ghosts ; replicants ; parasites — Excavating the Spectravideo CompuMate

These are the notes I used for my final presentation in the summer Media Archaeology Class, alongside images I used as slides. As such, they're quite provisional, and once I have some time to hammer them into more coherent thoughts, I'll update this post accordingly!

So this presentation is about articulating and beginning the work of theorizing what I'm provisionally calling "computational parasites." This is provisional because I don't particularly like the term myself but I figured it would be good to give it a shorthand so I don't have to be overvague or verbose about these objects and practices throughout this presentation.

[youtube https://www.youtube.com/watch?v=hB6eY73sLV0]

As most of you know, I came to this class with a set of research questions about a particular hack of the SNES game *Super Mario World*, wherein a YouTube personality was able to basically terraform the console original into playing, at least in form although we can talk about content, the iPhone game *Flappy Bird*. This video playing behind me is that hack. This hack is evocative for me for the way it's 1) really fucking weird, in terms of pushing hardware and software to their limits, and 2) begins to help me think through ideas of the lifecycle of software objects, to pilfer a phrase from a Ted Chiang novella, and how these lifecycles are caught up in infrastructures of nostalgia, supply chains, and different kinds of materiality.

But as fate would have it, I haven't spent that much time with this hack this week because I got entranced by a different,

just as weird object: the Spectravideo CompuMate.

This is a hardware object down in the Depot that I've been messing around with. You plug it into an Atari 2600 and it transforms the game console into a functional, though extraordinarily rudimentary, microcomputer. Or rather, it reveals the console as always already a functional microcomputer (indeed, one that's actually more powerful than the CompuMate leads you to believe). I wish that I could say that the language of "parasite" is my own. It's not, it's Darren's. But I'm glad that he said "parasite" around me and this object because that framework has helped me start to think about different theoretical articulations for thinking through the *Super Mario World/Flappy Bird* hack, as well as some other objects and practices in video game hacking culture that have been long-standing fascinations of mine.

What ties these two objects together is how they terraform the video game object into a functional programming environment. That's basically how the *Super Mario World* hack works: a series of glitches renders the program receptive to then allowing a payload to be written byte by byte and then executed arbitrarily. It's not inconceivable that, if you had the time and energy and precision, you could write in a game more formally complex than *Flappy Bird*. The same is sort of true with the CompuMate: you could, in theory, program really "complex" things with it. This is the "parasite" that I'm interested in thinking through: physical hardware configurations that harness different kind of software productions towards increasingly strange and evocative ends.

So part of this presentation is really about beginning to bring these threads together. I have nothing but provisional claims, and much of this is a little show-and-tell-y, but I

hope you'll bear with me.

—

Let's start by traveling back in time to a world that may or may not have actually existed anywhere outside of advertising copy. It is 1983. You are the parent of a child in her early tween years. A few years back you purchased an Atari so that your child and your family and you could play *Space Invaders*. You like *Space Invaders*. But now it's 1983. Where there used to just be game consoles there are now computers that you can buy for your own home. They are catastrophically expensive, but your child is entranced by them, and you're also thinking more and more about whether or not you need one of these computer doohickeys.

Enter the CompuMate. The CompuMate's advertising copy tells you that you don't need to buy another computer, because didn't you know you already had one in your home? Yes indeed, plug the CompuMate into your Atari and watch it turn into a fully-functional computer! Indeed, as the manual tells us, you have made a "wise decision."

PART ONE

CHAPTER 1
A New Beginning

The introduction of the Atari® VCS 2600 video game machine in 1978 revolutionized the world of toys. Children and adults alike were now able to play a new kind of game. With its flashy graphics and hair-raising sounds, the Atari® machine was much more exciting than any toy truck or toy doll.

However, the home computer became the hit of the 1980's and probably will continue as such well into the future. Millions of Atari® game machine owners had little choice but to give up their little Atari® machines and buy expensive personal computers in order to join the microcomputer generation. That is until now, with the introduction of Spectravideo's Compumate™.

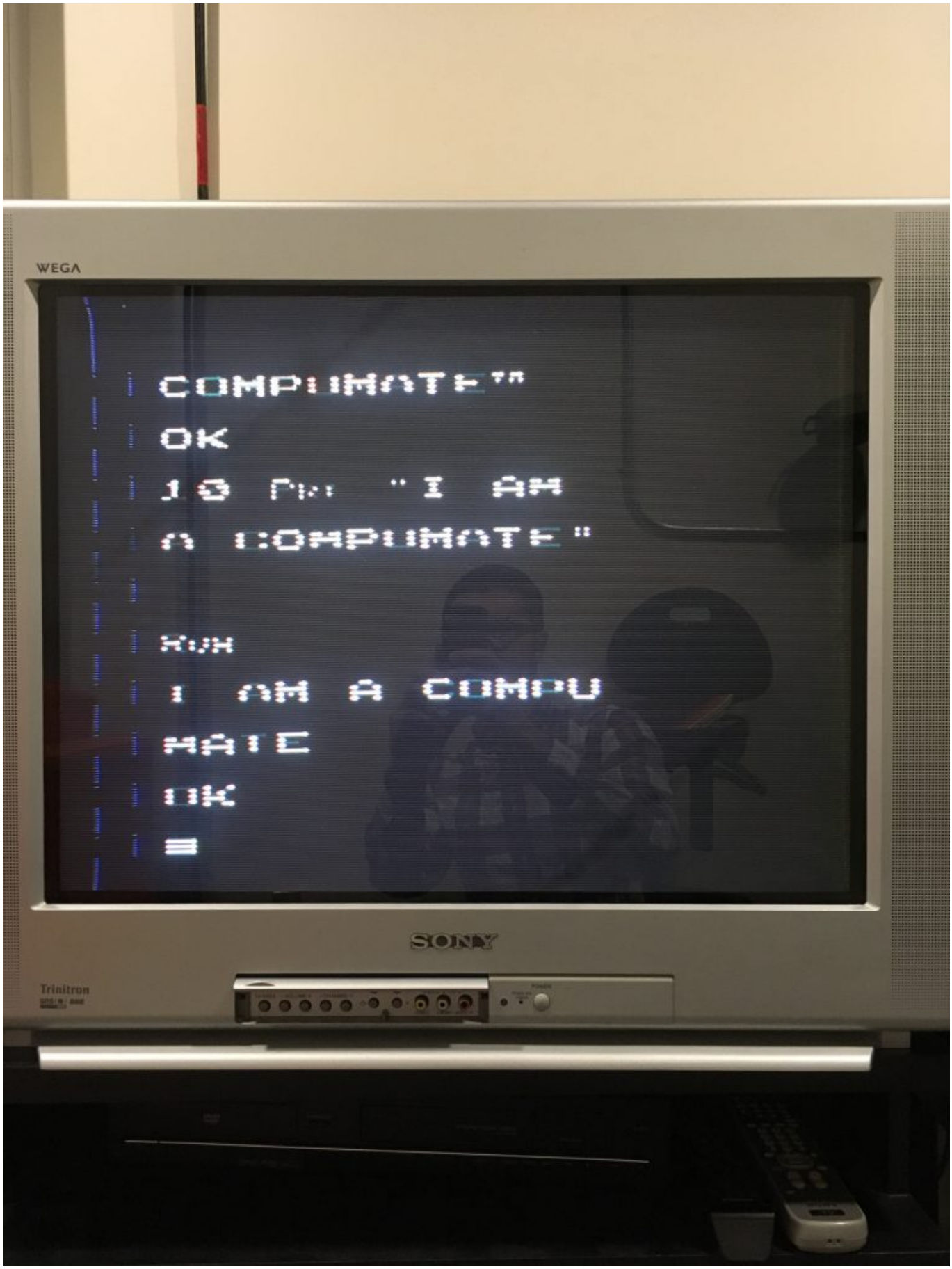
You have made a wise decision in selecting the Compumate™, which enables you to transform your Atari® game machine into a real computer! You already know that you can play games with your VCS, but we will show you that now, with Compumate™, you can also play music, draw pictures, print words and even learn how to program a computer. And by the time you finish reading this guide, you will agree that computers can be easy and fun to use.

The reality, as is usual with these sorts of things, is a little less entrancing, but far more strange. The CompuMate is

a tentacular beast. It’s principally two components, a cartridge with 16 kilobytes of ROM and 2 kilobytes of RAM and the worst keyboard ever made. (It’s a membrane keyboard so it’s kind of like typing on a food processor.) The ROM holds three programs—a stripped-down BASIC interpreter; a two-channel sound editor; and a rudimentary graphics program.

There are also two audio ports on the keyboard that are actually for storing programs, songs, and images to cassette tape. I wasn’t able to get my hands on a tape to test this, so that’s for the next time I come to visit the Depot.

These are really simple programs that interestingly map out some of the formal constraints of the Atari as a system. Since the CompuMate doesn’t add any processing power, the limitations of the program give some insight into the hardware configurations of the Atari itself. Here are some of the things I was able (or not able) to do with it.



1. Programming. The programming one is able to do with the CompuMate is more allusive than anything else, particularly

given that you don't have full access to BASIC as a language. For example, I wanted to try the famous one-liner from the collaboratively written book *10 PRINT CHR$(205.5+RND(1)); : GOTO 10*, which would print a maze. But I was immediately stymied by the fact that the only functions available to me were a suite of pre-selected ones that you can see on the keyboard itself. I'll say more about this palimpsestic keyboard in a moment, but you might see that while I have PRT and RND and GOTO as functions, I don't have CHR…or a semicolon. So the character set is limited as well. I was mostly resigned to a lot of "LET A = 10 AND B = 20 AND C = A + B AND PRNT C" kinds of programs. Demonstrative, but not powerful.

> *Philip Glass, "Opening," on the glorious two-channel CompuMate at @residualmedia pic.twitter.com/Vbn202Yc18*
>
> *— Jeffrey Moro (@jeffreymoro) May 25, 2017*

//platform.twitter.com/widgets.js

2. Music. Music is a little bit more expressive. I'm limited to two channels, which is a limitation of the Atari's TIA chip. In those two channels, I can write fifteen different notes and a rest; I assumed that these sixteen bytes are coded into the CompuMate's ROM and don't represent the full range of sounds available to the Atari. I can also set durational values to the notes, which actually makes writing music into this interface a little confusing, because just because notes are next to each other between channels doesn't actually mean they're happening at the same time. But it does let me do some fun stuff with polyrhythms. After about an hour of plodding with the keyboard, I was able to write the first four measures of Philip Glass's "Opening."

3. Graphics. The graphics are a 40 x 40 grid that you draw one pixel at a time. You have ten different colors but the screen can only handle two at a time, so you have to pick one for the

background and one for the foreground—which is also how the Atari draws its in-game graphics. Interestingly you can chain together nine drawings saved to the RAM into a rudimentary animation, but I haven't tried out how to do that yet.

The Keyboard. I want to say a bit about the keyboard right now because it's such an information-dense artifact. All of these programs have systems laid on top of each other on the keyboard—there are necessary FUNC values and SHIFT values and buttons are contextual depending on the program running. I was able to find a ROM dump of the CompuMate's programs that I can run in the Stella Atari emulator, but without the keyboard it's very difficult to orient myself. Back at Maryland, we have a Kaypro IV word processor with a keyboard overlay with tricks and reminders for how to use WordStar (this image is a Kaypro 10, not IV, but the principle is the same). I think of this keyboard in a very similar way—a remnant of a moment

wherein one had to be trained into regarding the computer as a particular kind of physical artifact, with particular habits, poses, and physical relationships. I also like this keyboard because it's just as unwieldy as the CompuMate itself. It's trying just a little bit too hard—is just a little too insistent on walking you through how to work it.

In Nick Montfort's and Ian Bogost's book on the Atari, *Racing the Beam*, which I'm sure many of y'all have read before, they argue that a key conceptual underpinning of the Atari is one of imitation. The Atari's games, at first, imitated those available in arcades, or imitated other kinds of real-world sports games. The choice to use cartridges, they argue, also constituted a strategic imitation of the personal computer—cartridges increased versatility and made the device feel more useful in the home. The CompuMate works alongside this sense not only in its function, but how it was sold.

I was able to track down this ad for the CompuMate that I found really telling. The copy at the top says, "With Spectravideo's Compumate and your Atari VCS, you can PLAY PROGRAMMER, PLAY COMPOSER, OR EVEN PLAY PICASSO!" and goes on with the rhetorical question, "Why just play video games?" Certainly these are claims to a kind of skill building or strategic utility to the computer, but I'm interested in what kinds of ideological work the verb "to play" is doing here. There's "play" in the sense that one can certainly have a limited amount of fun with these programs. But "play" also seems to suggest a kind of approximation—"playing at" programming, "playing at" drawing. It's play in the sense of "playing house." And it's a space where play becomes explicit training into a particular way of being with a computer—and it's interesting to me that the site of this training is the video game console, which otherwise seems to go out of its way to design itself *away* from being a functional computer. Indeed, the CompuMate doesn't do anything to add computation to the Atari, it just reveals the computation running

underneath *Space Invaders,* after a fashion.

So I want to actually try to theorize some of these concerns here at the end. I think you can see that there are some clear sympathies between the CompuMate and the *Super Mario World/Flappy Bird* hack. Both are about making a closed computational system open to various kinds of programmable, iterable input, and then using that input to ends perhaps not previously imagined by the creators of those programs. Spectravideo was a fairly bootleg operation, and had nothing to do with Atari itself. Indeed, if you work your way through the CompuMate's manual, you find at the end that the whole enterprise is a kind of elaborate upsell for Spectravideo's own microcomputer, the SV-318.

In the spirit of media archaeology's repeated image of "dead media" that we discussed yesterday, I've been trying to theorize some of these issues around ideas of haunting. Haunting and hauntology has so far felt a useful mode to me, particularly when thinking about the Mario/Flappy Bird hack, because it proposes particular kinds of ghostly or oscillating spatiotemporal relations, akin to Derrida's initial formulation of the hauntological as that which is deferred in presence and absence, life and death. I also find Wendy Chun's idea of the "enduring ephemeral" useful here.

But the CompuMate is starting to challenge some of my thinking, or at least provide some new territories to expand beyond a dialectic of living and dead that ghosts seems to traffic in. Particularly, as I've repeatedly suggested, I'm getting interested in the figure of the *parasite*, which I think has an advantage of complicating temporal relationships—forcing us to think about different kinds of collapsed spatiotemporalities that emerge out of these hacks and manipulations—while also foregrounding physical relationships, literal parasitism.

In conversation with a few of you, I've talked about taking up

Michel Serres' *The Parasite* in starting to map new theoretical directions for this work. Serres' book is about human relations and communicative networks, and relies on an untranslatable French pun, wherein *parasite* means both parasite and static in English. A discussion of parasitism becomes one not only of production and consumption, but also signal and noise. The parasite, Serres writes, "is a thermal exciter."

> *Far from transforming a system, changing its nature, its form, its elements, its relations and its pathways . . . the parasite makes it change states differentially. It inclines it. It makes the equilibrium of the energetic distribution fluctuate. It dopes it. It irritates it. It inflames it. Often this inclination has no effect. But it can produce gigantic ones by chain reactions or reproduction. Immunity of epidemic crisis. (191)*

It takes and does not give, but nevertheless seems to invigorate the system. Indeed, when thought of as static, in information theory, it constitutes the system as such, is the ever-present-but-excluded-third between channels of signification. The parasite enters in and paradoxically makes new possibilities.

I'm probably running long so I'll come to some provisional points. Thinking of these objects as parasites helps me push past the living/dead binary towards different kinds of circulations of energies and possibilities—how these objects take processing power without returning the results expected by the system. At the same time, it also seems to suggest interesting ways forward in theorizing how these devices actually function alongside human operators. How long it takes to input this information, byte by byte. If it is a parasitism, it is slow, complex, and introduces plenty of space for noise and failure in itself.

Works Cited

Bogost, Ian and Nick Montfort. *Racing the Beam: The Atari Video Computer System*. MIT P, 2009.

Montfort, Nick et al. *10 PRINT CHR$(205.5+RND(1)); : GOTO 10.* MIT P, 2013.

Serres, Michel. *The Parasite*. Translated by Lawrence Schehr. U of Minnesota P, 2007. 1982.