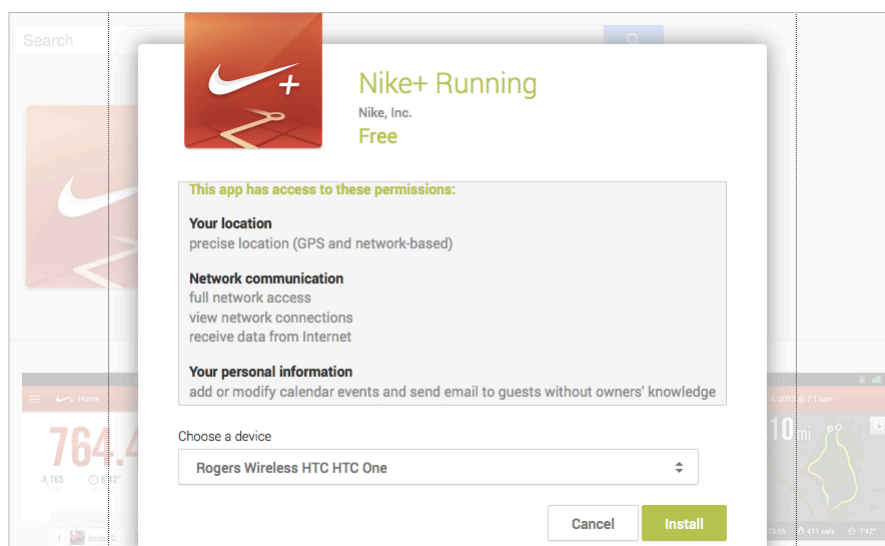


*Figure 3: Botanicalls sensor in a potted plant and a Twitter account for the device.*

On the other hand, when the ubiquitous personal computers and other networked digital devices—which were, as Gibson and Steven E. Jones suggest, “windows” into the “elsewhere” of cyberspace—are integrated into physical artifacts and environments, they gain access to an enormously expanded realm of information that can be processed, stored, and communicated across networks. In both cases, the end result is that objects are afforded a certain kind of agency within a network of objects, people, and processes—they are both less inert (in the case of artifacts) and less confined to the limitations of the digital realm (in the case of the device).

The eversion not only signals the arrival of artifacts that behave like computers; it also reconstitutes the physical world as machine-readable—in other words, it enables computers to gain access to the boundless data produced by physical bodies, objects, and their behaviours, thus rendering the actions of those things productive of value. The expansive range of a machine-readable world is especially germane where the human body is concerned. Steeped in Marx's work on sensual labour, Jonathan Beller's theory of the attention economy explains how attention becomes a capitalistic value-object through the act of viewing the cinematic or digital image—in short, seeing and clicking produce valuable information, such as through usage data obtained from an operator's Google searches or advertising revenue from Facebook clicks. With the eversion, Beller's sensual economy can be extended from a strictly visual domain to include the expressions of the entire body—in this model, the possibilities for rendering the intrinsic behaviours of the body as value-productive extends to the innumerable potential interactions between people and devices that record and share data. This notion of physical behaviour as at once machine-readable and value-productive can be observed in the user-agreement menus of, say, a “free” fitness application for a smartphone: the individual pays for use of the application by agreeing to provide data on his or her location, exercise patterns, and device usage habits, among other things (see Figure 4). Not only that, the device will enlist that data in the service of *making decisions for the user*—for example, by using that data to make personal calendar entries for the user (in other words, make a decision about that person's daily schedule), send invitations for others to join in exercise activities, and share personal information with others online. In short, the eversion of cyberspace indicates an expansion or acceleration of Beller's cinematic mode of production—the

degree to which the body becomes increasingly productive of data is welded to how computers can “see,” “hear,” “sense,” and otherwise reach out and touch us. Everted technologies that read and record bodies and their behaviours not only articulate people as the new big data; they expand the degree to which machines assume control over our daily decision-making, regardless of whether we are aware of those decisions.<sup>16</sup>



*Figure 4: A partial list of permissions that the user must agree to in order to use the “free” Nike+ Running application for the Android operating system.*

Open-source physical computing networks digital and non-digital environments, and therefore requires a mix of programming, electronics, and mechanical knowledge on the part of the operator. Such knowledge is gained through praxis—code must be written and de-bugged, schematics must be translated into hands-on circuit-building, and parts

<sup>16</sup> It should be noted that computational control precedes the eversion. Wendy Chun writes about how computers can be understood historically as modes of enacting governmentality, writing that “historically, computers, human, and mechanical, have been central to the management and creation of populations, political economy, and apparatuses of security” (*Programmed Visions* 7). I take up this point in the context of personal drones and the fiduciary subject in chapter three.



must be machined, cast, hammered, altered, welded, soldered, glued, sewn, and otherwise physically manipulated. As such, physical computing's investment in working with one's hands corresponds closely with "maker culture," a term that broadly encompasses the practices of people interested in building their own tools, devices, and interactive technologies. Spurred by ever-increasing access to materials, tutorials, and advice,<sup>17</sup> the maker movement is indicative of a watershed shift in the way that individuals and small groups are exploring the materiality of HCI with the independence, zeal, and creative spirit of the Whole Earth Catalogue subscribers of the 1960s, the Silicon Valley garage-programmers of the 1970s,<sup>18</sup> and the open-source software programmers of the 1990s and 2000s. Maker culture encompasses a vast range of practices, from hardware hacking to the development of new tools and prototypes to the adaptation of previously proprietary technology for private, individual use—practices that emerge from a shared investment in eschewing the black-box opacity of screen-based, proprietary technologies. This hands-on approach to better understanding and creating new technologies is broadly viewed within the maker community as a mode of personal empowerment; it both pushes back against the sometimes suffocating proprietary tech culture and facilitates a better understanding of how and where networked technologies impact our bodies, as well as

---

<sup>17</sup> The Internet is the primary resource for gathering such information, through the ever-growing number of DIY online file repositories (such as GitHub or Thingiverse), user-generated "how-to" websites (such as eHow.com or Instructables.com), online retailers (such as AdaFruit or Sparkfun), and the myriad message boards, forums, and blogs of people making, sharing, and talking about various projects.

<sup>18</sup> "The parallel [of the maker movement] with the hobbyist computer movement of the 1970s is striking. In both cases enthusiastic tinkerers, many on America's West Coast, began playing with new technologies that had huge potential to disrupt business and society. Back then the machines manipulated bits; now the action is in atoms. This has prompted predictions of a new industrial revolution, in which more manufacturing is done by small firms or even by individuals" ("More than" 3).

the things and events around us. Accordingly, a maker ethos is bound up in a techno-culture narrative of open access (see Figure 5) and affirmative responses to social, economic, and cultural issues.<sup>19</sup>



*Figure 5: “The Maker Bill of Rights,” one of many declarative statements that reflect the open-source values of the maker movement (Make Magazine).*

The discourses of the technical community of the maker movement and physical computing, not to mention the artistic, critical, and design-oriented community of computer phenomenology and the New Aesthetic, are each invested in the social impact

<sup>19</sup> That said, it is vitally important to note the vast networks of institutional, corporate, and governmental capital that make possible the conditions for an “open” maker movement. This point is especially pertinent given the culture’s heavy reliance on and integration with the Internet. For example, while the exchange of information (such as project ideas or designs) can appear to take place across seemingly immanent channels, such a rhetoric of “access” obscures the immense infrastructural networks of the Internet itself, which consists of physical elements (fiber-optic cables, physical servers, electricity), labour, and regulatory systems, all without which this access would cease. Alternately, makers inevitably purchase component parts from retailers, thus further relying on infrastructures of matter, policy, labour, and even resource management (i.e. the raw materials, often mined overseas, or the fuel required to transport materials) in order to participate in this affirmative culture.

of the eversion. Yet, while they focus heavily on the material particulars of the everted technology, they rarely pay much meaningful attention to its intersections with human subjectivity. The maker movement comes close, though its rhetoric is more invested in outlining a socioeconomic subject that is at once communitarian (sharing with and borrowing from the open-source community) and libertarian (practicing resistance against the cultural hegemony of proprietary goods and regulated services). Here, what often begins as a positivist discourse on individual empowerment and the egalitarianism of open-source projects<sup>20</sup> is subjected to a hermeneutics of suspicion of the maker movement. For instance, in his recent *New Yorker* essay, “Making It,” Evgeny Morozov attempts to historicize prominent entrepreneurs’ monetization of the products or services they create under the “maker” banner as evidence of a capitalist politics that exploits the fetishization of hardware—what he calls the “technical sublime”—that seduces people to the culture, where they can be easily exploited, with their interests capitalized by an emerging maker “empire.” Morozov’s argument stems from his critique that “technological solutionism”: the idea that digital technologies are sufficient to solve all of society’s problems in fact unnecessarily creates issues for the sake of creating technological solutions, and in the process, effect a dangerous naturalization of networked devices in everyday life under the pretense of techno-utopianism. Morozov believes that “for technology to truly augment reality, its designers and engineers should

---

<sup>20</sup> For example, Adrian Bowyer’s 2006 speech “The Self-Replicating Rapid Prototyper—Manufacturing for the Masses” describes how the biomimetic properties of “the self-copying and evolving RepRap [3D printer] machine may allow the revolutionary ownership, by the proletariat, of the means of production.” Bowyer calls his biomimetics-as-economics theory “Darwinian Marxism,” and suggests that, with access to raw materials (such as corn, that can be converted into plastic), RepRap “may preferentially allow the world’s poorest people to step onto the rungs of the manufacturing ladder” (“Philosophy Page,” n. pag.).

get a better idea of the complex practices that our reality is composed of” (*To Save Everything* 13). However, he reduces material practice and technicity to cultural function, and in so doing, overlooks the properties and affordances of technology as political artifacts in themselves<sup>21</sup> in order to focus on the conditions for the emergence of those properties. Morozov does not call it eversion, but he is clearly aware of the concept—his concern is that the proliferation of smart technology into banal objects sets the stage for the continued erosion of privacy and increased surveillance, subjugation, and exploitation of human bodies by corporate and state mechanisms. Yet his dismissal of maker culture, while illuminating the movement’s lack of attention to the cultural politics of capitalism, nevertheless fails to account for the ways in which the practices and outcomes of maker culture’s praxis-based model actually work to uncover the material elements of the eversion. Read this way, maker praxis does not naturalize technology; rather, it emphasizes a material understanding of the digital processes taking place all around us. Accordingly, Morozov’s position is indicative of an inverse symptom within political discourses of the eversion to those of maker ethos: there is a heavy consideration of the impact of everted technologies on subjects, particularly on the topic of surveillance and data-sharing—topics that are often sorely overlooked or glossed by makers—yet there is also a tendency to ignore or dismiss the material ways that technological processes of the eversion themselves work to articulate subjects. In short, Morozov’s hermeneutics of

---

<sup>21</sup> Langdon Winner’s influential 1986 essay “Do Artifacts have Politics?” take a social constructivist approach towards objects, arguing that artifacts intrinsically manifest (and are manifested by) social relations. Winner cites Long Island’s system of overpasses designed by Robert Moses in the mid-twentieth century as an example; his thesis is that Moses’s racist sentiments are manifested in the low clearance of his parkways, which were apparently built to prevent busses (and therefore poor people and people of colour) from accessing the beaches and neighbourhoods on northern Long Island.

suspicion is premised on a merely conceptual understanding of technologies, including technologies of the eversion.

As Banzi's definition of physical computing mentioned above describes, physical computing is about the integration of microcontrollers into objects. As such, it is bound up in a history of production and consumption, as the emergence of microcontrollers can be traced from these narratives, specifically those of manufacturing and the miniaturization of circuits for consumer goods. Such histories gesture towards a genealogy of how subjects become value-productive. Additionally, the history of manufacturing not only sets the conditions for the emergence of physical computing technology; it also gestures to a history of technological process beyond the advent of personal computing—how these technologies work, and how they have evolved as highly complex machines capable of integration in nearly any environment. The processes that emerge from physical computing praxis respond to and contrast with the increasingly “plug-and-play” standards of black-boxed technology; moreover, they dialectically reveal the ontology of the naturalized interface that such black boxes represent. Whereas the human-computer interface has, in the past several decades, increasingly become identified with the screen, the keyboard, and the mouse, physical computing works towards a de-naturalization of computation, foregrounding the materiality of transduction. This study of histories and transduction reveals a technocultural narrative of the human-computer interface—congealed around physical computing—that yields an evolved definition of interfaces. Physical computing recognizes that the interfaces of the eversion no longer begin or end at the screen, but constitute a complex mesh of overlapping configurations of matter and information that are somewhat mutually non-

coherent and paradoxical yet able to persist among one another. These configurations articulate bodies and computers as transduction points between source and expression.

The integration of microcontrollers into objects and environments via physical computing results in physical stuff made partly of code and capable of detecting behaviours, gathering ambient data, and algorithmically processing or actuating expressive reactions to that data via the material interface. In the context of HCI, physical computing is invested in cybernetic recursivity between operators and machines—machines and operators respond to each other, and each consequently experiences an alteration in behaviour or perception. This cyberneticized model for HCI makes it seem like a logical step to define a proposed technocultural framework as posthuman. After all, what is staked by the eversion is a model in which cognition or consciousness is neither detached from nor privileged over embodiment, but rather, as N. Katherine Hayles claims, one that reifies the “significance of embodiment” to both humans and machines (284). Through her careful consideration to the recursivity between embodiment and information, Hayles describes posthumanism as an articulated subject constituted by a cybernetic ontology—the result of a relationality between embodiment and information in which, “as with cybernetics, observer and system are reflexively bound up with one another” (284). Yet Hayles still draws a hard line between the two—she argues that “there is a limit to how seamlessly humans can be articulated with intelligent machines, which remain *distinctively different* from humans in their embodiments (284, emphasis added). In other words, posthumanism suggests the idea of interface as a boundary; thus, posthuman subjectivity is articulated through difference from the machines with which it

cybernetically interfaces. The delineation posited here, between human and machine embodiment, is useful for locating the posthuman subject in a cybernetic ontology.

This “distinct difference” between computers and humans has more recently been taken up in philosophical and speculative realist thought,<sup>22</sup> specifically in Ian Bogost’s theory of “unit operations.” Bogost contends that all components within a system or a network are “units”—discrete and isolated yet behaving in a way that allows that system to function. Conversely, units are themselves systems, and the units that allow for their functions are themselves ultimately discrete entities as well. In this way, objects can work within a system and yet retain some unchanged element that is ineffable to the other things with which it interacts. Bogost originally developed his theory in the context of video game studies in order to account for both physical and non-physical elements—a unit can be an executable code just as much as it can be a button, console, or a game-development industry—and has since expanded the theory as a way of interpreting the function of other media forms, as well as any object or system. Bogost contrasts unit operations with “system operations,” or “totalizing structures that seek to explicate a phenomenon, behavior or state in its entirety” (*Unit Operations* 6). In particular, Bogost observes two such dominant systems that organize meaning in contemporary culture: scientific naturalism and social relativism (*Alien Phenomenology* 13).

---

<sup>22</sup> Speculative realism takes its name from a 2007 conference at Goldsmiths College, London, between Ray Brassier, Quentin Meillassoux, Graham Harman, and Iain Hamilton Grant. Broadly speaking, speculative realism emerged from a shared interest among philosophers to reject what Meillassoux has termed “correlationalism”—the Kantian view that, as Harman puts it, “we cannot think of humans without world, nor world without humans, but only of a primal rapport or correlation between the two” (Harman 122).

Bogost's theories provide an entrance into a consideration of machine experience in both the physical and symbolic world, yet the unit's effects on subjects are glaringly absent (a point Bogost and other speculative realists would likely claim is exactly their point). For speculative realism, a "flattened ontology"<sup>23</sup> that equalizes the being and experience of all things necessitates the relegation of the subject. Indeed, by referring to an entity as a "unit" rather than an "object"<sup>24</sup> (such as is the more common practice among other speculative realists, most notably Graham Harman in his work on object-oriented ontology, or OOO), Bogost seeks to shed the entity's ontological relationality and posit it as a discrete thing that is fundamentally withdrawn from the system in which it functions: "The notion of the object also carries the timbre of a reference or relation to other things, as do grammatical predicates—a verb takes a *direct object*, on which it acts" (*Unit Operations* 5). Here, Bogost reinforces his claim that "unit operations are modes of meaning-making that privilege discrete, disconnected actions over deterministic, progressive systems" (1)—in short, while a flattened ontology works to reveal the limits of the human, the elimination of a teleological function precludes the possibility for subjectivation vis-à-vis objects such as computers or other networked or interactive machines. While such an inquiry is vital for shaking the foundation of Enlightenment metaphysics—one that reifies the centrality and privilege of human experience—speculative realism nonetheless elides the politics of subjectivity that must remain salient

---

<sup>23</sup> "Flat ontology" is a term originating in the work of Manuel DeLanda, who uses it to describe an ontology that refuses the hierarchical ontology of humanistic thought and instead articulates the single ontological category of "individual" for all entities (47).

<sup>24</sup> Bogost also has a more pragmatic reason for avoiding the term "object," which, as he points out, has a particular meaning in the field of computation (*Alien Phenomenology* 23).



in a post-Kantian landscape. What is needed is a study of the political and ontological relevance of the eversion to emerging forms of subjectivity.

Although Hayles's posthumanism and Bogost's unit operations are both invested in interpreting the limits and conditions of possibility for humans and objects in the current technocultural moment, they each effectively reify the interface as a space that demarcates discrete entities. This difference has the effect in both approaches of defining object-experience as something that is fundamentally withdrawn, and in turn authenticates feelings of anxiety and awe, not to mention a fetishized outlook on non-human entities. For both, the interface functions to separate entities; thus, although each theory is useful for understanding the ontology of people and things, they are nonetheless insufficient models for studying the eversion. What I propose instead is that the eversion of cyberspace and the ways it articulates subjectivity is best explored through the context of interface itself—I wish to examine this site as a space in which the discrete boundaries of things and people are transgressed in order to understand the mediation between people and machines. Because the interface in an everted landscape exists at the border of the “distinctly different” forms of human and machinic embodiment and can itself be defined as both informational and embodied, it constitutes a technocultural formation through which the eversion can be studied.

### **Three Interfaces**

Cybernetic modes of production and mediation, as well as the transductive properties of computation, are concerned with access to both operation and expression (such as productive output or clear message transmission)—or, put another way, direct attention at the levels of matter and information. Through Arduino, three discrete yet overlapping

interfaces can be studied, each representing a set of conditions for access and operation: 1) the visual interface of the computer screen, alternately understood as the “user-friendly” interface, which facilitates the access of information for the operator through metaphor and immediate visual feedback; 2) the physical, or non-visual, interface, which engages the question of how the computer accesses the physical world and understands its relation to that world; and 3) the haptic interface, or the interface that allows operators and machines to experientially interact with one another—an effect that can be partially described as “thinking with your hands” (though in this function, machines are also able to “think” with their physical sensors). Importantly, these interfaces do not always cohere with one another; rather, they each articulate a particular configuration of the oblique and recursive relationship between information and materials, the abstract and the particular.

### **Interface 1: The Visual Interface, or How We See the Computer**

While the focus on this essay falls primarily on how Arduino facilitates the rise of programmable objects, it is important to understand how the device remains invested in a visual, user-friendly interface—after all, it still needs to be programmed in order to facilitate any kind of interaction with different environments. In order to make the Arduino more accessible to people without expertise in computing or engineering, its developers relied heavily on visual interfaces and other abstract modes of HCI. In 2001, Casey Reas and Benjamin Fry wrote the integrated development environment (IDE)<sup>25</sup> that ultimately informed the construction of Arduino four years later. Their IDE, which they named Processing, was a coding environment for generating digital visual graphics

---

<sup>25</sup> An integrated development environment (IDE) is a piece of software that integrates various tools for coding and program compiling and implementation. In the case of Arduino, the operator writes or pastes a “sketch” into the IDE, which then compiles the code and sends it to the microcontroller as a set of directions for operation.

and designs. Visual artists themselves, Reas and Fry set out to create a programming language that could be easily learned by designers and artists, with an emphasis on writing “sketches” (or programs) for interactive graphics (Reas and Fry vii). In his writing and comments about Processing, Reas is clear that visuality is the motivating force that guided the language’s development: “The focus is on writing software within the context of the visual arts” (Shiffman n. pag.). Operators receive information from the program via visual representations on the interface, and in turn create a sketch, which, in Reas and Fry’s words, is akin to drawing ideas on paper (Reas and Fry 2). Reas and Fry articulate this position in their guidebook, *Getting Started with Processing*, unpacking how the IDE—beyond acting as a simplified coding platform—was also developed as a learning tool, which emphasizes a visual mode of learning as a way to encourage designers’ further exploration with coding languages: “Processing offers a way to learn programming through creating interactive graphics. There are many possible ways to teach coding, but students often find encouragement and motivation in immediate visual feedback” (1).

Such a view—that engagement with a user-friendly, programmable system is reinforced through “immediate visual feedback”—is echoed elsewhere by Arduino developers and practitioners. In *Making Things Talk*, Tom Igoe cautions his readers to remember the operator end of the interaction: when creating interactive design projects, it is vital to “give some indication as to the invisible activities of your objects” and build indicators such as “an LED that gently pulses while the network transfer’s happening, or a tune that plays” (47). According to Igoe, people using a device or interacting with a

system do not need to know what is being communicated—or how this becomes that—at all points. But they do need to be aware that communication is taking place.

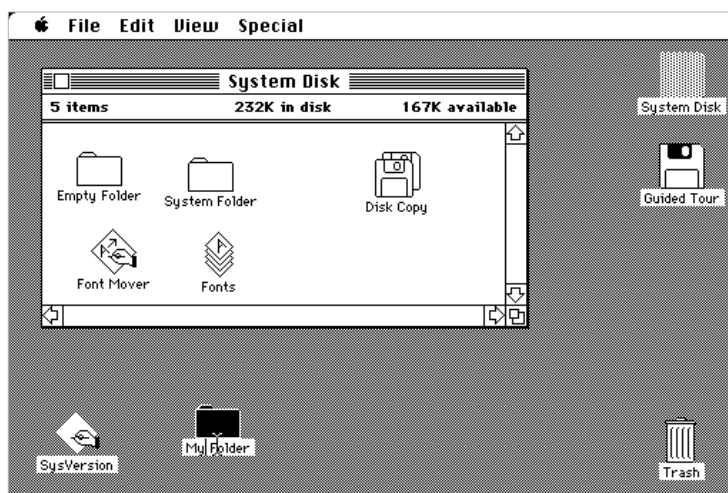
At the core of what Reas, Fry, and Igoe suggest is that operators remain invested in the process of computing—that when a given function’s invisibility is reified as a physical mechanism, people become aware of (and presumably invested in) the network’s communications. Thus, such mechanisms are rendered both knowable (in that we are aware that they are happening, and we are told that they are taking place) and unknowable (in that we do not actually know how the process is taking place). This making visible the invisible perpetuates a sort of fetish: the physical indication of an otherwise invisible software process constitutes in operators a sense that the unknowable functions of software have tacitly exposed themselves in a way that surpasses the mode of representation (i.e., light, text, or sound). The pulsing LED or signaling song is a fetish-object: it provides the operator such privileged access.

While, through Arduino, process indicators can ostensibly free operators from the visual domain (e.g., a “tune that plays” makes processes knowable via auditory conveyance), such sensory diversity is nevertheless tied to the visual insofar as what is knowable is metaphorically expressed in visual terms (e.g., Igoe’s “invisible activities of your objects”). Generally speaking, unknown information is rarely described as, say, “silent” or “unheard.” In the specific context of software and computing, Wendy Chun argues that people’s relationships with personal computers (specifically the software processes represented via the interface) are necessarily contingent upon metaphors of visibility, and that those metaphors practically define people’s epistemological experiences with computers. Building on the work of George Lakoff and Mark Johnson,

she claims that “metaphors govern our actions because they are also ‘grounded in our constant interaction with our physical and cultural environment’” and, furthermore, that “metaphors do not simply conceptualize a preexisting reality; they also create reality” (*Programmed Visions* 56). To return for a moment to the previous examples of personal computing graphical user interfaces (GUIs),<sup>26</sup> the visuals are almost always grounded in metaphors (e.g., a window, folder, or desktop) that situate or orient their operators, and—by extension—metaphors of visibility become central to understanding the construction of schemas within computing systems (see Figure 6). When programming an Arduino, this continued adherence to the visual persists because the construction of a schema for translation and communication is written in the IDE (which is based on Reas and Fry’s Processing). As most software interfaces do, this IDE functions according to a metaphorical relationship—anchored in visual paradigms—between operator and machine. While such paradigms are conducive to rendering technologies friendly to operators without expertise in computing or manufacturing, they also reduce the complexity of technological processes, mask or reify them, and curb the range of critical or creative approaches.

---

<sup>26</sup> A graphical user interface (GUI) is a screen-based interface that enables an operator to interact with the computer through the manipulation of metaphorical, graphical representations (such as icons, folders, buttons, and so on) of computational processes and applications.



*Figure 6: The Mac OS for the Apple Macintosh 128K, the prototypical GUI that uses representations of office productivity—such as folders, files, and trash bins—to facilitate a metaphorical relation between the operator and computer operations. Such representations mask the complex processes taking place within the machine.*

Yet the proliferation of such interfaces has also resulted in what Chun describes as the empowerment of operators, whose ability to directly manipulate and engage with computational processes afforded by the GUI “offers [them] a way to act and navigate an increasingly complex world” (176). Interestingly enough, empowerment remains tied to the ability to manipulate and engage with process, though the process has grown increasingly mediated by automation or digitization over time, from manipulating the transduction that enables automation or analog computation (e.g., flipping relays in the 1950s) to attending largely to screens (e.g., writing a sketch in Processing). In other words, the emphasis has gradually shifted from a hands-on interface with electronics to a visual and arguably abstract mode of HCI. Such screen essentialism results in precisely such a mode, which privileges the visual display of information while obscuring the balance of a platform’s processes, hardware and electronic circuitry included.

To be sure, many aspects of the Arduino platform are subtended by this kind of essentialism, especially where programming the microcontroller board is concerned. For

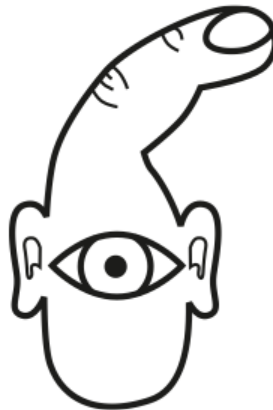
instance, operators who lack programming knowledge can simply copy and paste sketches from repositories such as GitHub into Arduino's IDE software, and then push those sketches to their boards. From Reas and Fry's perspective, this accessibility to machine function is precisely the point: it results in a kind of operator empowerment. Visual artists and other practitioners can get their microcontrollers working and can program behaviours even if they cannot explain how their program is actually running. Plus, even if they do hand-code their sketches, they will never be able to perceive or fully account for everything at work in the platform anyway—hence Igoe's insistence on an awareness of communication over its technical particulars. In other words, while the visual interface renders matter subordinate to informational processes, it nonetheless facilitates a wider epistemological, cybernetic relationship between operator and machine—the operator is at least aware of the machine's processes, and can respond accordingly.

### **Interface 2: The Physical Interface, or How the Computer Sees Us**

Despite their partial reliance on the screen, physical computing devices such as Arduino, RepRap, and ArduPilot interrogate user-driven paradigms and screen-based logics by integrating the analog environment into the computer's network—sensors that attach to the platform can, for example, read the surrounding environment as data. This expansion of HCI affects not only the way that operators engage with computational technologies, but also how computers interact with the physical world—thus paradoxically affecting a naturalization of computing and manufacturing technologies while broadening access to their functions in ways akin to Mark Weiser's vision for ubiquitous computing—a near-future in which computers that function invisibly all around us facilitate complete

naturalization of the interface (Weiser 104). From this naturalization, an interface emerges that is primarily reliant on the behaviours and states of matter (both organic and inorganic), which in turn facilitates a broader range of potential interactions between operators, computers, and things. Ironically, this expansion of interactivity among non-digital agents is contingent upon an interface that emphasizes a computer’s access to analog environments as well as people’s embodied actions.

For instance, in Dan O’Sullivan and Tom Igoe’s *Physical Computing*, consider the section titled “How the Computer Sees Us.” There, the authors construct an image of operators from the perspective of desktop computers—a perspective that interacts with its human counterpart through non-visual means: “a computer’s image of human beings is reflected by its input and output devices. In the case of most desktop computers, this means a mouse, a keyboard, a monitor, and speakers” (O’Sullivan and Igoe xix). Accompanying this description is a curious drawing (see Figure 7), consisting of one eye, one finger, and two ears.



*Figure 7: “How the Computer Sees Us” (O’Sullivan and Igoe xix).*



Here, a visual epistemology is read through a physical interface paradigm, a reversal of the typical relationship whereby people only perceive the computer via the GUI. This drawing not only offers an idea of how the computer “sees” us, but also how the computer perceives any object with which it communicates. Such a relationship relies on seeing as well touching, talking, and hearing.

O’Sullivan and Igoe’s point, then, is to articulate a problem that physical computing praxis seeks to resolve: how to make the computer “see” the “rest of us”: “We need computers that respond to the rest of your body and the rest of your world. GUI technology allows you to drag and drop, but it won’t notice if you twist and shout” (xvii). Although they write in the parlance of conventional, GUI-centric approaches to HCI—falling back on the metaphor of visuality that the computer “sees” our physical, embodied expressions—O’Sullivan and Igoe nevertheless shed light on the inherent constraints and limitations of the desktop computer’s interaction with non-digital environments (interactions that are typically limited to the computer screen and tools, such as the mouse and keyboard, that allow operators to manipulate the metaphorical representations of data on a screen). Their point is that the screen is for people, and that physical computing praxis is an exploration of what aspects of HCI are (or can be) for the computer. Returning for a moment to the question of access, physical computing makes computer access to physical matter its primary function, and the actuation of the resultant data secondary—a kind of epistemic reversal of screen-based HCI (where data expression on the screen is the primary means by which the operator access information).

As computers’ access to the physical world expands, they become more epistemologically aligned with matter; thus, the physical interface affords operators a

better understanding of the materiality of computational processes. As mentioned above, Matthew Kirschenbaum describes how, in new media criticism, the “history of codes” (30)—or the increasing miniaturization of inscription technologies in computing—is typically conflated with the withdrawal of information encoded at the level of the bit. Kirschenbaum’s response to a medial ideology is a turn toward “forensic materiality” (10) in new media studies, or a critical attention to the signatures, inscriptions, and hardware of platforms that cannot be flattened into “formal materiality” (10) of style, displays, software, and symbols. Such a shift redresses what he considers a turning away from the material histories of computing technologies; it also suggestively broadens the scope of media studies, highlighting how—as Chun, Fuller, Bogost, and other media studies scholars demonstrate—technologies and platforms are layered entities connected through a series of complex (and often ignored) transductions. In other words, the interface can still and should be mapped materially, beyond the advent of the screen and the reduction of information to bits. After all, the personal computer screen is a historically recent phenomenon, one that only partially tells the story of HCI—yet as can be seen in the histories of, say, military, manufacturing, or even prosthetic technologies, HCI is constituted by a material history of the intersections between people and machines.

The cybernetic function of microcontrollers in physical computing interfaces—specifically the feedback loop of sensors, software, operators, and actuation—also facilitates a broader discussion about access between humans and objects, objects and objects, and so on. In so doing, such interfaces resist the screen essentialism at work in the GUI. Jef Raskin, an interface designer who created the Macintosh project at Apple,

echoes O'Sullivan and Igoe's critique of the GUI. He claims that GUIs are not conducive to the cognitive processes at play in the way people work: "Human adaptability has its limits and . . . GUIs have many features that lie outside those limits, so we never fully adapt but just muddle along at one or another level of expertise" (Raskin n. pag.).

Arduino functions as a response to the limitations of GUI reliance by expanding the range of interaction. For instance, an operator can communicate with a digital environment via a Kinect sensor, an electret microphone, or a photoresistor, thereby exposing the ableism that informs most interface designs while also increasing the scope and capacities of computer perception. Consequently, physical computing devices not only reimagine how digital computers and the analog world might communicate; they can also reveal the ways we tend to think about computers and manufacturing through restricted metaphors, a reminder that a kind of Platonic "viewer paralysis" is tied to an epistemology predicated largely (if not entirely) on visual paradigms.

### **Interface 3: The Haptic Interface, or Thinking with Your Hands**

The third interface that can be articulated by Arduino, RepRap, and ArduPilot is that in which the operator and the machine interact at the material level, in real space. Here, the hardware of the device—its sensors, actuators, and other components parts (such as thermoplastics, in the case of the RepRep)—becomes manipulable matter at the hands of the operator, and the materiality of the body and matter is likewise accessible to the machine. What emerges from this interface is an ontology of how the material processes and expressions of both operator and machine begin to overlap in the physical space, thus calling into question the notion of some of the distinct differences between the embodiment of each, and further illuminating the particularities of their respective

recursive relationships between information and matter, or operation and expression. Arduino, RepRap, and ArduPilot all signal a partial departure from visual paradigms. As such, they gesture towards affordances in broadening the range of intersection between computers, operators, and objects characterized by the alternative metaphors of sense often found in physical computing research. While references to haptics are hardly metaphorical when it comes to building or assembling physical computing components, Banzi uses bodily metaphors when describing the experience of software. In *Getting Started with Arduino*, he observes that the Arduino community “developed ways of thinking with our hands,” and that Arduino’s IDE enables “constant manipulation of the software and hardware medium” (Banzi 5). Such haptic “thinking” especially occurs when Arduino operators navigate physical processes through a combination of programming with electronics, sensors, and actuators. For instance, when writing a sketch, operators must account for how bodies and embodied experiences are measured through temperature, units of time, or motion. Banzi also traces the outline of the body through some slippery metaphors of exploration, emphasizing the “pioneers” of Arduino and how someone who practices the “Arduino Way delights in the possibility of getting lost on the way” (5).

The haptic interface works to reorient the ontology of the subject through its dislodging of metaphors that privilege a clean distinction between human and non-human embodiment. A romantic emphasis on tactile engagement over metaphors of visuality marks a subtle yet significant irony in the development and use of such physical computing technologies: while the computational side of the interface conveys information to operators via visual representations, operators must still haptically engage

with or respond to such information. As mentioned earlier, given the integration of mechanisms such as infrared sensors and RFIDs into digitally networked environments, people may not always be aware of when they are contributing data, how that data is being processed, who or what that data is being shared with, or where an interface actually begins or ends. That said, the express interest in balancing explicit knowledge with tacit knowledge, or visual paradigms with tactile paradigms, or accessibility with ineffability, is one reason why Arduino, RepRap, and ArduPilot are suitable case studies for understanding the intersections of the eversion with subjectivity.

While the three interfaces described above are more or less present in most computing technologies (for example, personal desktops have physical components; video game systems increasingly utilize infrared and motion capture technology to expand the range of HCI; and even a car mechanic diagnoses many problems with a screen-based tool that senses the world), they are all brought to the fore through the particular qualities of physical computing technologies that integrate the three case studies mentioned above. Yet they remain ironically non-coherent to the effect that they each illuminate a discrete way in which information, embodiment, and embodied knowledge control and communicate with one another.

Arduino, RepRap, and ArduPilot all principally function to facilitate networked communication among computational devices and physical environments and objects. The outcome of this transduction is that the subject becomes interpellated as a media object. Following Lev Manovich's definition of new media as the "translation of all existing media into numerical data accessible through computers" (6), a new media machine is a device that is able to represent, interpret, and communicate media in digital

(i.e., data-based) form. Alexander R. Galloway expands on the particular way that the computer, as a media device, “remediates metaphysics *itself*” (20). That is, computers do not simply represent media in a new format; they remediate that media’s essential relation to its representationality. For example, computers do not simply codify film into a digital format (such as a compressed MPEG file); they expose the way the entire cinematic condition itself is influenced by the medium through which it is represented. (For instance, the viewing of film in a movie theatre affords a different epistemological experience than the viewing of film on a handheld device or laptop; it is more than just format, it is an altering of the essential condition of film as an ontological category.)

The three case studies are thus examined here as computational devices that re-mediate how the subject itself can be represented and experienced. As a boundary object, Arduino is a functioning media machine—the form of media it communicates is, among other things, an everted subject. It does so by both reading the body as a source of data and actuating behaviours reminiscent of the body. Paradoxically, it also interfaces with the very media it represents. The result of this recursive form of HCI thus exposes the way the ontology of the subject is influenced by the medium through which it is represented—for instance, the way that the body is rendered value-productive. In other words, through a recursive HCI, physical matter and expression become the primary source of data for both the machine (which interpellates the human body as expressive or productive of information) and the subject (who is afforded greater access to hardware through open-source technologies and practices).

Like Arduino, RepRap is a media machine, one that mediates the transduction of virtual objects into real objects. In so doing, it, too, invites a consideration of the

recursive relation between information and embodied expression, this time in the context of the production of physical artifacts. The device's operation is therefore pertinent to the changing nature of manufacturing, labour, design, and the socioeconomic aspects of HCI in an age of open-source hardware. Yet it also relates to human subjectivity itself, as the notion of machinic production is bound up in histories of human practice and experience. Cultural anxieties over the replacement of human workers by automated machines<sup>27</sup> can be historicized as an early form of HCI, one that ultimately exists in the same narrative as the ecstatic communication of the screen in the 1980s and 1990s and yet is no less important to understanding the eversion as the paradoxical and politically messy extrapolation of value from bodies and machines. In the case of RepRap, the production of objects by machines and humans, how each relates to source and expression, and how that relation articulates subjectivity can all be gauged by attention to haptic interfaces.

ArduPilot's mediation of human bodies is premised on its ability to locate bodies in physical space. Drones utilize computer vision technologies to render human bodies locatable media—bodies that provide information in the form of location, and through that data, infer their activity. In this HCI, human activity is evaluated in the context of location and indiscretion—computer vision is programmed to recognize that which stands out from its environment, be it the indicating features of a human face, the heat of a human body, the shape and motion of a vehicle, or the like. With the recent open-sourcing of UAV technology made possible through physical computing, and the resulting rise in the use of personal drones—which are often deployed by individuals to

---

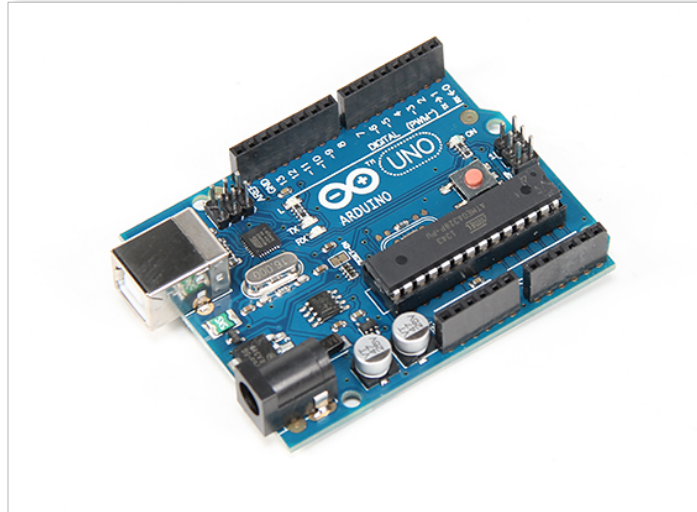
<sup>27</sup> David F. Noble's *Progress Without People* offers an interesting counterbalance to the notion of new Industrial Revolutions—his affirmative study of Luddism can be read as an argument against the kind of neoliberal techno-utopianism Anderson and others laud.

gather data on terrains and bodies for private surveillance or reconnaissance purposes — the issue of how the subject is articulated through such locative technologies shifts from a question of governmental technologies to one of social relations. In short, the question becomes: what kind of HCI emerges when bodies are articulated as locatable data sources by autonomous machines, and how does that computer vision articulate subjectivity?

The three interfaces that emerge from physical computing as the framework for welding various embodiments consequently bring into relief a particular mode of subjectivity that results from the eversion of cyberspace. While the scope of this argument could be quite broad, the examples chosen for this study—Arduino, RepRap, and ArduPilot—represent perhaps the most significant technocultural developments to emerge from physical computing thus far. They therefore provide an appropriate starting point for new ways in which to consider how everted technologies and the resultant HCI works to articulate subjectivity in the current technocultural moment. Moreover, these devices emerge from the historical development of technologies through automated manufacturing and miniaturization—histories that extend our understanding of computers beyond the advent of personal computing, cybernetics, and artificial intelligence. Put differently, this approach adds automated manufacturing to the corpus of media studies materials, and in doing so brings a material dimension beyond those articulated by the claims of screen-essentialist narratives, posthumanism, or OOO to bear on the discourses of computing and subjectivity.



## Chapter 1: Arduino, Manufacturing, and Productive Subjectivity



*Figure 8: The Arduino UNO microcontroller board.*

Variouly described as an open-source software and hardware package,<sup>28</sup> a prototyping device,<sup>29</sup> and an inexpensive tool designed to introduce non-programmers to the practices of physical computing, Arduino is a line of physical computing devices that consist of a software-based Integrated Development Environment (IDE) and a hardware component: a microcontroller attached to a circuit board with a USB port and header pin ports for connecting various sensors and actuators (see Figure 8).<sup>30</sup> Arduino was developed in

---

<sup>28</sup> See Gibb 8.

<sup>29</sup> See Banzi 6.

<sup>30</sup> At the time of this writing, there are currently 19 different models of Arduino boards available, all designed to suit a variety of prototyping needs. These range from the most popular Arduino UNO to the Arduino Mega (a larger board for projects with high processing power needs) to the Arduino Lilypad (a small, flexible version of the board designed to be sewn into clothing). The range of available devices can be viewed at <<http://arduino.cc/en/Main/Products>>.

2005 by a design team in Ivrea, Italy,<sup>31</sup> led by Massimo Banzi, who oversaw the development of the hardware component, and Casey Reas, who adapted Processing, an earlier IDE he authored (along with Benjamin Fry), for compatibility with the board. Collectively, the team was guided by a desire to make Arduino both easier to use and more affordable than its predecessors (Gibb 8). By writing a software “sketch” on the IDE, uploading it via USB to the microcontroller board, and attaching various sensors and actuators to the board, an operator can build a specialized tool that reads various physical phenomena via the sensors and converts the resulting data into a form of physical output expressed via the actuators. These expressions include physical actuation—such as blinking lights, emitted sounds, and rotating servomotors—as well as networked communication, such as short-message service (SMS) texts to a mobile phone or G-code instructions to a 3D printer. The board is small enough and cheap enough that it can be unplugged from the computer, connected to an external power source, and permanently installed inside the device it powers. Through Arduino, analog materials may be programmed and networked much like digital materials (Gershenfeld 3–4), and specific tasks (e.g., “watch this space,” “detect the temperature here,” or “send this message”) can be delegated to particular objects.

Invested in the creative use of microcontrollers, artistic design communities, hackers, and makers initially converged around physical computing in order to reduce the reliance on computer engineers in the manufacture of interactive exhibits, networked devices, and tools. In other words, physical computing practitioners share a common

---

<sup>31</sup> The device takes its name from Arduin of Ivrea, an important local historical figure who famously opposed German Roman Catholic rule in Northern Italy in the eleventh century (Arnulf n. pag.).

interest in making electronics prototyping, software programming, and manufacturing more accessible to non-specialists. The cultural tenants adopted by physical computing—open-source sharing, low barriers of entry, affordable materials, and a vast, distributed community of like-minded practitioners—have resulted in what *Economist* magazine (among others, especially in the maker community) recently declared as a “new industrial revolution, in which more manufacturing is done by small firms or even by individuals” (*Economist* n. pag.). Although this hyperbolic claim elicits justifiable skepticism about the politics of maker culture (including the issue of how corporations such as Google or even Home Depot could capitalize on open-source culture or exploit free-labour economies), it nevertheless accentuates the materials, practices, and metaphors at play in the Arduino community and how they intersect with the language and effects of a programmable interface, which—as I demonstrate in the following paragraphs—emerges from a longer history of programmable technologies in automated manufacturing.

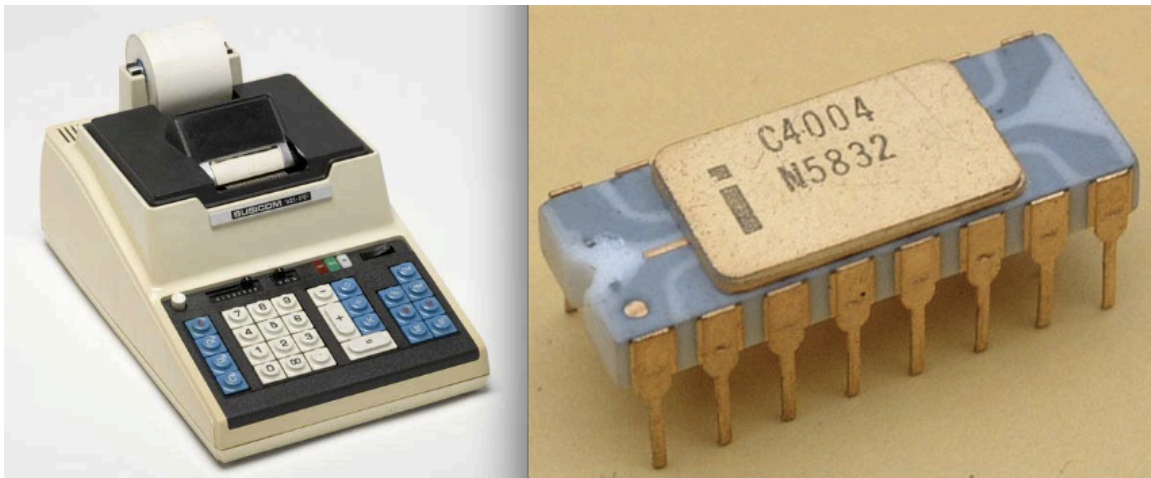
Arduino’s material history precedes the proliferation of personal computing in the 1980s—instead, the device’s emergence can be genealogically traced from the introduction of programmable technologies into both manufacturing and miniaturized digital consumer devices. The programmable microcontroller is a descendent of both the programmable logic controller (PLC)—a reprogrammable digital controller that allowed for the easier customization of automated industrial manufacturing machines—and the first microcontrollers used to miniaturize the digital circuits inside personal calculators and other consumer products in the early 1970s.

In 1968, machinist and engineer Dick Morley designed the PLC, a small, solid-state digital controller that used a simple ladder logic programming language (based on

the relay logic of analog circuits) to automate tasks on larger manufacturing machines. Morley had recently formed a company called Bedford Associates, which designed technologies for small-scale, machine tool firms that were making the transition to solid-state manufacturing technologies (Dunn 2008). Frustrated by the time and labour required to design unique controllers for each client, Morley created a computer that could be programmed to address the needs of particular projects. The PLC automated the industrial process by employing multiple input/output arrangements in real time. It was far more robust than the relatively delicate, relay-based industrial controllers that it replaced, and it was able to withstand the extreme temperature ranges, electrical noise, and vibrations of heavy industrial machinery (Dunn 2008). In addition to being solid state, the PLC's electronics were also housed in a compact, airtight casing, meaning the device could be mounted directly on a machine, thus eliminating the need for the room-sized housing cabinets required by the previous technology. Although, when compared with custom-built technologies, the affordability and small size of PLCs allowed them to rapidly become an industry standard, their programmability sparked a sea change in manufacturing. As Morley observed years later, manufacturing in the decades following the industry adoption of PLCs had largely transitioned from a "build-to-stock" to a "build-to-order" model (Morley n. pag.). PLCs allowed industrial production machinery to work more like machining tools because large automated processes could be programmed and reprogrammed according to the unique requirements of each product. The ladder logic language used by PLCs also meant that processes could be written and altered easily, with little to no programming involved. While still reliant on human

programmers, the PLC's malleability meant its interface with matter was ontologically responsive to and contingent upon the physical space.

The proliferation of PLC-automated manufacturing techniques coincided with the advent of early microcontrollers. In 1969, Intel (then a fledgling company) was commissioned by the Nippon Calculating Machine Corporation to develop a computer chip for its Busicom 141-PF printing calculator (see Figure 9).

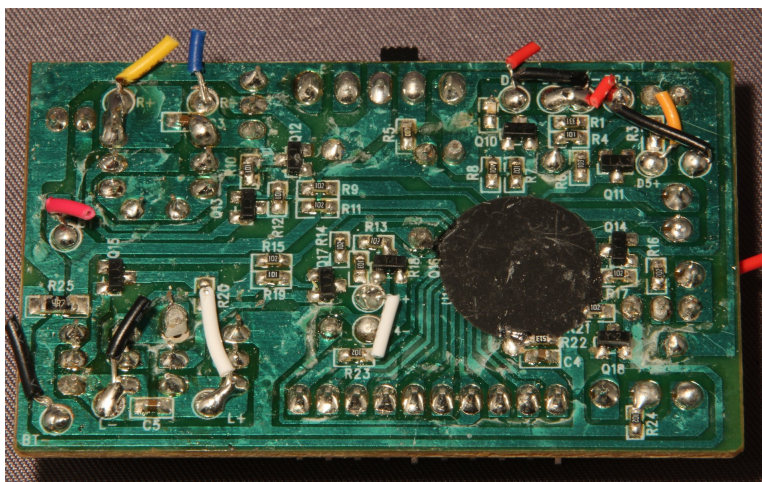


*Figure 9: Developed in 1969, the Busicom 141-PF printing calculator (left) was the first machine to utilize a programmable microcontroller, the Intel MC4-004 (right).*

Prior to the use of microcontrollers, electronic calculators relied on large, relay-based circuits to function, and were consequently heavy, prone to breaking, and consumed a high amount of energy. When compared with previous models, microcontroller-powered calculators were vastly more functional, far smaller, and able to run on far less energy. Similar to PLCs, the microcontroller's solid-state circuitry and small size led to an enormous range of applications. However, unlike PLCs, microcontrollers typically rely on complicated programming languages, and are thus less conducive to alteration or

adaptation for different projects. Today, microcontrollers are ubiquitous, appearing in nearly all electronic devices, especially simple or small ones.

The shift in production techniques, not to mention the programmable interface afforded by PLCs, combined with decreased costs and the increased miniaturization of the digital circuits utilized by programmable microcontrollers, set the stage for the integration of reprogrammable microcontroller boards such as Arduino into small, ubiquitous devices. That said, it is important to understand that, along with the improved performance and efficiency of these devices, came an increasingly obscured material process and thus a naturalization of digital technologies not possible with the bulkier analog technologies they replaced. For example, the relays and transistors that PLCs and microcontrollers replaced were—along with being bulky, inefficient, and liable to breakage—also analog technologies, housed in a room or large removable casing and accessible by hand. Their processes could be observed and analyzed, and nonworking parts could be replaced with relative ease, albeit through extensive diagnostic labour. Due to its miniaturized technology and enclosure within extremely strong materials (usually an epoxy resin that cannot be removed without highly toxic chemicals), a malfunctioning PLC or microcontroller would generally be deemed broken and then simply replaced with a new unit. The replacement and subsequent reprogramming or recalibration (especially in the case of PLCs) could prove both costly and time-consuming. Yet more importantly, such black-boxing—literal, in the case of epoxy-covered circuits (see Figure 10)—resulted in the exchange of access to material processes for increased productivity and operational versatility.



*Figure 10: A populated circuit board with an epoxy-covered integrated circuit (the large black “blob” on the board). The epoxy coating protects the circuit from corrosion, damage, and tampering, and can only be removed with corrosive materials.*

As the technologies used to automate manufacturing processes became naturalized, computers were epistemologically refigured as responsive machines that interfaced not just with the raw and outputted materials they manufactured into objects, but also with the operators who programmed them to produce these objects in a specific way. In open-source rhetoric and practice, operational versatility is somewhat conflated with access—since all source code is readily available (and editable), the array of possible applications seems to be endless.

Precisely because it both reads and actuates data, the Arduino board now acts as a principal microcontroller for many complex devices. Its popularity can also be attributed to the fact that Banzi and his team made its hardware open-source: the company’s website offers schematics for all of its manufactured boards, meaning those who do not want to purchase the board can build one themselves, with individually purchased components and materials.<sup>32</sup> To further increase the range of the hardware’s applications,

<sup>32</sup> It must be noted here that “open-source” hardware remains implicated in the larger materialist discourses of the costs of industrial practices in the production of electronic components. In this context, while Arduino has the look and narrative of “straight from

the Arduino development team created a number of different board sizes and specifications that make it suited for permanent installation inside any object it controls, such as a tweeting houseplant, a refrigerator that emails a user to buy more milk, or an espresso machine that can receive orders for drinks via text message, and then print custom messages onto the foam (see Figure 11).



*Figure 11: Textpresso, an Arduino-integrated espresso machine. The machine receives drink orders via text message and prints the order number into the foam using edible ink.*

While Arduino is by no means the first or most powerful microcontroller board on the market,<sup>33</sup> it forms the object of the current analysis because its specific and deliberate design model make it the most culturally relevant device of its kind. According to Banzi in his guidebook, *Getting Started with Arduino*, Arduino distinguishes itself from similar devices on the market because of a handful of features: it is a “multiplatform environment” (meaning it can run on any major operating system); it uses the Processing IDE, a relatively simple, Java-based programming language that already has wide usage

---

the source,” the fact that its component parts (integrated circuits, resistors, copper-etched circuit board) may originate in parts of the world with substandard industrial and labour laws makes such open-source hardware remarkably similar to the oppressive labour conditions and negative environmental impact surrounding the production of, say, Apple products in the Foxconn factories near Shenzhen, China (Parikka n. pag.).

<sup>33</sup> See, for example, Arduino’s predecessor, the Wiring platform, which was in part developed by Banzi, and also runs on the Processing IDE.



among artists and designers; it connects to computers using a USB cable (as opposed to a serial port); both the hardware and software are open source; the hardware is relatively inexpensive (compared to similar products, such as the Raspberry Pi platform<sup>34</sup> or the Wiring board); and “there is an active community of users”<sup>35</sup> (1–2). In other words, Arduino has succeeded in dominating the maker market for microcontroller boards because of a particular recipe that combines various socioeconomic and physical properties: again, it is affordable, accessible (both in terms of its open-source format and its easy-to-learn IDE), community-driven (due to its open-source design and its vast repository of public knowledge), and physically suited to fit a wide range of applications.

It is crucial to emphasize that all of Arduino’s features were deliberate choices made on the part of the design team, because the device’s combination of properties enables it to play such an important role in the everted landscape. For example, according to Banzi, the diverse physical dimensions, power, and connectivity options among available Arduino boards, combined with the low cost of each unit, were intentional design outcomes made for the purpose of encouraging users to both take greater risks in designing their prototypes and to consider the device as a building block (rather than an external terminal) of the finished project (Banzi 5). Simply put, a smaller board requires

---

<sup>34</sup> While the Raspberry Pi board costs roughly the same price as an Arduino, it requires a dedicated mouse, keyboard, monitor, network connection, SD card with an OS running on it, and other components, whereas the Arduino simply requires a USB cable to connect to a computer. Arduino therefore constitutes a less expensive investment.

<sup>35</sup> This community driven aspect of Arduino is described on the Arduino homepage:

A core strength of Arduino is the community of users who develop Arduino projects, contribute ideas and code to the Arduino project, help answering questions, and provide valuable feedback to the Arduino developers.

The community of Arduino enthusiasts is vast, and includes region specific groups and special interest groups. The community is an excellent further source of assistance on all topics such as accessory selection, project assistance, and ideas of all sorts. (“Arduino” n. pag.)

less voltage to run (meaning less potential for damage if something goes wrong), while a cheaper board means that, if it does get fried by a jolt of electricity, it is inexpensive to replace (hence the increased opportunity to take risks in design—a more expensive unit might lead to a more conservative design due to budgetary considerations). Likewise, a smaller, cheaper board means that it can be built directly and permanently into the project (the boards are cheap enough that each project can have a dedicated Arduino running inside of it), a feature that affords greater creative freedom in project design. All told, such decisions make Arduino a tool built to cultivate risk-taking and expanded creativity, qualities very much in line with the socioeconomic figure of the “maker.”

As another example, consider the various ways that Arduino’s community-driven model (both in shared knowledge among users and open sourcing of project resources) increases the potential for creative applications of preexisting projects and allows for new possibilities of project interoperability. In other words, with enough digging into the various message boards, YouTube videos, and blogs created by the Arduino (and wider maker) community, operators can find both precursors to their own project ideas and on-the-fly training on *how* to adapt existing designs to their own goals and particular budgetary and material limitations—by tinkering with lines of code that someone else wrote or with capacitance values of someone else’s project—and by learning how to do so from the same community that provided these blueprints—operators are able to work around the kinds of constraints that both proprietary systems and previous, non-community driven platforms place on open-ended development. As well, projects can be made to communicate with one another, across media, format, and other physical differences. Such allowance for interoperable systems (or, in Tom Igoe’s words, “making

things talk”) suggest that the emerging technocultural moment is indeed that of an Internet of Things.

In *Makers: The New Industrial Revolution*, Chris Anderson demonstrates the effectiveness of the community-driven model by describing a project in which he seeks to use an Arduino board to improve a sprinkler system that his grandfather invented in the 1960s and subsequently sold to commercial enterprise. Anderson’s goal in this exercise is to underscore the “maker revolution” as something that provides the means for designers to become entrepreneurs; he claims that, in his grandfather’s time, selling designs to large firms was the individual’s only means of distribution for his or her product. Anderson decides to modernize his grandfather’s sprinkler by giving it network capabilities; he settles on a sprinkler that can be controlled remotely using an application on his smartphone. To do this, he surveys the base of knowledge by reading message boards to learn the possibilities for the scope of the project and also *how* to make it. He finds within the larger community of makers and open-source coders the inspiration, knowledge, and training needed in order to create a project of which he had no previous understanding or conception. In short, the community provided the conditions through which his innovation can be developed.<sup>36</sup> By fixating his attention on the different bases of existing

---

<sup>36</sup> Anderson’s point is to underscore the economic power of the online community. Referencing his economic theory of “the long tail,” Anderson explains how the proliferation of digital goods and services has led to a cultural shift towards niche online markets. According to Anderson, retailers such as Amazon succeed because their business model allows them to sell niche market items just as easily as big-ticket items. Open-source design communities such as the one he utilizes in his sprinkler anecdote function according to the long tail in that they provide the means of accessing goods (such as open-source code for powering the sprinkler, or .STL files for 3D printing sprinkler components) and services (information and help from message boards) according to any niche demand. Despite his affirmation of the radical power of the long tail, the politics of this model still gesture towards the reinforcement of global,

knowledge, he is able to connect his own work to that collective body. Once completed, he adds his own refinements to the community's repositories (in the form of both project documentation and open-source code ), and thus makes them discoverable for future individuals to access and repurpose.

### **The Productive Subject**

While Anderson's anecdote provides an effective example of how the growth and increasing sophistication of the Arduino community owes to the preexisting levels of technocultural knowledge that spur new ideas—which in turn, become embedded as new layers of knowledge—it nonetheless falls short of articulating what the implications of Arduino and the physical computing technologies of the “maker revolution” are for subjectivity. Anderson's do-it-yourself narrative is common among others that emerge from maker culture—the movement's “revolutionary” aspect typically champions the socioeconomic virtues of open-source communities and hardware for the individual with entrepreneurial aims. In doing so, the question of what kind of subject is formed in relation to everted technologies is subtended by what can be read as a reified capitalist narrative. In fact, Anderson's maker figure—who is both “inventor and entrepreneur” (Anderson 7) and has nearly total access to resources—can be read as an idealized neoliberal capitalist subject: maker culture cuts a pathway to pure individualism, freedom from socioeconomic networks, and the ability to produce goods and services without much regulatory obstruction. Ironically, although physical computing practices

---

monopolistic corporatism: As Anderson explains, “manufacturing has now become just another ‘cloud service’ that you can access from Web browsers, using a tiny amount of vast industrial infrastructure as and when you need it. Somebody else runs these factories; we just access them when we need them, much as we can access the huge server farms of Google or Apple to store our photos or process our email” (66).

foreground the transductive elements of everted technologies, the individualistic nature of the popular socioeconomic figure of the maker largely divests such a figure of the kinds of subjectivity that get articulated via interaction with networked technologies.

Against the subjectivation articulated by this socioeconomic figure, a more metaphysical way of exploring the subjectivity articulated in and through Arduino is to consider how physical computing technologies that arise from the histories of manufacturing and digitized computation construct a productive subject, not in a socioeconomic (i.e., revenue-generating) sense, but rather in a data-productive manner that stems from the cybernetic, physical processes foregrounded by the eversion. In short, the programmable devices that constitute the eversion of cyberspace also work to render subjects as value-productive in their embodied behaviours. This understanding of subjectivity problematizes Anderson's maker figure—who enacts the politics of C.B. Macpherson's "possessive individualism"—because, given the embodied nature of subjectivity, the technologies of the eversion work to erode the concept of subjective "possession" of the body and its productive values. Another way of understanding this value-productive yet non-possessive form of subjectivity is by extending Jonathan Beller's notion of attention economy beyond attention: in a cybernetic relationship in which machines can read the data generated by the operator at any time—not necessarily just when the operator is actively engaging with the machine—productive value happens beyond the eye and hand, extending across the entire body. The Quantified Self movement<sup>37</sup> provides a helpful illustration of this dynamic. The idea that "self-tracking"

---

<sup>37</sup> Quantified Self uses banal technologies, such as phones, watches, and step-counters to record and monitor data from a user's own body. This is often done to glean information

one's embodied practices is a mode toward self-improvement means that data must sometimes be gathered without the conscious attention of the operator. In other words, while such data might be offered willingly in the name of self-improvement, the networks through which it is shared (as well as the extent to which data can be collected) exceed the operator's awareness, hence the slippage of the belief that one "possesses" oneself. In the eversion, your body routinely and casually produces value, as discrete data, for other entities.

The historical development of programmable devices underscores the material specificities of a technocultural subject because it extends beyond the logic of Fordist production that, from a technological perspective, arguably realized its true economic potential with the introduction of industrial robotics in the 1960s.<sup>38</sup> Importantly, the narrative of programmable manufacturing—that of a technologically mediated, non-static interface that both facilitates and is conditioned by the interplay of operation and output—also situates the emergence of the microcontroller in the context of the transformation of human bodies into value-productive entities through feedback loops with the reprogrammable, intelligent devices that are ubiquitous in post-Fordist, networked conditions. Given how programmable machines emerged from the needs and constraints of both manufacturing and computation, productivity is linked simultaneously to access and expression across physical and informational levels. PLCs were built as a mode of expanding productivity in material manufacturing, but they were also designed as productive solutions to the issue of customization needed in manufacturing. In short,

---

on biorhythms or health issues, to monitor athletic performance, or, increasingly, to gamify everyday physical activities.

<sup>38</sup> The Unimate was the first patented industrial robot, and was installed in a General Motors factory in Ewing Township, New Jersey in 1961.

what facilitates access, and thus productivity, are boundary objects such as Arduino: objects that are capable of—to borrow a term from linguistics, feminism, and the work of Gloria Anzaldúa—physical and informational “code-switching,” while remaining internally consistent.

The introduction of programming to both manufacturing and personal computing, and the subsequent proliferation of PLCs and microcontrollers, suggest a way in which the notion of “access” indicates the development of devices that obscure transduction processes in order to improve performance, programmability, and user-friendliness. In other words, in the history of digitized programmable technologies, operators have been required less and less to engage with the complicated processes needed to facilitate computational tasks. Process, with its clunky physicality, becomes de-emphasized as the fluid and more “productive” metaphorical interfaces of informational communication are privileged. Wendy Chun elaborates on this point in the context of GUIs, writing that “‘user-friendly’ interfaces have been key to . . . creating ‘productive individuals’” (8) and that “GUIs have been celebrated as enabling user freedom through (perceived) visible and personal control of the screen. This freedom, however, depends on a profound screening: an erasure of the computer’s machinations” (59). Indeed, the GUIs that have become de facto interfaces for productive personal computing<sup>39</sup> reveal this gradual

---

<sup>39</sup> The two most widely used GUIs (Microsoft Windows and the Macintosh Operating System) both emulate an established site of personal productivity: the office space. Through representations of desktops, folders, files, and trash bins, these two GUIs constitute the setting in which computational processes bear results.

transition from hands-on manufacturing in the 1960s and 1970s to a screen-based model for productivity from the 1980s forward.<sup>40</sup>

In order to make the Arduino more accessible to people without expertise in computing or engineering, its developers relied heavily on user-friendly, visual interfaces and other abstract, metaphorical modes of HCI. Operators receive information from the Arduino IDE via visual representations on the interface, and in turn create a sketch, which, according to Reas and Fry (who developed the IDE), is akin to drawing ideas on paper (Reas and Fry 2). As most software interfaces do, this IDE functions according to a metaphorical relationship—anchored in visual paradigms—between operator and machine. To be sure, many aspects of the Arduino platform have a stake in the visibility of screen metaphors, especially where programming the microcontroller board is concerned. Communication between operator and machine is represented on-screen, and processes are metaphorized in the visual domain as well (e.g., through the small progress bar that indicates the progress of a sketch as it is compiled and uploaded into a machine, and the blinking light on the Arduino that indicates when the sketch has finished uploading to the board). As I argue in the introduction to this essay, these paradigms are conducive to user-friendly technologies. But to reiterate an important point, they also reduce the perceived complexity of technological processes, mask or reify them, and curb the range of critical or creative approaches.

---

<sup>40</sup> Of course, this transition is not uniform and does not wholly abandon or disregard previous materials, practices, and concepts. It instead integrates them into emerging models and structures. For instance, “soft PLCs” combine the ladder logic of PLCs with the user functions of personal computers, including GUIs.



Despite its partial reliance on the screen, the Arduino platform does interrogate user-driven paradigms and screen-based logics by integrating the analog environment into the computer's network—sensors that attach to the platform can, for example, read the surrounding environment as data. The networks constituted by Arduino, bodies, and environments accord with the classic definition of the cybernetic system because they generate feedback loops by recursively responding to embodied information that has been filtered through its software and actuated in real or virtual space. Arduino thus facilitates an interface between these spaces, yet its programmable hardware remains functionally oriented toward its environment. Such configuration upends the posthumanist cybernetic model that procedurally maps information onto bodies, thus privileging cognition over embodiment in order to study the effects of information on embodiment, and instead emphasizes the importance of bodies and things situated in the physical world to cognitive (or informational) processes. Put pithily, Arduino foregrounds the roles of transduction in the relationship between body and mind, objects and information.

While the screen essentialist model dichotomizes the symbolic against the machinic (and also privileges the former), Arduino works against this model of the interface by underscoring the continuity of praxis (e.g., the writing of a sketch on the Arduino's IDE) with mechanical process. Additionally, as seen on the screen, code is functionally useless, and must be physically transferred to the device in order to control its hardware's behavior.<sup>41</sup> While this process is not a uniquely new development in the

---

<sup>41</sup> There is a degree of physical interaction involved here, such as the clicking and dragging of a mouse to open or move a window on-screen. Yet such motions have become largely naturalized, and tend to dissociate the physical action required from the informational results. A good way of historicizing this naturalized interface is by considering the case of Windows Solitaire, which was integrated into the Microsoft

history of personal computing (think dot-matrix printers), Arduino's pronouncedly accessible IDE and flexibility with regards to inputs and outputs opens up a unique vista of possible interactions between operators, machines, and digital and analog environments.

Returning, then, to the notion of “three interfaces” detailed in this essay's introduction, Arduino is a curious case study for examining HCI in the eversion. Through the visual interface, the operator produces the conditions for technological expression through the visual metaphors of the screen-based interface, and the machine produces behaviour via the cybernetic feedback loop of the physical interface. The visual interface provides a mode through which the conditions for technological expression emerge: the IDE extends the design environments previously meant for virtual actuation (graphics design, in the case of Processing) to physical transduction. Despite “beginning” at the screen, Arduino's orientation towards the physical domain means that the operator produces data *for* the extended networked mechanisms of everted technology. The data that forms this basis for computational interaction with bodies thereby renders bodies productive of data. Finally, the haptic interface is—like the visual interface—where the conditions for the cybernetic function are created, via the physical construction of the device. Also in the physical interface, the productive relation between operator and machine is enacted as a mode of access afforded by the translation of this into that—of data into actuation, or vice versa, as in the case of Quantified Self technologies, where machines convert a body's output into information. Yet the haptic interface is precisely

---

Windows 3.0 operating system in 1990 (and included in every subsequent version of Windows to date) as a tool for allowing new desktop users to familiarize themselves with the dissociative properties of a mouse (Levin n. pag.)

where embodied interactions between operator and machine take place. Through this interface, both Arduino and its user produce physical expressions that blur their respective ontologies. Put simply, the haptic interface is where the embodied states of operator and machine overlap. This ontological blurring becomes evident through two more case studies—the RepRap desktop fabricator and the ArduPilot UAV—that follow in chapter two and three, respectively.

## Chapter 2: RepRap, Handicraft, and the Expressive Subject

While 3D printing has existed since the mid-1980s, its recent rise in popularity is due in large part to the proliferation of open-source, consumer-oriented desktop fabrication devices such as the Replicating Rapid Prototyper, or RepRap. With the introduction of RepRap—an open-source, Arduino-powered, desktop 3D fabricator—the reprogrammable microcontroller has come full circle as a technology that has radically changed the possibilities for the modes, means, and practices of manufacturing. The articulation of the productive subject through the analysis of Arduino in the previous chapter is also present across the visual, physical, and haptic interfaces of RepRap, which provides a case study for understanding how informational patterns are expressed in material forms. Here, this productive subject becomes further articulated as an *expressive* subject in the context of the relation to source and output. Broadly considered, RepRap functions as a gateway to new modes of manufacturing, modes that fundamentally refigure manual and machine labour as well as interactions between the two. In the process, this technology invites the question of how handicraft becomes naturalized through microcontrollers and their attendant interfaces. In the case of RepRap, the machine's operative and expressive processes arise from forms of transduction that mediate between the immanence of the source material, the executability of its software, and the ineffability of its own expression.

The recent proliferation of open-source desktop fabricators has resulted in the term “Industrial Revolution” being used as a common rhetorical theme through which 3D desktop printing is discussed. The device is perhaps the most frequently cited example of

how open-source hardware has the potential to radically disrupt the socioeconomic imbalances brought about by post-industrial capitalism. For example, in 2012, Massimo Banzi began his TED<sup>42</sup> keynote address by discussing how his friend had recently “printed” a toy car for his son using a 3D printer and an open-source design. Banzi’s anecdote introduces the radical potential of the open-source movement:

This idea—that you can manufacture objects digitally using [3D printers]—is something that the *Economist* magazine defined as the “third Industrial Revolution.” Actually, I argue that there is another revolution going on, and it's the one that has to do with open-source hardware and the “makers” movement. Because the printer that my friend used to print the toy is actually open-source. (“Massimo Banzi” n. pag.)

He goes on to explain that “at the heart of this printer there is . . . this Arduino board: the motherboard that powers this printer” (n. pag.). In other words, Arduino is a fundamental part of this particular technology (the 3D printer), which is just one example of the way that reprogrammable microcontrollers and the devices they power enable people to create their own tools or design projects. Moreover, Arduino’s adherence to open-source principles allows users to simply build their own board rather than purchasing one. The remainder of Banzi’s talk is mostly devoted to providing his audience with examples of how people around the world have used Arduino to create similarly innovative tools and

---

<sup>42</sup> TED (Technology, Entertainment, Design) is a global non-profit conference series that brings provides a public forum for innovative thinkers and organizations working in these three sectors. Short presentations delivered at these conferences are made available for free online as “TED Talks,” and are immensely popular—these talks have been viewed over a billion times (source: [blog.ted.com](http://blog.ted.com)).

technology solutions that bypass many proprietary devices and other obstacles to access.<sup>43</sup>

Banzi's point is that the potential applications of this type of hardware, when combined with the creative and DIY-driven culture of the maker movement, constitute the possibility that the *tools* we use to do work (and, by extension, the *ways* we work with such tools) fundamentally alter modes of production, the makeup of labour, the distribution of capital, and the potential for innovation at both the micro and macro level. Not only does open-source hardware allow us to create versions of, or improvements to, tools that were once proprietary; it also provides opportunities for innovation. In other words, rather than relying exclusively on proprietary means to find technological solutions for a given task, why not invent and produce your own tool that can achieve the same or similar results? This gesture suggests that not only does open-source hardware stand to disrupt entrenched, industrial modes of production and consumption. It could also profoundly transform economies of labour, leisure, and play. Arduino's versatility posits it as a device for a possible cultural and social shift in which users easily construct devices that can address their given needs without total reliance on expert intervention or on proprietary goods and services. Yet technologies that provide individuals with renewed access to modes of production still function as interfaces that mediate the operator's relation to both source material and output.<sup>44</sup> In the case of desktop

---

<sup>43</sup> These solutions also include technologies that exist but are expensive or otherwise inaccessible. Banzi notes several such examples in his talk, such as simple Geiger counters distributed to people living near the Fukushima Daiichi Power Plant. These devices were designed to provide real-time updates on radiation levels via Twitter.

<sup>44</sup> Such technologies also continue to mediate the operator's relation to extended infrastructures constituted by experts, proprietary goods, and services as well.

fabrication, that mediation sparks a discussion about the nature of computationally derived handicraft.

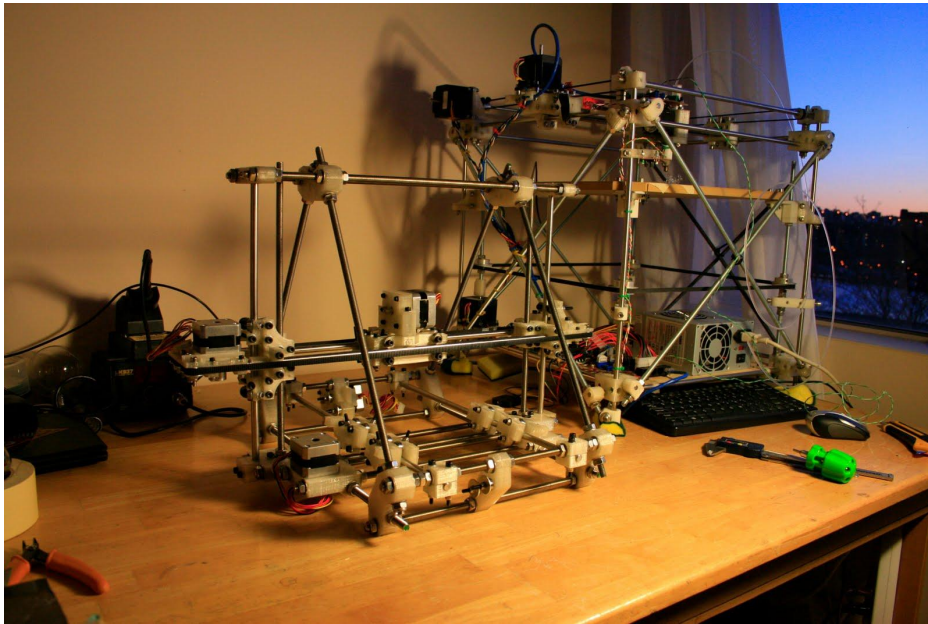
In 2005, Adrian Bowyer sparked the RepRap initiative in Bath, England. A former lecturer of Mechanical Engineering, Bowyer sought to develop the world's first open-source, "self-replicating" 3D desktop printer. The basic premise of RepRap—that a self-replicating machine comprised of inexpensive, mostly printable materials would proliferate widely, thus giving the broader public access to fabrication technology—was inspired by John von Neumann's Universal Constructor<sup>45</sup> and shaped by Bowyer's previous work in computer engineering and biomimetics at the Centre for Biomimetics and Natural Technology at Bath.<sup>46</sup> The project consists of several sets of designs for different versions of the RepRap printer, code repositories for firmware, digital models for printer parts, and a community of developers who work to refine the code, shape the machine's design, and use their printers to print parts for other printers. RepRap's firmware was written for compatibility with the Arduino Mega microcontroller; since then, the development team has developed a customized microcontroller board based on and fully compatible with Arduino's IDE.

---

<sup>45</sup> Von Neumann's model for a Universal Constructor, the logical requirements for a kinematic self-replicating machine, was first published in 1966. The machine was comprised of a chassis, tape encoded with instructions, kinematic mechanisms, a controller, and a sequencer. These components, combined with a storehouse of materials from which the machine drew its components allow for the device to build a replica of itself and also give its copy the means to replicate further. The RepRap development team writes that, according to von Neumann's definition, "RepRap is a kinematic assisted self-replicating and self-manufacturing machine" (Jones, Haufe, *et al.* 178).

<sup>46</sup> Biomimetics is the study and application of biological processes and functions in mechanical and computer engineering projects. Biomimetics covers a vast range of study—from research on artificial limbs that incorporate physical stimulus response (i.e. an artificial nervous system) to the development of nontechnology that mimics the behavior of cellular systems in nature.

After two years of research and development, Bowyer's team at Bath created the first RepRap printer in September 2006. The reprintable (RP) components were created using a StrataSys Dimension—a commercial grade 3D printer that also prints via thermoplastic extrusion. The Darwin was officially released in Spring 2007; a little over a year later, in May 2008, the first reproduced machine made with RepRap-printer RP parts was assembled by Vik Olliver, an associate team member based in New Zealand (“A History” 895). Because it was the first “child” of an existing RepRap, Bowyer and his team refer to this device as the first “true” RepRap printer, thus signaling a major milestone in the history of the initiative. In 2009, the RepRap Mendel was released: a smaller, more efficient device with an altered Cartesian mechanism. The Mendel also has a greater proportion of plastic parts over machined parts than the Darwin, thus lowering its overall cost and weight (see Figure 12).



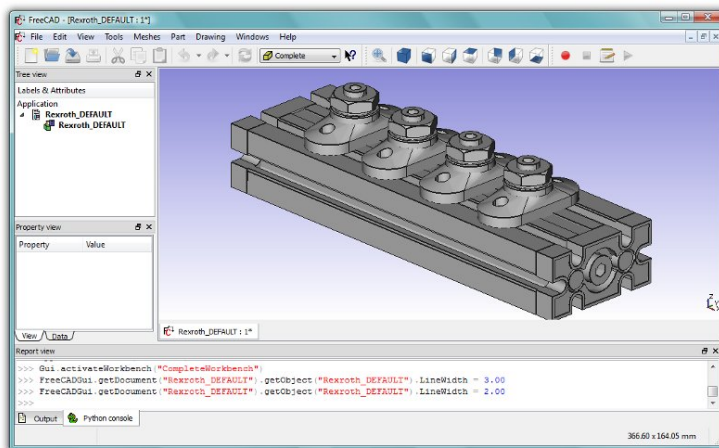
*Figure 12: A side-by-side comparison of the RepRap Mendel (left) and the Darwin (right).*



Although specifications vary, a RepRap printer consists of a thermoplastic extruder head and print bed attached to a Cartesian platform, and it networks with a computer from which it receives files for printing. Simply put, RepRap is an interface that turns virtual designs into actual objects: the device converts algorithmic signals (in the form of STL and G-code files) into printed artifacts. Motivating this process is an interface constituted by layers of transduction—of object (data from a physical object already in the world, or an operator’s rendering of a concept) to model (the virtual representation of this object rendered as an STL file in CAD software) to code (the conversion of STL to G-code in order to print instructions) to object again (the printed version of the virtual object). Part of the new trajectory in manufacturing enabled by desktop fabrication is the integration of user-friendly IDEs and the concurrent development of open-source, easy-to-use transduction programs such as Skeinforge (a computer-assisted manufacturing [CAM] program that converts virtual models into G-code) and ReplicatorG (the printer’s software, which processes G-code into print instructions). The work of the visual interface is apparent here, as user-friendliness in the context of machine-printed objects necessarily involves a degree of on-screen, WYSIWYG<sup>47</sup> design work (see Figure 13). The logic of the screen is even present in the modeling of actual, physical objects into virtual models—including the use of photogrammetry—through the process that renders the material to the level of data.

---

<sup>47</sup> WYSIWYG (pronounced “wizzywig”) is an acronym for “what you see is what you get.” It is typically used to describe visual-based GUIs.



*Figure 13: : Screenshot of FreeCad, a WYSIWYG CAD program compatible with RepRap.*

The printer's microcontroller executes the print instructions it receives by depositing heated thermoplastic filaments via its extruder onto the print bed. The extruder moves along the machine's X-Y axis to deposit melted plastic in a two-dimensional pattern; once the pattern is completed along the two axes, the extruder head is moved away slightly from the print bed, and deposits another two-dimensional layer on top of the previous one. The object is thus built up through successive layering of the plastic through a process called "fused filament fabrication" (FFF). The degree to which the extruder moves away from the print bed constitutes the resolution of the print.

Although the actual transduction processes that convert the WYSIWYG models are unknown to most operators, the open-source element of the RepRap program follows Tom Igoe's insistence on the operator's investment in process: that, when building and programming the printer, the operator is aware that transduction is taking place. Nevertheless, the process whereby a virtual model is converted into a plastic prototype becomes largely naturalized once the machine is running properly. The printer requires no further hands-on work to make a printed object; once the operator hits the print button, the interface between the modeling software and the microcontroller takes over, and the

object is printed based on the transduction of code. The operator can step away; again, no further interaction is required. Comparable to Mark Weiser's ubiquitous computing, this naturalization results in an HCI that affords operators a novel interface with digital and analog objects. Conversely, it also results in a seemingly dematerialized labour process. The complexity of such an HCI thus begs the question of how the configuration of hybrid environments, objects, machines, and operators generates a unique interface effect.<sup>48</sup>

RepRap's physical interface—how the computer relates to the physical world—involves the function of the human hand, and therefore also the process of production as a relation between source and output. This interface is conditioned by the affordances and constraints of matter and the analog environment. Because the plastic filament is molten when it is deposited on the print bed (or a prior z-axis layer), it needs to cool to a sufficiently solid state before the next layer can be deposited onto the prototype. Consequently, the maximum output speed of a FFF printer is constrained by the physical characteristics of the print materials—if the printing process speeds up, then subsequent layers are deposited on soft or even molten plastic, resulting in the warping of the object. Conversely, a mechanism (such as a fan) that provides for more rapid cooling of the deposited plastic also results in warping. Although a finer-gauge deposition theoretically lowers cooling time, the overall output remains at a somewhat consistent rate relative to a lower gauge print, as the machine is required to work harder (i.e., make more X-Y movements) to produce the same layer. These physical constraints translate into a far lower rate of output than is possible with an industrial-grade, mass-manufacturing

---

<sup>48</sup> Alexander R. Galloway writes that “interfaces themselves are effects, in that they bring about transformations in material states” (*Interface Effect* vii). Here, I use this phrase to suggest that the RepRap's interaction with digital and physical processes and materials is effectively different than that of a human producing similar output.

machine. The RepRap is therefore best suited for small-batch printing or outputting component parts. Accordingly, the device's target user base is primarily artists, designers, inventors, and hobbyists, or those whose vocations depend on the production of prototypes and customized, handcrafted, or artisanal objects. For such practitioners, RepRap codifies design and computationally produces—through the transduction afforded by microcontrollers—what was previously built by hand.

In the sense that the printer's particular characteristics make it ideally suited to the production of small batches or component parts, the printer is not ontologically dissimilar from the operator—more specifically, the printer's thermoplastic extruder produces what can be described as a digital craftwork similar to objects produced by hand. In this way, the RepRap echoes the motivations for the Arduino team to build a physical computing tool that an artist or designer could easily and efficiently make use of: an early sketch or design is transferred to a computer screen, where the 3D model is designed and rendered. Using an Arduino, the machine then transduces the model into an exact, printed rendition through a series of computational and mechanical steps. Because the model is represented as code, multiple copies can be made, each formally identical to the last; alternatively, the model can be refined or improved, and a better version can be printed. Yet, for instance, to call the RepRap a prosthetic for the hand is to ignore not only how the interface between source and output functions but also how vital the role of the haptic interface is to understanding the kind of craftwork that is produced manually by people and computationally by the RepRap.

Here, Walter Benjamin's concept of the "aura" of art is informative, and returns us to the important idea mentioned at the beginning of this essay, namely how embodied

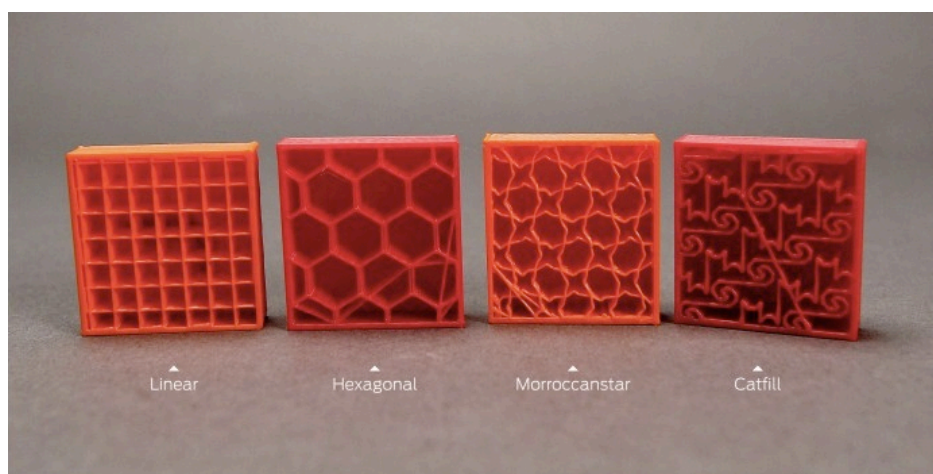
knowledge represents a resistance to the abstraction of information from materials. According to Benjamin, “the technique of reproduction detaches the reproduced object from the domain of tradition” (221). A hand-produced work is the result of an interpretive act that draws on knowledge, training, and intuition gained from the tacit knowledge of the craft gained through prior experience. The artist’s source (such as a blueprint, a photograph, or a musical score) is always to some degree ineffable to the artist, and it is this ineffability that results in the work constituting an act of interpretation. In other words, artists must make decisions as to how to represent what they do not know or understand about the source. Artists’ and philosophers’ sense of the ineffable (and their attempt to interpret it) has in previous times been called the “sublime.”<sup>49</sup> Such interpretation is at the core of what Benjamin describes as the aura of an artistic work.

Conversely, though transduction, the machine understands the source as executable code. From the machine’s perspective, an interpretation is not taking place; gaps in the data set or schema are never filled outside the machine’s procedure for filling them—the printer does not print what it does not receive as an instruction through its own operational protocol. The ineffability of the object is gone, replaced instead with information. (For RepRap, this information is the G-code instructions it processes.) The machine’s act of production is thus an operation that renders this informational totality of

---

<sup>49</sup> English poets of the Romantic era, such as William Wordsworth or Samuel Taylor Coleridge, as well as philosophers of metaphysics and aesthetics, such as Immanuel Kant and Edmund Burke, saw in the grandeur of nature a reflection of the transcendence of the ineffable essence of the human spirit, and sought to represent the feeling of briefly grasping that ineffability in written form.

the source in physical space.<sup>50</sup> What results is a structural, optimized transduction of computational process, facilitated by a microcontroller. Curiously enough, this process can be illustrated by considering the infill patterns created by a RepRap printer. To maximize speed by minimizing the volume of material used in a print, the machine prints a reinforced “web”: an infill that provides structural stability while allowing an object to be mostly hollow. The infill is typically a repeating pattern, commonly hexagonal (giving the appearance of a honeycomb as the object is printed), but one that can also be customized (see Figure 14).



*Figure 14: Customizable infill patterns for fused-filament desktop fabricators.*

Although the designer of the digital object can direct the printer to fill the object with hexagons or even repeating cat patterns, this process represents an abstraction between the virtuality of the object and the physical construction required to make the object viable in its physical form. In other words, the digital object still appears (and is) solid in

---

<sup>50</sup> While computation can be understood as machinic “interpretation,” my point here is that is such interpretation is functionally accounted for in the machine’s protocol. Granted, its schema performs an act of interpretation in that it takes code and makes something that is not code, but this act is constrained by its instructions for execution. A computer will not interpret the source in any way outside of what it is programmed to do—rather, following Benjamin, I understand the transduction of code into matter an act of reproduction.

digital space. For operators, the solidness of the object rendered in the visual space of the CAD software ends at the surface, and what lies underneath is abstracted and therefore ineffable. For example, take a “solid” printed block. For the operator, the exterior surface of the object is continuous, closed off to direct observation. The operator is therefore forced to infer that the inside of the object is a solid mass. Conversely, as it prints the block, the computer will “fill in the blanks” with the infill pattern. The machine is privy to the information regarding both the digital model and the prescribed infill pattern, and so it computationally “understands” the solidness (or lack thereof) of the model in its totality and prints the prototype accordingly.

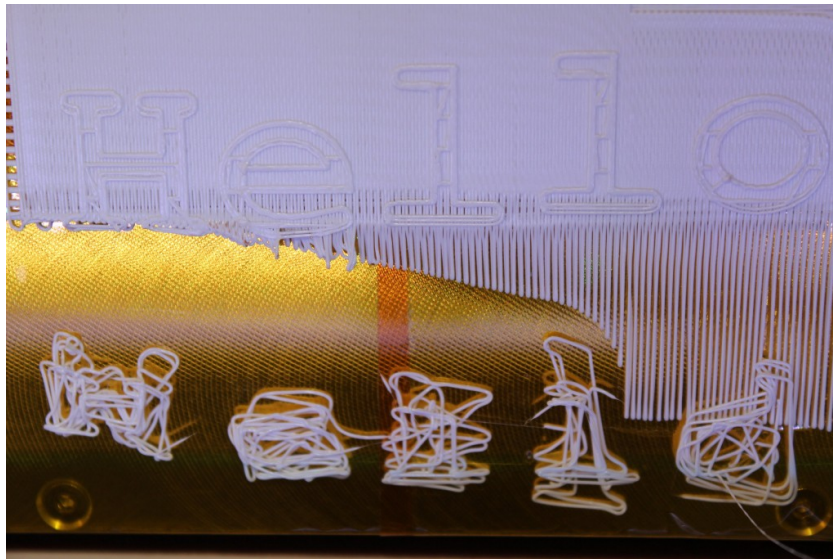
The difference in relation to the source by both the operator and the machine is one of ineffability versus immanence, respectively. Yet relation to the source is only half of the interface equation. The other half is the relation to the output (or expression), and in this case, the dynamics are reversed: the operator’s relationship with the object is one of immanence, where the machine’s is one of ineffability. As artists work with material to create objects, they are engaged in a tactile dialogue with the materials used to make the object, and thus the object itself. For example, a heavily knotted piece of wood shaped by a woodcarver effectively communicates its unique thingness to the carver, who may have to alter a planned design or a technique on the fly in order to work with and through the unique properties of that given material. The resulting piece of handiwork is thus the representation of the expressive interpretation of the ineffable coming into contact with the immanence of the material.

Whereas the hand engages in a haptic dialogue with the object, thus engendering a relation of immanence, no such process occurs when a machine prints an object. The

extruder simply follows the directions expressed in G-code and enacted via its schema. Once it has deposited the pattern for a given layer, it ceases to engage with the material, hence the need for proper calibration prior to printing: such labour ensures that objects are produced without error. Without calibration (including the debugging process), the printer is incapable of self-correcting or remedying errors (e.g., an incorrect temperature setting for a particular thermoplastic or an unlevelled print bed). And when something goes awry in the interface's expressive function (see Figure 15), what results is a misprinted object.

Yet misprints do not represent a Heideggerian "present-to-handedness" so much as a making evident (to the observer) that the machine has no hermeneutic relationship with the object it produces. It has no knowledge of the object in physical space; the object is perfectly ineffable. Of course, it is reasonable to interpret this ineffability as an engineering problem, one that future machines could remedy—in which case, autopoiesis becomes a matter of refinancing the mechanism, of integrating sensors into the machine to give it a more complete understanding of the object's material. Yet, in such a case, the immanence of the output to the machine fundamentally remains a result of computational processes, and therefore of reproduction, not interpretation. According to Benjamin, the issue here is that of authenticity, which, in the work of art "has its basis in ritual" (230). Although the computer could algorithmically interpret the contours of the object, it nonetheless remains divested of the contingencies of artist practices, interpretations, and rituals so central to creating interpretive work out of materials at hand.





*Figure 15: A misprinted object on a 3D print bed.*

Given that, in the context of manufactured objects, both people and machines work as interfaces between source and output, the interface is naturalized to the degree to which transduction between concept and execution appears fluid (Galloway, *Interface Effect* 25). In the case of people, production becomes naturalized through recursive and deeply ideological relations with materials. The object is created in a way that makes absolute the process whereby the ineffable source is interpreted through the immanence of the material used to craft the object. For the 3D printer, this naturalization occurs in reverse epistemological fashion when the perfect immanence of the virtual object (transduced as code) is rendered in actual (yet equally ineffable) space. If the interface breaks down, then, for people, it is at the interplay between source and transduction (such as when an interpretation is muddled, or the source is opaque) and, for machines, at the interplay between transduction and output (such as when a bug or error results in a misprinted object).

Despite its genealogical and functional ties to manufacturing, RepRap was not designed as a model for upending socioeconomic modes of production; rather, it was

intended to demonstrate the effectiveness of 3D fabrication in a biomimetic reproductive system (Jones, Haufe, *et al.* 177). RepRap biomimetically incorporates several “natural” systems: it is designed to self-replicate using materials in its environment and, according to Bowyer, forms a “symbiotic” relationship with its operators, who assemble the machine in exchange for the opportunity to print things they need. More significantly, its ability to self-replicate means it can ostensibly evolve and adapt to its environment:

[RepRap] is really a piece of biology. This is because it can self-replicate with the symbiotic assistance of a person. Anything that can copy itself immediately and inescapably becomes subject to Darwinian selection, but RepRap has one important difference from natural organisms: in nature, mutations are random, and only a tiny fraction are improvements; but with RepRap, every mutation is a product of the analytical thought of its users. . . . Evolution can be relied on to make very good designs emerge quickly. It will also gradually eliminate items from the list of parts that need to be externally supplied. Note also that any old not-so-good RepRap machine can still make a new machine to the latest and best design. (“Philosophy” n. pag.)

Bowyer’s claim is that RepRap enables a kind of Darwinism-by-intelligent-design—a device that substitutes a network of operators for the natural environment as the mitigating factor for its “survival.” Because of this ability to rapidly adapt to the needs of an open-source network, Bowyer suggests that the RepRap ought to, in practice, eventually dominate the desktop 3D printer market.

The biomimetic evolutionary program of the RepRap initiative suggests that the machine that suits the widest number of users will be the design that proliferates the most

widely. In 2010, programmer Josef Prusa shared his designs for a cheaper, easier-to-assemble version of the RepRap Mendel on thingiverse.com, which is a repository of open-source objects. Due to its streamlined reworking of the Mendel, the “Prusa Mendel” quickly became the most popular version of the RepRap family. Today, the RepRap community highlights the “evolution” of RepRap from the early Darwin models to the widely proliferated Prusa Mendel (and all the lesser known variants in between) as a demonstration of why their model is effective: as long as the demand exists for low-cost 3D printers, RepRap—with their formula for “fit” technologies—will invariably survive. In fact, Prusa himself points to the biological nature of the project—due to frequent updates to the software and hardware (the latter of which can simply be printed and installed), the machine functions like a living entity that grows, adapts to its environment, and thrives. As Prusa puts it, “because of its complexity, the Prusa Mendel can be a living thing” (“Prusa Mendel” n. pag.). And in the context of the haptic interface, RepRap’s biomimetics take on a meaning that overlaps the ontology of machines with the ontology of operators.

The three interfaces at work in RepRap convey ironic friction in the mediation of the material production of informational patterns. The visual interface—typically configured through a WYSIWYG design where the digital model of the object is constructed—reifies a knowing-doing split that characterizes screen essentialism as an abstract representation of process. This split is at odds with both the physical interface, which posits a machine performing the work of artisan or craft worker—and the haptic interface, which makes apparent how the screen-based, virtual models constitute an ineffable source for operators and, conversely, an entirely immanent source code for the

machine. A similar friction occurs between the physical and haptic interfaces. The haptic interface reveals the machine and operator's respective relations to source and output. The machine receives the source as a code-based model, and can therefore comprehend the totality of that model and thus has an immanent relation to the source. Output is the expelled actuation of the execution of code; consequently, the material element of the output is therefore ineffable. This ineffability is evidenced in the production of misprinted objects, which suggest a success/failure binary in terms of the machine's material expression. For the operator, the source is always incomplete, and therefore ineffable. Yet the operator's body engages in a deeply recursive, responsive relation with the raw materials used to create output—this relation can therefore be described as one of immanence. In short, the interpretive or creative act of the operator in the production of an artifact is understood in the machine as a system of transductive events that convert code into physical thing. Both are ontologically similar. However, the operator's immanence to the materiality of productive expression articulates an expressive subjectivity, one that cannot be reduced to machinic function.

### Chapter 3: DIY Drones and the Fiduciary Subject

As everted networked technologies increasingly blur the line between physical and digital environments, government surveillance has become a foremost political topic of the digital age. Perhaps no device illustrates the political implications of networked surveillance technologies more than the unmanned aerial vehicle (UAV), or drone. Developed and deployed by militaries and commercial enterprises, drones utilize geolocation, self-correcting flight mechanisms, and algorithmically derived flight patterns in order to map landscapes and monitor the locations of people. Critical discourses on drones tend to focus on their sociopolitical and juridical impacts on contemporary society and civilization, specifically within the context of military and government use. These discussions are framed as ethical and legal issues, such as those raised by concerns over the US government's use of armed UAVs to bomb people overseas, an activity that has been interpreted as eroding international law and sovereignty, or the deployment of drones along US borders and, increasingly, over domestic airspace to track illegal immigration or search for fugitives.<sup>51</sup> Military and government use of drones also more broadly suggests how they redefine a military epistemology of warfare: as visual artist Jordan Crandall puts it, “the redistribution of manpower in the ‘unmanning’ of unmanned

---

<sup>51</sup> A notable case of domestic armed drone deployment occurred in February 2013 over the wooded San Bernardino mountains, when authorities searched for Christopher Dorner, an ex-LAPD officer who shot and killed several people, including a police officer. Before fleeing to the mountains, Dorner had posted “manifesto” on his Facebook page declaring that he would commit further acts of violence against LAPD officers and their families. The use of drones equipped with thermal imaging technology to search for Dorner was disclosed by US Customs and Border Patrol officials, who consequently became involved in the manhunt (Parker n. pag.).

combat machines” indicates that the “past ideals of heroic masculinity” are being replaced with ideals “embroiled in new forms of agility, knowledge, and prowess display” (n. pag.). Yet as drones are increasingly deployed, not just on domestic borders, but in domestic airspace, the nature of the surveillance of bodies has become an issue of governmentality.<sup>52</sup>

Primarily, drone technology functions as an instrument of governmentality by mediating the human body as locatable media. Sandy Stone describes how locative devices represent a biopolitical apparatus that extends governmental jurisdiction over individual bodies as its territorial sovereignty erodes in both virtual and real spaces. She writes that “the purpose of location technology is to halt or reverse the gradual and pervasive disappearance of the socially and legally constituted individual in a society” (39). Writing in the context of virtual reality, Stone reminds us that “this slippage, of course, does not refer only to the physical or geographic, but to the other, non-Cartesian modes of location” (39)—in order to frame the conditions for the emergence of what she calls the “fiduciary subject,” an individual constituted by the “tie between what society defines as a single physical body and a single awareness of self” (40).<sup>53</sup> Stone uses the term “warranting”<sup>54</sup> to refer to this articulation, which, as she describes, is “the

---

<sup>52</sup> Michel Foucault use the term “governmentality,” or “the art of governing,” to describe how the technologies of power are underpinned by political rationality. More specifically, governmentality refers to the maintenance of power through the apparatuses and techniques employed to manage populations, sovereignty, and economics.

<sup>53</sup> “Fiduciary” is a legal term that denotes a legal relationship between two parties. Stone utilizes this term to articulate the way subjectivation is constituted by the point of relation between discourse and embodiment (Stone 40).

<sup>54</sup> Stone’s concept of warranting is inspired by the work of William Gibson, specifically in *Neuromancer* (1984). According to Stone, “William Gibson was the first writer to deal with warranting. Although he did not specifically refer to the phenomenon, it is implicit in his understanding of the ontology of cyberspace . . . that there is an implied link

production and maintenance of this link between a discursive space and a physical space” (40). In other words, warranting functions as a mode of articulation and consolidation—it makes the discursive and physical links required for the constitution of subjectivity.

While their ability to locate bodies and share data across networks makes them ideally suited as apparatuses of governmentality, drones’ technologies have become increasingly inexpensive and accessible to private citizens. As such, drone production and usage has become a popular activity among physical computing practitioners. In recent years, amateurs, academics, and hobbyists have worked within the open-source community, and with technologies such as Arduino, to develop UAV technology beyond the scope and control of the military and commercial paradigm. Open-source UAV technology offers a new context for understanding drones, a narrative in which the operator technologically augments his or her ability to navigate and monitor physical space. Accordingly, drones become complex sociopolitical artifacts—the “eye in the sky” metaphor generally applied to drones stems from the technology’s alignment with a libertarian political agenda that resists state territorial hegemony.<sup>55</sup> Moreover, the eye in the sky metaphor also reifies drones as mechanisms for objective, totalizing vision; as Donna Haraway writes, such mechanisms “signify a perverse capacity—honed to perfection in the history of science tied to militarism, capitalism, colonialism, and male

---

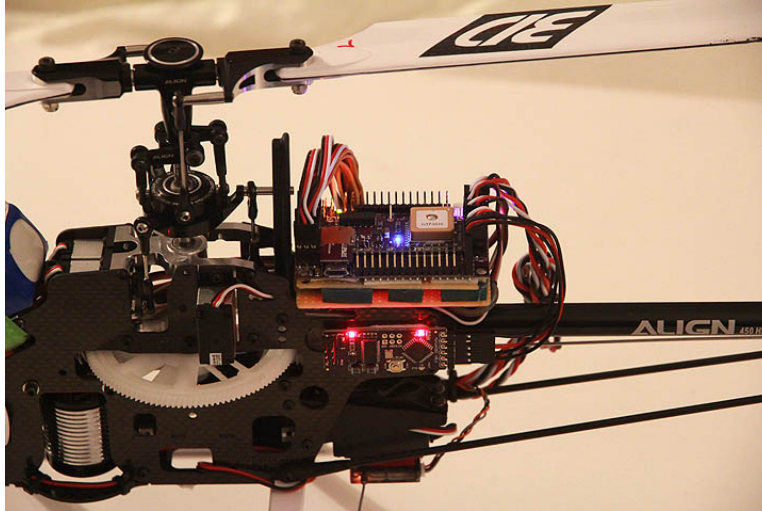
between the virtual body . . . and the convergences of discourses . . . that constitute the body in physical space” (41).

<sup>55</sup> Also, following William Gibson’s assertion that “the Internet has everted. Turned itself inside out,” drones, as physical manifestations of the eversion of cyberspace, also evert the libertarian political logic of John Perry Barlow’s 1996 “A Declaration of the Independence of Cyberspace,” a statement against the extension of governmental regulation and territorialization of the Internet. In other words, drones allow for the cyber-libertarian agenda to carry over into physical space—the issue becomes one of asserting the sovereignty of one’s own body and physical reach over that of any governmental regulatory power.

supremacy—to distance the knowing subject from everybody and everything in the interests of unfettered power” (“Situated Knowledges” 581). Yet, as explored in the discussion of RepRap in the previous chapter, the notion of drones-as-prosthetics fails to account for the machine’s autonomy in flight and ability to retrieve information. In essence, what is conceptualized as a prosthetic is actually a device that has been afforded decision-making abilities on the part of the operator. Furthermore, a purely political evaluation of open-source drones elides the topic of the HCI between the machine and the operator, specifically where it relates to their respective modes of transduction between environment and information. In the case of the drone, what appears at first to be an operator’s mediation of data obtained from a physical environment (this is the prosthetic interpretation) is in fact the drone’s semi-autonomous rendering of a physical environment into storable and transmittable information.

The ArduPilot Mega (APM) is described as “the world’s first universal autopilot”—an open-source Arduino-based autopilot that works to turn any remote-controlled (RC) vehicle into an unmanned vehicle (UV) or unmanned aerial vehicle (UAV). The APM platform consists of a circuit board that uses the Atmega328 microcontroller (the same used in the Arduino UNO, currently the most widely used Arduino board) programmed with algorithm firmware, the autopilot software that is uploaded via the Arduino IDE, and optional headers to connect a global positioning system (GPS). By connecting the RC receiver to the APM circuit board (see figure 16), the operator can program the vehicle to perform autonomous flight while still retaining the ability to manually override the autopilot and control the aircraft remotely.





*Figure 16: The ArduCopter, a helicopter-compatible version of the ArduPilot Mega (APM), installed on an RC helicopter.*

The APM was developed and marketed by 3D Robotics, a company started in 2009 by Chris Anderson, the former editor-in-chief of *Wired* magazine,<sup>56</sup> and Jordi Muñoz, a 23-year-old amateur robotics enthusiast from Ensenada, Mexico. Anderson discovered Muñoz<sup>57</sup> via the latter’s involvement with the message boards at DIYDrones.com, a social media site created by Anderson in order to bring together amateur UAV enthusiasts. Apparently impressed by Muñoz’s innovative approach to programming autonomous flight robots and the respect he garnered on the messages

---

<sup>56</sup> This is the same Chris Anderson discussed in chapter one who wrote *Makers: The New Industrial Revolution* and who built a networked, open-source version of his grandfather’s automatic sprinkler. He also appears in a promotional video for Project Tango, claiming that the device “solves the problem” of “mapping indoors” (“Project Tango” n. pag.). In other words, Anderson has situated himself as one of the most prominent voices of the maker movement.

<sup>57</sup> Anderson’s decision to start a robotics company with Muñoz is a manifestation of Anderson’s own business philosophy called the “Long Tail,” detailed in his trilogy of books (*The Long Tail*, *Free*, and *Makers: The New Industrial Revolution*)—in short, he believes that the vast distribution of resources and information afforded by the Internet will result in a larger percentage of future innovation coming from smaller online communities than from large firms.

boards, Anderson recruited Muñoz for his new start-up (called 3D Robotics) to help Anderson develop its flagship product, the original ArduPilot. Partially due to Anderson's heavy involvement in both projects, DIY Drones and 3D Robotics form a recursive relationship—3D Robotics sells the hardware and software that fuels community praxis at DIY Drones, and DIY Drones operators work to continually improve the technology that 3D Robotics provides.

For example, early releases of the APM platform relied on infrared sensors for craft stabilization. While such technology allowed small fixed-wing RC planes without ailerons<sup>58</sup> to achieve somewhat stable flight, it was not as effective for more complex behaviours such as return-to-landing (RTL) or aerobatic maneuvering; nor did it allow stability with such maneuvers in rotary wing aircraft (such as helicopters or quadcopters). William Premerlani and Paul Bizard, two members of the DIY Drones community, developed an algorithm that they called the “direction-cosine matrix” (DCM), which allows for better stabilization for fixed- and rotary wing aircraft that can perform more advanced aerobatic maneuvering (such as planes with ailerons, or, in other words, planes that can bank sharply). Premerlani and Bizard, in a paper posted to the DIY Drones message boards, lay out their DCM theory and describe a piece of firmware they wrote and built into an IMU which allowed their RC planes with ailerons to fly and maneuver with far greater stability. The firmware was adapted by Muñoz for use with the original Ardupilot board, dubbed the “ArduIMU+,” and licensed for retail sale by 3D Robotics. While the ArduIMU+ has since been replaced on the APM by a closed-source IMU (developed by a multinational corporation called InvenSense), it can still be purchased

---

<sup>58</sup> Ailerons are the rudders on a fixed-wing aircraft's wings—they allow the plane to bank.

and integrated into the Ardupilot system. The ArduIMU+ provides an effective example of how a technology and the community of practice that has grown around it come to form a recursive relationship—a cybernetic model of open-source technology is spurred and reified by the community who defines itself in relation to the same technology. While the Arduino-based products developed by the 3D Robotics/DIY Drones partnership are not the only open-source autopilot projects available, they have become successful due in large part to their adoption of the same principles that have made Arduino itself successful—low-cost hardware, accessible programming requirements, and a large user community.<sup>59</sup>

The “Follow Me” mode developed for use with the Arducopter (the APM connected to a helicopter or multi-rotor copter) is one example of a particular application of open-source personal UAV technology. Follow Me allows operators to link a UAV to a GPS unit in an Android phone and have the drone follow the GPS signal as they carry or rotate the phone. The function is described on the DIY Drones message boards as turning a UAV into a “personal droid.” Anderson points out in another blog post that the Follow Me mode is DIY Drones’s answer to the technology used by the “Joggebot,” a quadcopter UAV designed by the Exertion Games Lab at RMIT in Melbourne to follow a jogger and act as a sort of fitness trainer/motivator. The Exertion Games website states that “with Joggebot, we want to propose the idea of robots as companions for physical activity. We believe this is a promising approach, as both robots and exercise are embodied, by which we mean they are both heavily body-focused” (n. pag.). Of course,

---

<sup>59</sup> As of this writing, the original ArduPilot board is available on Sparkfun’s online store for \$24.95 USD. By comparison, the most widely used open-source autopilot prior to ArduPilot was the Paparazzi project, which requires components with costs upwards of seven hundred dollars.

besides its role as a fitness trainer, Joggobot could also feasibly function as a personal security detail, providing a means of alerting authorities if the runner is, say, attacked or injured; alternately, the device could function as a deterrence mechanism against would-be assailants or dangerous animals. Returning to the libertarian politics of Anderson's maker subject in chapter one, both Joggobot and Follow Me provide individuals with the means to secure their own locations and security, thus generating a kind of self-built independence from larger social framework previously relied upon to secure one's well-being and safety.

Joggobot utilizes a "built-in camera and tag detection software" to maintain a lock on the jogger's movements—the "tag" consists of a specific bright orange and green pattern on the front of the runner's t-shirt. The Exertion Labs website contains videos of runners with the distinctive t-shirts running around an empty athletic track or empty sidewalk, the quadcopter leading the way and intuitively changing directions when the runner turns to maintain orientation in front of the person's body. While the notion of a drone reading a runner's t-shirt for "training" purposes is a novel application of UAV technology, the limitations of this way of reading the body become evident when compared to the way Follow Me mode uses telemetry and geolocation, rather than camera and tag detection. With the open-source Follow Me, the drone can be programmed to track a target in multiple orientations—behind, beside, above, or in front of the body, or wherever it needs to be to avoid other obstacles—while the proprietary<sup>60</sup>

---

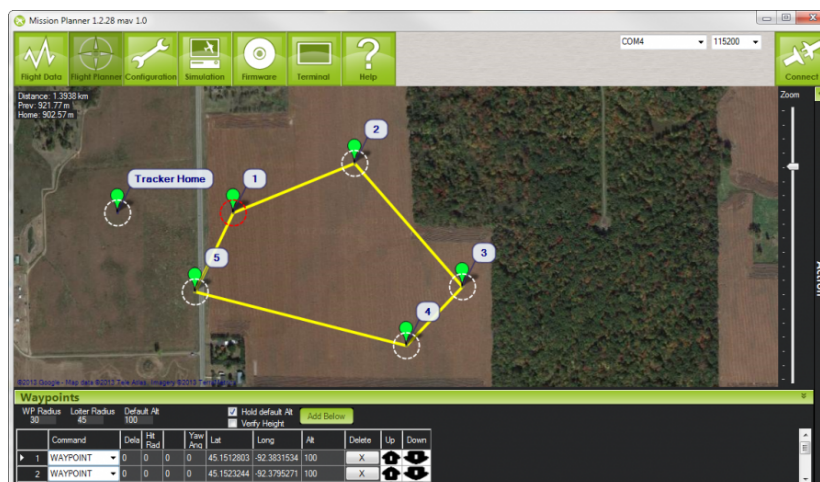
<sup>60</sup> Follow Me speaks as much to the proprietary model's constraints as it does to Arduino's affordances: while Joggobot's tag-identification system was designed for the specific purpose of reading a t-shirt, it is a custom tool, both (currently) non-compatible with the APM and prohibitively expensive for many APM users. The telemetric communication with a GPS unit is a pragmatic method of achieving largely the same

Jogobot is constrained to maintain its position relative to the front of the jogger, so that it can read the tag on the jogger's t-shirt. As well, any obstructions, such as trees, busy sidewalks, or low light levels, might cause the Jogobot to lose track of its target, while Follow Me is freed from such constraints of visual tracking.

Like RepRap and Arduino, ArduPilot is designed for non-programmers to equip their RC vehicles with UAV technology: the DIY Drones community has developed visual, user-friendly interfaces for non-programmers who simply wish to own UAVs and not work on development. To this effect, several WYSIWYG interfaces for programming flight routes exist. Depending on the operator's level of expertise or interest, route plotting can be as simple as dragging points across a 2D or 3D landscape representation on-screen or as involved as hand-coding coordinates and flight instructions (see figure 17). Yet, at least in the case of larger-scale missions, the operator's range of access to controlling the device stops short of controlling flight itself—ArduPilot's autopilot technology affords the craft with the means to self-correct in its environment (against wind, for example, or to maintain altitude over uneven terrain).

---

effect, using a more inexpensive toolkit and an open community of developers, and, ironically, results in a more robust performance—freed from some of the limitations of the software in the Jogobot's subject-detection technology. Because the Arduino allows for easier integration with different sensors, it functions as a tool that gives the designer a wider range of options for designing other tools (in this case, a UAV) that suit a given purpose.

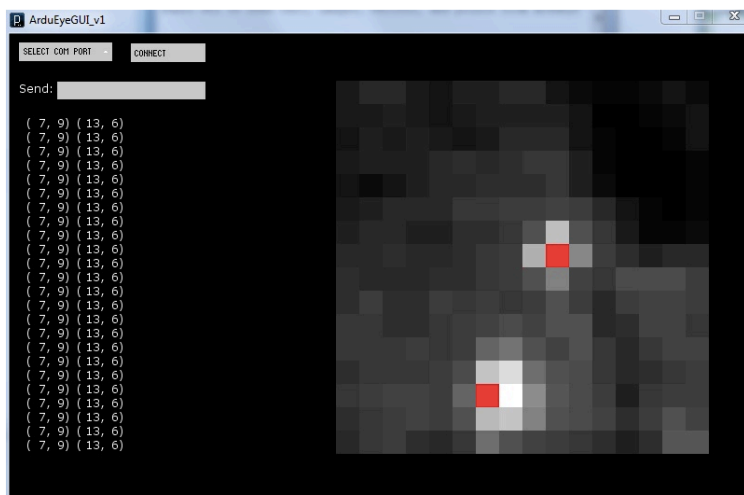


*Figure 17: A Screenshot of Mission Planner, a WYSIWYG software interface for programming flight routes on an ArduCopter craft.*

ArduPilot’s physical interface consists of the way it relates to the physical environment—both through its means of self-correcting, autonomous flight and to the various other sensors and instruments integrated into the craft used to capture data from the environment. There are numerous discussions on the DIY Drones message boards concerning the types of “payload”—instruments that can record field data—that can be attached to the crafts. Commonly, payloads consist of recording instruments, such as digital cameras, thermal imaging sensors, and others that can deliver a cache of data upon return-to-landing (RTL) or transmit that data to the operator mid-flight.<sup>61</sup> One such payload is the ArduEye, a small camera capable of detecting light sources. ArduEye senses the location and intensity of light sources on a sixteen-by-sixteen pixel grid (up to ten different sources can be sensed) and outputs the data to a serial monitor and visual representation (see figure 18). The location of the source can be refined to a tenth of a pixel. Multiple light sources display in the serial monitor as discrete sets of coordinates.

<sup>61</sup> Any range of sensors can be attached to an ArduPilot craft—for example, temperature or humidity sensors are used in craft dedicated to meteorological research.

ArduEye can also be used as an autopilot mechanism, able to correct yaw and height according to light sources, and can thereby augment or replace a craft's gyroscope.



*Figure 18: The serial monitor readout and visual representation of discrete light sources detected by ArduEye (source: DIY Drones).*

The APM drone's haptic interface consists of the way its physical interface—the mechanisms that allow it to move, self-correct, and otherwise perceive and navigate space—is analogous to the way bodies navigate in physical space. That is, the APM drone can perceive its surrounding environment through visual means, but also through means of touch, balance, and so on. In doing so, this interface sheds light on the way that physical environments are processed as information by both bodies and computers. The device achieves this ability through a complex system of sensors and controllers that enable a craft to pilot itself autonomously. The APM predominantly utilizes an inertial motion unit (IMU), which uses a combination of gyroscope, accelerometer, and magnetometer to provide for stable flight and maneuvers relative to the earth. Equipping the APM with a GPS enables the vehicle to geolocate, and, in conjunction with the

APM's memory,<sup>62</sup> as well as further sensors, provides options for flight logging and for complex flight routes to be programmed into the vehicle, sent remotely from the operator, or even for allowing the craft to fly stably in relation to an external target.

The limitations of the operator's control over a device that uses sensors to fly itself provides an important challenge to the cultural understanding of the personal drone as a kind of prosthetic eye. Unlike a prosthetic, a drone is capable of making its own decisions, and must constantly do so to maintain autonomous flight. Operators can understand that processes related to such autonomy occur, and, in many cases, can comprehend how the process works. They can also interpret output, such as the readout of a serial monitor, or a graphic representation. But operators are estranged from the device's computational reconstruction of the environment—in other words, operators simply cannot view the world or objects the way a computer does—as data, not as a communicational conduit to human cognition. If anything, the drone functions, like RepRap, according to the logics of biomimetics—the conditions for its operation are self-contained; it navigates space and gathers data without direct human intervention. Once the autopilot takes control, the drone is a self-functioning entity. Its computational matrix—gyroscope, GPS, and other sensors and communications networks—is capable of making autonomous decisions, albeit decisions algorithmically mitigated by a program constructed by a human operator and ultimately interpreted by people. It exists in a far more complex interface with the operator than as a simple prosthetic; it is less a tool than an autonomous agent—an actor with a subjectivity of its own, which thereby interpellates

---

<sup>62</sup> As of this writing, the most current model of the APM is the APM 2.6, which provides 4 megabytes of flash memory and options to upgrade.



the people it locates as others, bundles of data to be recorded, processed, and shared across networks.

Yet again, ironic paradoxes between interfaces emerge: the physical interface of an ArduPilot drone works to enable an operator to expand his or her realm of knowledge and influence while paradoxically excluding the operator from the computational decision-making process the drone itself is required to make to sustain operation and gather data in the physical environment itself; moreover, much like a map, quadrant, or globe, this interface ironically reduces the operator's physical proximity to the very bodies and things the drone senses; yet it transmits dynamic data on these things that exceeds the representational format of a map—this data is contingent on the ever-changing material conditions of the environment. In the context of the physical interface and its effects on the machine's comprehension of the physical world, Stone's description of the means of warranting—the apprehension of the fiduciary subject by political or social apparatuses—is prudent. Stone originally described warranting in the context of political and social apparatuses for location—warranting was a means of sovereign response to the erosion of boundaries wrought by the migration of social spaces to the virtual domain. With the eversion of cyberspace, location technologies are turned outwards from virtual to physical spaces. Paradoxically, the embodied part of Stone's fiduciary subject becomes reworked as informational—technologies such as drones read bodies as sources of data.

The fiduciary subject is simultaneously constituted by various political and social discursivities and a physical, locatable body. This subject is both a “self” in a network of social and political relations, and a “being in place” (Stone 90). The drone, too, can be

understood in the context of subjectivity: it is functionally autonomous, capable of making decisions, and also consists of a locatable, physically embodied form. It is discursively constituted as well—in the case of personal drones that are used for various purposes related to the motives of the operator, political and social discourses are supplemented with algorithmic schemas for operation. Its schema articulates the drone as a location device—it locates itself within a physical environment, and locates bodies and other entities, while political and social discursivities construct it as an apparatus for governmentality or for personal liberation from the constraints of grounded, embodied space. For the drone, the locatable human body is reducible to data that it collects and shares according to its schema for operation (drones are, as mentioned above, primarily used for surveillance and reconnaissance purposes, which involve the sharing of collected field data). In so doing, the drone, according to Stone's definition, functions as a mode of warranting: it locates the embodied individual and understands (or interprets) him or her as an entity constituted by discursivity—thus facilitating the "production and maintenance of this link between discursive space and physical space" (40). And if the drone can be defined as a subject in its own right, its warranting of a human subject can be read as a social relation that consists of human and non-human entities. In this context, the haptic interface interrogates the clear line between the subjectivity of the operator and the subjectivity of the machine. Read this way, personal UAVs, as an everted technology, render HCI as a *social* relation.

While Stone's understanding of warranting is as a means by which political or juridical apparatuses articulate subjects, in the context of personal UAVs, warranting functions as a means of a technocultural apparatus that relate to the management of

private interests—when drones can be controlled by individuals, the subjects that drones warrant take on a more fluid discursive dimension. Put another way, if drones function as a mode of warranting, and they can be controlled outside of the domain of governmentality (i.e., they are not used exclusively as apparatuses for population management or the securitization of sovereign territory), then the fiduciary subjects they articulate are constituted by both physical embodiment and discursivities not strictly confined to the kinds of subjugated formations typically associated with the effects of power on subjectivation. For example, a body detected by a UAV's thermal imaging system illegally crossing the Mexican border into the United States is warranted as an illegal alien, a smuggler, or possibly even a threat to national security; in short, an individual encroaching on the sovereign territory of a nation and thereby necessitating a response from the state's security apparatus. This process of warranting articulates a fiduciary subject that is sufficiently real to elicit some kind of response by the state. But what about a body detected by a privately controlled drone to be walking across someone's private property? What sort of fiduciary subject is warranted, and what apparatus is articulating the discursivities that help subjectivize that body? And, more importantly, what modes of security (or otherwise) are deployed to deal with that subject, and under what right, and to what detriment to the other ways that two subjects might recognize one another? This line of inquiry can be reversed as well; take, as another example, the questions that arise when the articulation of fiduciary subjects occurs on their own property, or inside their own homes, by drones engaged in trespassing or spying. What kinds of discursivities are folded into the warranting of a subject here? Although the specific answers to these speculations exceed the scope of this paper, they

all point to a way in which social relations between people change in the eversion. The hypothetical issues raised here are intended to illustrate how certain modes of access to technology afforded by physical computing practices—in this case, the ability of individuals to build, program, and fly their own drones—sparks a discussion on the ways that human relationships can be mitigated by modes of warranting performed by everted technologies. What results is a potentially new and possibly dangerous mode of social relationality, in which individuals interpellate each other not as physical bodies in proximity to one another, but as discursive, informational patterns to be assessed, recorded, and, to some degree, ultimately abstracted, ironically distanced from one another despite their physical locatability.

### **Conclusion: HCI as Breached Boundary and the Presence of the Subject**

The visual, physical, and haptic interfaces all underscore the different elements at play within the HCI, elements that work in and through everted technologies to articulate subjectivity. The visual interface ironically works to obscure transductive processes taking place in order to provide a mode of access to the operator, a function that can be understood in the context of WYSIWYG interfaces designed to allow non-programmers to, say, plot a flight route for their personal UAVs, or copy and paste a simple sketch into an IDE to get their Arduino board to do something. Physical computing technologies provide means of operator access beyond the screen; their indications of transduction, algorithmic function, and data processing are expressed through near-limitless varieties of actuation. In this configuration, operators become epistemologically attentive to process—they are made aware that the devices they interact with function according to layers of transduction taking place at the physical and computational levels, even if they don't entirely understand the how or why of such transduction.

In some ways mirroring the visual interface, the physical interface works to provide computers with a means of accessing physical bodies, environments, and processes. In the case of Project Tango, for example, various visual processors combine to form an algorithmically constituted three-dimensional understanding of the space around the device. Again, this interface is epistemological—the machine ascertains the physical as legible data that it can process and cybernetically respond to. In short, the visual and the physical interface are both primarily epistemological, because both illustrate how everted technologies and physical entities function in networks via modes

of recursivity—the physical world is made apprehensible through the mediation of information, and information is in turn made apprehensible in the physical realm through actuation or expression. Here, recursivity can be understood as mediation between physical and informational states.

Things get interesting, however, with the articulation of the haptic interface that emerges in physical computing and everted technologies. Unlike the visual and physical interfaces, the haptic interface is ontological because it articulates the blurring of the boundary between the particular embodiments of the operator and the machine, thereby complicating the posthuman notion of the “distinct difference” between the two. Take, for example, the “expression” of machinic actuation mentioned above. Returning to the case in chapter two of the expression of the RepRap via the printing of a plastic artifact versus the expression of the operator producing a work of handicraft, the haptic interface articulates the ontologies of both by revealing their respective relations to source and output as being constituted by ineffability or immanence. In this example, the machine’s relation to output is understood as a result of algorithmic function, and therefore its “expression” is ineffable to it, while the operator’s expression of handicraft speaks to a certain recursive relation to the physical object as it is produced. While the machine is capable of producing expression, it is ontologically aligned with the source, while the operator is ontologically aligned with expressivity; accordingly, the operator can be articulated as an expressive subject. That the ontologies of operator and machine get articulated through the haptic interface is important because it allows us to examine the ways in which the embodiments of machines and operators overlap, align with one another, or distinguish themselves as different. Here, the study of a boundary object such

as Arduino and the HCI that it articulates challenges the dualistic model of recursivity of both posthumanist and OOO notions of human and non-human forms of embodiment. In so doing, it provides the opportunity for conversations about subjectivity in the context of humans and non-humans to move beyond somewhat stifling dualistic categories like human/non-human, cognition/non-cognition, and vitality/withdrawal, and instead talk about subjectivity as a particular configuration of discursive or informational elements with a physical form.

If distinct difference between human and non-human embodiments and modes of information is an insufficient model for articulating subjectivity in the current technocultural moment, then what emerges as a suitable way of understanding HCI and the subjectivities that get articulated by it? The resulting technocultural framework I've endeavoured to formulate through the concept of the three interfaces aligns with feminist theories of how subjectivation happens within and through the enigmatic construct of boundaries. Sandy Stone's concept of warranting as the mode of locating and mediating the fiduciary subject (discussed in the context of personal drones in chapter three) posits the subject as just such a constellation of information and embodiment—a subject that is articulated as “a link, a coupling between the phantasmatic space that the location technology calls into being and the physical space of pain and pleasure that the human body inhabits” (40). For the fiduciary subject, the boundary between information and embodiment is actually the condition for its articulation.

The boundary between embodiments can also be understood as a space of ambiguity, hybridity, and code-switching that functions to erode differences across bodies and things. Gloria Anzaldúa's figure of the *mestiza*, who emerges from the

borderlands to exist across social and political categories, is analogous in some ways to Arduino's characterization as a boundary object—both are able to code-switch, communicate across different registers, yet remain ironically consistent across categories. Though this comparison perhaps insufficiently addresses modes of subjugation that Anzaldúa points out, it nonetheless underscores the idea that boundary objects such as Arduino effectively function as transduction points or modes of mediation among objects and subjectivities. Moreover, it foregrounds the ontology of the interface, not as a border that divides different embodiments, but as a space of hybrid functions between them.

Donna Haraway's eponymous figure in the "Cyborg Manifesto" is the paradigmatic representation of such hybridity. Haraway's cyborg emerges as a result of "breached boundaries" between humanist categories such as human-animal, organic-machine, and physical–non-physical. Haraway's description of the breached boundary between human and technology reads as a premonition of the eversion and its consequences on the articulation of subjects:

Late twentieth-century machines have made thoroughly ambiguous the difference between natural and artificial, mind and body, self-developing and externally designed, and many other distinctions that used to apply to organisms and machines. Our machines are disturbingly lively, and we ourselves frighteningly inert. ("Cyborg Manifesto" 152).

While Haraway's reference to the increasingly ambiguous distinction between such categories as mind and body was written in the context of the virtualization of cognitive activity in cyberspace, or the migration of the mind from "meatspace," the eversion allows for this ambiguity to drift the other way, thereby signalling that the breached



boundary between humans and machines is physical as well as cognitive. Such a breached ontological boundary can be contextualized through the ArduPilot drone: its partial autonomy, physical locatability, and ability to make decisions posit it as a kind of subject *and* a mode of warranting that locates human subjects as well. Breached boundaries also invariably open a space through which bodies become subjectivated in relation to power even as they resist the binaries that populate either side of a boundary line.

Judith Butler's theories of subjection in *The Psychic Life of Power* provide a model in which such recursivity can persist even when the fixity (or very existence) of boundaries is called into question, thus allowing us to think through the cybernetic function between entities without reducing that function to a mediation across boundaries. She writes that the "process of internalization *fabricates the distinction between interior and exterior*" (19, original emphasis)—in other words, subjectivation is, for Butler, where the boundaries get "fabricated"; that is, the boundary that interpellates the internal/external binary is artifice. What is important here is the process of mediation, or of transduction, does not need to involve a reification of difference. For Butler, that reification is a function of power.

### **The Politics of a Present Subject**

The technocultural framework of the three interfaces—analyzed through the HCI constituted with and through Arduino, RepRap, and ArduPilot—articulates forms of subjectivity among operators and machines that emphasize the permeability of the boundaries between both machines and operators and also between physical matter and

information. In other words, HCI in the eversion renders information as material, and matter as informational, and in so doing, reconstitutes subjectivity as a function of both. The materiality of the interface speaks to its reification of presentness. What results from this configuration is a subjectivity that is bound up in the politics of presence. This presence can be understood as spatial and temporal. Temporally, this subject functions according to what Wendy Chun describes as the “enduring ephemeral.” Chun argues that in studies of cyberspace, “digital is a conflation of memory and storage” that undermines “digital media’s archival promise” (148)—that is, what is understood as the speed or fleeting ephemerality of the digital by such theorists as McKenzie Wark or Paul Virilio critically ignores the materiality of media and the link between material and information. Following Chun, I argue that the subjectivity that emerges from the HCI of the eversion provides a framework through which the “bearings of the individual” may be recuperated (151). On the other hand, the spatiality of the subject can be understood through Stone’s warranting—a subject that is produced and maintained in part by its physical locatability.

The spatiality of the subject also speaks the epistemology of what Donna Haraway and others refer to as “situated knowledge”—a critical undoing of the privileging of vision, and its attendant masculinist politics of detachment from the physical world. Haraway argues “for [a] politics and epistemologies of location, positioning, and situating, where partiality and not universality is the condition of being heard to make rational knowledge claims,” and “for the view from a body, always a complex, contradictory, structuring, and structured body, versus the view from above, from nowhere, from simplicity” (“Situated Knowledges” 589). Politically, Haraway argues, knowledge derived from situated physicality and experience undermines the

oppressive historical violence of the humanistic, vision-centric, detached, scientific mode of knowledge. Indeed, the cyberspace-as-elsewhere construct certainly speaks towards such detachment, and studies such as Virilio's "information bomb" or Jean Baudrillard's "ecstasy of communication" recognize and emphasize the violence of disembodied experience. With the eversion of cyberspace into the physical world, I recognize that such violent detachment persists—it is clear that banal everted technologies such as GPS-enabled smart phones function as juridical and political apparatuses, able to locate, exploit, and do violence upon bodies. Stone historicizes the warranting of the subject through location technologies, tracing this function back to the juridical apparatuses that constituted a physical citizen. These pre-digital apparatuses included such devices as "the census, the introduction of street addresses, passports, telephone numbers," and other "documentations of citizenship" that were instituted as legal means to "produce a more 'stable,' manageable citizen" (90). Now, in the context of everted technologies, such documents can be both virtualized and integrated into the physical body of the subject. An example of this can be seen in various identity-verification devices, such as iris-scanners or fingerprint scanners, used at international border crossings for background checks and also increasingly by US security forces to verify identities of allied individuals (see figure 19).



*Figure 19: An Afghan man has his iris scanned by a US Army officer to determine his history of relations with security forces in Afghanistan.*

With physical computing technologies and practices providing a means for the open-sourcing of everted technologies, these juridical and political apparatuses have the potential to be appropriated to increasing degrees by private interests—private corporations or groups as much as individuals. This potential towards privatized power can again be understood in the context of personal UAVs, which allow individuals to pursue a libertarian cultural politics that through technologies that allow them the means to locate, monitor, and possibly even regulate the actions of other individuals.

That said, I follow Haraway’s insistence that situated knowledge offers a means of resistance to objective, detached knowledge—as Arthur Kroker writes, “Haraway privileges intermediation ... not as utopian imaginaries but as ways of deepening her epic story of domination” (15). I argue that the *present subject*, one articulated through the three interfaces, affords an affirmative politics in which spatiotemporal presence reifies a subjective mode of experience that escapes *some*—not all—of the logics of an informatics of domination. Despite, or perhaps because of, the manner in which machines

can work to warrant the subject, and extract information from human bodies in ways that often surpass the subject's ability to "know oneself," the three subjectivities articulated in this paper—the productive subject, the expressive subject, and the fiduciary subject—remained ontologically aligned with the processes of the physical world. Their bodies exist, from the perspective of the machine, in real time, and in real space, albeit time and space that is algorithmically processed and informationally expressed in often ephemeral networks. The machine that senses the physical environment is always operating in the present moment, and, as such, the subjects articulated as in proximity to physical and temporal presence. Conversely, the physical computing practices taken up by the operator to varying degrees also roots the subject in the present time and space. This alignment can be observed via the material practices of physical computing: here, Heidegger's *Vorhandenheit*, the presence-at-hand of the broken tool, is less the interruption or rupture of a naturalized experience of being and more the starting point through which an embodied relation to knowledge and experience takes shape. Physical computing begins with attention to the tool, to the object, and from there creates the conditions of possibility for reconstructed subjective experience of the world articulated in part by the interaction, blending, and hybridized interface between people and machines, bodies and things, matter and information. While the physically present subject cannot entirely escape the logics of power as manifested in, say, locative apparatuses, it nonetheless illustrates the opportunity for a mode of experience in which the material world once again provides the basis for experience.

### Works Cited

- Anderson, Chris. *Makers: The New Industrial Revolution*. New York: Crown Business, 2012. Print.
- Anzaldúa, Gloria. *Borderlands/La Frontera: The New Mestiza*. San Francisco: Aunt Lute, 1987. Print.
- “Arduino—Main Page.” *Arduino.cc*. n.d. Web. 23 Aug. 2013.
- Arnulf of Milan. *The Book of Recent Deeds*. Trans. W.L. North. Orig. in Claudia Zay. *MGH Scriptorum Rerum Germanicarum 67*. Hannover: Hahnsche Buchhandlung, 1994. n.d. Web. 12 Jan. 2013.
- Banzi, Massimo. *Getting Started with Arduino*. Sebastopol, CA: O'Reilly, 2009. Print.
- Barlow, John Perry. “A Declaration of the Independence of Cyberspace.” Electronic Frontier Foundation, 8 Feb. 1996. Web. 23 Aug. 2013.
- Beller, Jonathan. *The Cinematic Mode of Production: Attention Economy and the Society of the Spectacle*. Lebanon, NH: Dartmouth, 2006. Print.
- Benjamin, Walter. “The Work of Art in the Age of Mechanical Reproduction.” *Illuminations: Essays and Reflections*. Ed. Hannah Arendt. Trans. Harry Zohn. New York: Schocken, 1950. Print.
- Bogost, Ian. *Alien Phenomenology, or What It's Like to Be a Thing*. Minneapolis: U of Minnesota P, 2012. Print.
- . *Unit Operations: An Approach to Videogame Criticism*. Cambridge: MIT Press, 2006. Print.
- Bowyer, Adrian. “Philosophy Page.” *RepRapWiki*. RepRap.org, 21 July 2006. Web. 23 Aug. 2013.
- . “RepRap Achieves Replication.” *A History of RepRap Development: Posts from the RepRap Blog*. RepRapWiki. RepRap.org: 895. PDF file.
- Butler, Judith. *The Psychic Life of Power: Theories in Subjection*. Stanford: Stanford UP, 1997. Print.
- Chun, Wendy Hui Kyong. “The Enduring Ephemeral, or the Future Is a Memory.” *Critical Inquiry* 35 (Aug. 2008): 148–71. Print.
- . *Programmed Visions: Software and Memory*. Cambridge: MIT Press, 2011. Print.

- Crandall, Jordan. "Ontologies of the Wayward Drone." *CTheory* tbc027. 2 Nov. 2011. Web. 23 Aug. 2013.
- DeLanda, Manuel. *Intensive Science and Virtual Philosophy*. London: Bloomsbury Academic, 2002. Print.
- Dunn, Alison. "The Father of Invention: Dick Morley Looks Back on the 40th Anniversary of the PLC." *Automation Magazine*. 12 Sept. 2008. Web. 23 Aug. 2013.
- Foster, Thomas. *The Souls of Cyberfolk: Posthumanism as Vernacular Theory*. Minneapolis: U of Minnesota P, 2005. Print.
- Foucault, Michel. *The Birth of Biopolitics: Lectures at the Collège du France 1978–1979*. Trans. Graham Burchell. London: Picador, 2010. Print.
- Fuller, Matthew. *Media Ecologies: Materialist Energies in Art and Technoculture*. Cambridge: MIT Press, 2007. Print.
- Galloway, Alexander R. *The Interface Effect*. London: Polity, 2012. Print.
- . "Machines and Operators." *Gaming: Essays on Algorithmic Culture*. Minneapolis: U of Minnesota P, 2006. Print.
- Gershenfeld, Neil. *Fab: The Coming Revolution on Your Desktop--from Personal Computers to Personal Fabrication*. New York: Basic Books, 2007. Print.
- Gibb, Alicia M. "New Media Art, Design, and the Arduino Microcontroller: A Malleable Tool." Master's Thesis. Pratt University. Creative Commons Attribution-NonCommercial-ShareAlike. 2010. Print.
- Gibson, William. "Google's Earth." *New York Times*. Nytimes.com, 31 Aug. 2010. Web. 23 Aug. 2013.
- Graether, Eberhard, and Florian Meuller. "Joggobot: A Flying Companion as Flying Companion." CHI 2012. Austin, TX. 5 May 2012. Interactive Presentation.
- Haraway, Donna J. "A Cyborg Manifesto: Science, Technology, and Socialist-Feminism in the Late Twentieth Century." *Simians, Cyborgs, and Women: The Reinvention of Nature*. New York: Routledge, 1991: 149–81. Print.
- . "Situated Knowledges: The Science Question in Feminism and the Privilege of Partial Perspective." *Feminist Studies* 14.3 (Fall 1988): 575–99. Print.
- Harmon, Graham. *Prince of Networks: Bruno Latour and Metaphysics*. Melbourne: re.press, 2009. Print.

- Hayles, N. Katherine. "Distributed Cognition in/at Work: Strickland, Lawson Jaramillo, and Ryan's slippingglimpse\*." *Frame* 21.1 (2008): 15–29. Print.
- . *How We Became Posthuman: Virtual Bodies in Cybernetics, Literature, and Informatics*. Chicago: U of Chicago P, 1999. Print.
- Igoe, Tom. *Making Things Talk: Practical Methods for Connecting Physical Objects*. Sebastopol, CA: O'Reilly, 2007. Print.
- Jones, Matt. "Gardens and Zoos." *BERG*. 6 Jan. 2012. Web. 10 Mar. 2014.
- Jones, Rhys, Patrick Haufe, Edward Sells, Pejman Iravani, Vik Olliver, Chris Palmer, and Adrian Bowyer. "RepRap—The Replicating Rapid Prototyper." *Robotica* 29 (2011): 177–91. Print.
- Jones, Steven E. *The Emergence of the Digital Humanities*. London: Routledge, 2013. Print.
- Kelty, Christopher M. *Two Bits: The Cultural Significance of Free Software*. Durham, NC: Duke UP, 2008. Print.
- Kirschenbaum, Matthew. *Mechanisms: New Media and the Forensic Imagination*. Cambridge: MIT Press, 2012. Print.
- Kroker, Arthur. *Body Drift: Butler, Hayles, Haraway*. Minneapolis: U of Minnesota P, 2012. Print.
- Lente, Harro can, and Arie Rip. "Expectations in Technological Developments: An Example of Prospective Structures to be Filled in by Agency." *Getting New Together: Studies in Making Sociotechnical Order*. Ed. Cornelius Disco and Barend van der Meulen. Berlin: De Gruyter, 1998. Print.
- Levin, Josh. "Solitare-y Confinement: Why We Can't Stop Playing a Computerized Card Game." *Slate.com*, 16 May 2008. Web. 18 Mar. 2013.
- Latour, Bruno. *Science in Action: How to Follow Scientists and Engineers Through Society*. Cambridge: Harvard UP, 1988. Print.
- Manovich, Lev. "What is New Media?" *The New Media Theory Reader*. Ed. Robert Hassan and Julian Thomas. New York: Open UP, 2006. Print.
- "Massimo Banzi: How Arduino is open-sourcing imagination." *TED Talks*. TED Conferences, June 2012. Web. 12 Feb. 2013.
- Mister Jalopy. "The Maker's Bill of Rights." *Make Magazine*. Makezine.com, n.d. Web. 23 Aug. 2013.



- Montfort, Nick. "Continuous Paper: The Early Materiality and Workings of Electronic Literature." MLA Convention. Philadelphia, PA. 28 Dec. 2004. Conference Presentation.
- "More Than Just Digital Quilting." *Economist Online*. 3 Dec. 2011. Web. 23 Aug. 2013.
- Morley, Dick. "Don't Fall in Love." *Manufacturing Systems Magazine* (Feb. 1994): 40. Print.
- Morozov, Evgeny. "Making It." *The New Yorker*. 13 Jan. 2014. Print.
- . *To Save Everything, Click Here: The Folly of Technological Solutionism*. New York: PublicAffairs, 2013. Print.
- Noble, David F. *Progress Without People: In Defense of Luddism*. Chicago: Charles H. Kerr, 1993.
- O'Sullivan, Dan, and Tom Igoe. *Physical Computing: Sensing and Controlling the Physical World with Computers*. Mason, OH: Cengage Course Technology, 2004. Print.
- Parikka, Jussi. "Dust and Exhaustion: The Labor of Media Materiality." *CTheory*. CTheory.net, 2 Oct. 2013. Web. 2 Oct. 2013.
- Parker, Mike. "Man hunt for ex-soldier who shot police chief's daughter and killed policeman." *Express*. Express.co.uk 10 Feb. 2013. Web. 10 Mar. 2014.
- Premerlani, William, and Paul Bizard. "Direction Cosine Matrix IMU: Theory." Draft Paper. Gentlenav Wiki. Google, 17 May 2009. PDF file.
- "Project Tango." *ATAP Project Tango*. Google, 20 Feb. 2014. Web. 21 Feb. 2014.
- Prusa, Josef. "Prusa Mendel." RepRapWiki. RepRap.org, 15 Sept. 2012. Web. 23 Aug. 2013.
- Raskin, Jef. "Down with GUIs!" *Wired* 1.06 (Dec. 1993). Archived at *Wired.com*. n.d. Web. 23 Aug. 2013.
- Reas, Casey, and Benjamin Fry. *Getting Started With Processing*. Sebastopol, CA: O'Reilly, 2010. Print.
- Shiffman, Daniel. "Interview with Casey Reas and Ben Fry." *Rhizome.org*. 23 Sept. 2009. Web. 23 Aug. 2013.
- Star, Susan Leigh, and James R. Griesemer. *Sorting Things Out: Classification and Its Consequences*. Cambridge: MIT Press, 2000. Print.
- Sterling, Bruce. "An Essay on the New Aesthetic." *Beyond the Beyond*. *Wired*, 2 Apr. 2012. Web. 10 Mar. 2014.

—. *Shaping Things*. Cambridge, MIT Press, 2005. Print.

Stone, Allucquère Rosanne. *The War of Desire and Technology at the Close of the Mechanical Age*. Cambridge: MIT Press, 1996. Print.

“TED reaches its billionth video view!” *TED Blog*. TED Conferences LLC, 13 Nov. 2012. Web. 10 Mar. 2014.

Weiser, Mark. “The Computer for the 21<sup>st</sup> Century.” *Scientific American* (Sept. 1991): 78–89. Print.

Wiener, Norbert. *Cybernetics: or Control and Communication in the Animal and the Machine*. Cambridge: MIT Press, 1965. Print.

Winner, Langdon. “Do Artifacts Have Politics?” *The Whale and the Reactor: A Search for Limits in an Age of High Technology*. Chicago: U of Chicago P, 1986: 19–39. Print.