

## Media Archaeology Class/Lab Assignment

Assignment by Matthew Kirschenbaum, University of Maryland (@mkirschenbaum, mgk@umd.edu).

Audience: Graduate students or advanced undergraduates with hand-on access to legacy hardware and software.

Please feel free to share and reuse, though I would appreciate acknowledgement where appropriate. I'm grateful to Lori Emerson for sharing ideas, which have informed some of the questions asked of the students in phase I.

### Phase I

#### In Brief

There are six antique computers set up for you to experiment with at MITH. In this, Phase I of the Media Archaeology module, you will learn basic facts about the pedigree of each, figure out how to turn it on (not always obvious!), and perform a few simple tasks to get a feel for how it works. You will work individually for this first phase of the module, logging your activity with each machine via GitHub. (Feel free to ask a classmate for help if you're genuinely stuck on something, but only after you've made an honest effort on your own.)

For the second phase of the module (to be distributed) you will work in pairs or small groups, choosing one or two computers for a more detailed exploration which may involve building or making something with its original software, and/or doing deeper historical and analytical work.

You will generally have access to the computers any time MITH is open to the public, which is Mon-Fri, 9-5. Feel free to contact me with any logistical or technical problems. Phase I of this module is due by October 27.

#### Safe Handling

Some of these computers are more than three and a half *decades* old! They built 'em strong back then so you don't have to handle them with kid gloves, but please do treat the hardware with respect. To that end:

- Absolutely no food or drink on the tabletop.
- Some of the plugs and connections holding everything together are loose or delicate. If a machine doesn't turn on check that all of its components are snugly plugged and connected, and that there is power going to it. Use common sense to try troubleshooting.
- Never force anything. If something doesn't seem to want to work or go or bend or fit, leave it be and ask me.
- Please keep fingers off of the recording surface of the diskettes. Also keep magnets away!
- All of the computers except the two Macs can simply be switched off at any point (though you will lose your work unless it's been saved to a disk). The Macs should always be shut down using the appropriate option under the "Special" heading of the home menu (top of screen).
- When you turn a computer off, wait about 10 seconds before turning it back on.
- Please neaten up the workspace around each computer when you finish. Return diskettes to sleeves and holders, stack any books in an orderly fashion, etc.
- Above all: please remember to turn these computers **OFF** when you are done with them. Remember, the learning experience for everyone starts with figuring out how to turn them on!

Moreover, they won't go to "sleep" on their own, and whatever is left of their lifespan will be diminished if they are left running.

### The Exercise

For each computer, answer the following questions. You should feel free to consult any reference sources you spot around MITH (check the spline!), and/or Wikipedia or other online sources.

- What year was it made?
- What operating system does it run?
- How much did it cost (then and now, adjusted for inflation)? How hard is it to find one on the secondhand market (check eBay and Craig's List).
- In brief (a short paragraph), what were its historical strengths and weaknesses as a platform? What was it best known for?

Hands-on exercises specific to each computer follow. Everyone should complete all of them. If you're stuck try reading some documentation, either from the books and manuals at MITH or else online.

Make some additional notes about each machine as you go: What's the most notable difference from a computer that you're used to? Can this computer and its software do anything that your current computer cannot?

- *Kaypro 4*

Turn it on (you must figure out how). Start WordStar (you must figure out how). Try typing a simple document, experimenting with some commands based on the keyboard template. (You don't need to save your document.)

- *Apple IIe*

Turn it on (you must figure out how). Work your way through the *Apple Presents Apple* tutorial. Try a game or two. (If you can beat one of my high scores on *Sea Dragon* you'll get a prize. Really!)

- *Macintosh SE*

Turn it on (you must figure out how). Read some of Deena Larsen's *Marble Springs*. Quit *Marble Springs*. What else is on this Mac?

- *Vectrex*

Turn it on (you must figure out how). Play *MineStorm*. Try another cartridge. Why do you think the designer's would have opted for the "vector" graphics?

- *Amiga*

Turn it on (you must figure out how). Try the demos. What else can it do? (Hint: try the "Amiga Appetizer" disk). Have a look at the Amiga magazines and the book, *The Future Was Here*.

- *Macintosh IIci*

Turn it on (you must figure out how). Open WordPerfect. Open MS Word. Try typing a simple document in each. Which do you like better, and why? Open MacPaint. Try drawing something.

(You don't need to save any of your work from these applications.) What else is on this Mac? What can you learn about its past user(s)?

## Phase II

"We no longer have access to digital tools for making," insists Lori Emerson in her book *Reading Writing Interfaces* (3). This second phase of the media archaeology module seeks to transport us back to an earlier moment in time so we can see what she might mean by contrast. In general then, the objective is to use one of the available systems and its software to build or to make something. You have broad latitude as to content (your work can be fiction or non-fiction) as well as many options for format, ranging from programming to writing hypertext to digital drawing or 3-D printing. Try to experiment and take risks: I will be looking at how creatively you use your chosen platform.

I have given some outlines for possible projects below. Choose just one! You may also think of other combinations of systems and software to do things I haven't anticipated—please feel free to discuss with me. You may work either individually or in small groups, which you can form at your own discretion. (Groups should probably be no bigger than four persons.) The deliverables for this phase of the assignment will consist of whatever it is that you make, coupled with a coversheet (delivered appended to this document via Github) with the following categories:

- The title of the work or project.
- Everything I need to know to find it (computer, diskette, folder, and file name).
- If more than one individual is involved, list the members of the group and their roles. For a larger group it is especially important to give me **detailed** information as regards the nature of each person's contribution.
- A 500-word (maximum—but: see "Writing Machines," below for exception) statement describing your objectives for the project, working habits, problems encountered, and self-assessment of the deliverable.

Projects should be possessed of some modest substance. Not the kind of effort you would put into writing a seminar paper, but something more than just a quickie little demo. You'll have to use your judgment and get a feel for what seems appropriate in the time available; but the bigger the group, the greater my expectations will be.

**ESSENTIAL:** Because it is not inconceivable that a computer or disk could fail at some critical moment (**taking both your own and others' work with it**), you must devise some alternative ways to document your progress. Taking pictures or video of the screen with your phone is one good idea—you may think of others. You are not responsible if a machine fails of course, but you *are* responsible if it fails and you have no other documentation to show me.

Phase II of the module is due by class time on November 10.

### 1. Garden of Forking Paths

Write a branching choose-your-own-adventure text with Apple BASIC. It can either be fiction or non-fiction. You'll need to learn at least the following commands: PRINT, GOTO, IF/THEN, and probably also

how to use some simple variables. You can even include comments in your code, using the REM statement. You'll also need to learn basic disk handling commands like how to load and save files, as well as listing and running them.

System: Apple IIe

Saving: Use the diskette labelled "DOS 3.3 Sample Programs." This loads Apple BASIC into the computer's memory. You can save your files to this same diskette.

Resources: The Apple II reference books at MITH, Google.

Hints: LIST some existing programs to see how they work. "Tron" is a good one.

## 2. Hypertext Hotel

Using **either** Storyspace **or** HyperCard, create a hypertext. It can either be fiction or non-fiction. (Of these two, HyperCard is probably the easier to work with, and has better documentation.)

Systems: Macintosh SE (Storyspace 1.5), Mac IIci (HyperCard 2.2).

Saving: Create a sub-folder for your own work in the folder on the desktop labelled "ENGL668K".

Resources: Storyspace and HyperCard reference books at MITH, Google. Also be sure to take the Tour in HyperCard.

Hints: Spend some time with *Marble Springs* (HyperCard) and/or *We Descend* (Storyspace). They'll give you some feel for the poetics of the form. Look at the demo HyperCard stacks. Also read around in Landow's *Hypertext* book.

## 3. Andy's Future Was Here

Use the multimedia capabilities of the Amiga to create a portfolio of pictorial work (just like Andy Warhol once did: <http://www.wired.com/2014/04/an-amazing-discovery-andy-warhols-seminal-computer-art/>). Some programs to consider working with: AmigaBASIC on the "Basic Extras 1.2" disk; PAINT on the "Appetizers" disk; GIZMOZ II; GRAPHICRAFT V. Others are certainly possible too: these are just suggestions. Play around a bit to see what you like.

System: Amiga.

Saving: Feel free to save on any of the Amiga disks (just be sure you and I both know where).

Resources: *The Future Was Here* and other Amiga documentation at MITH; Google.

Hints: For a given program, see if there are some demos available to give you ideas of what's possible.

## 4. Writing Machines

As he confided on the Conan O'Brien show last year, George R. R. Martin still writes his books using WordStar (you can find the clip on YouTube). The objective of this scenario (which is structured a little differently from the others) is to try to figure out what the appeal is. First, select an existing 250-500 word text (how about the beginning of *A Game of Thrones?*), and type it into both WordStar on the Kaypro *and* using a manual or electric typewriter (which you'll have to source yourself—the MT/ST Selectric is not in

working order). When using WordStar, exercise its commands to produce an exact transcription—try inserting and deleting, moving around a block of text, formatting text, and so forth; when typing, by contrast, don't worry about creating a perfect page—just type up your text, typos and blemishes and all. If, as Nietzsche famously said, our writing materials are shaping our thinking, what impact might each of these machines have on the way a writer would work? How different are they from what you're accustomed to today, and do you think that difference could ever be productive? Address this in a statement of up to 1000 words that you prepare on either the Kaypro or the typewriter (so unlike the other options, your project statement will not be submitted via Github, but you still must complete a coversheet with the other fields filled in).

Systems: Kayrpo, a manual or electric typewriter.

Saving: WordStar files on the Kaypro may be saved directly to the WordStar diskette. For the typewriter, be sure to turn in the original typewritten sheets to me.

Resources: Reference books at MITH, keyboard stencil, Google.

Hints: No whiteout on the screen please.

#### 5. Maker Madness

This is an open-ended option to “make” something using the facilities in the McKeldin MakerSpace and/or whatever other facilities, materials, and tools you wish to bring to bear. Your project should connect to or comment on the course's themes in some way; it can absolutely be playful, but it shouldn't just be arbitrary or capricious. If you choose this option, please give me a head's up as to what you're planning ahead of time.

Systems, Saving, Resources: all variable.