ctrl-6502

A C-based MOS 6502 Emulator

By J. D. Jackson

Table of Contents

* System Background
  + Major Components
  + Standard Memory Map
  + Binary/Machine Code File Format (.ctr)
* Description of Operating Sequence
  + ctrl-6502 Power-On RESET
  + Main Loop
  + Interrupt Service Routine Handler
  + Interfacing with Emulated Hardware
  + Current Burdens (Bugs, uncertainties)
  + Future Burdens (TODO)
* References
* GPL-3.0 License

1: System Background

I know that a 6502 emulator is nothing novel by any means. I have no fantasies of pushing the boundaries of efficiency or accuracy for the field, because I know that would take an amount of time, money, knowledge of computers and physics, and unrelenting pain that I am not fond of taking on at the moment. It’s painful enough having this obsession in the first place.

My goal with this project is to increase my knowledge and comfort with larger coding projects and the tools, systems, and environments surrounding them. I’ve learned a lot about C itself, like how to interconnect large amounts of user structures and functions, and how to process audio, video, code data simultaneously. I also had to get a lot more comfortable using Make than I had ever hoped to do.

This development has been quite a journey. It has taken me to all corners of the internet- from vintage computing subreddits, to dusty 90’s tech forums full of sweaty nerds arguing about which WD CPU is the best. It’s remarkable seeing how similar everything is, yesterday’s hobbyists tinkering with Commodore 64’s and today’s with Arduinos. It blows my mind that the randoms that supply us with answers to our Google searches via the forums they frequent are truly the backbone of the internet.

Many of them aren’t even computer nerds by profession; they could just be a construction worker that became fascinated by the stop light logic circuitry, and went home and learned about it, and then one day came up with an idea, and decided to try to implement it in a state machine or something. It is just comforting and humanizing to remember that there are just regular people on the other side of the screen. I feel like this notion has been all but abandoned due to the omnipotence of our beloved soul-sucking rectangles of glass.

It used to be special to connect with someone virtually. Before you could have a computer in your house, you were even lucky if your school/job would let you sign up to run a program for an hour or two. They were largely seen as a tool, a machine, which is exactly what they are in principle. That principle changed as the role of computers changed in society, though, and computers are now additionally our entertainment, our school, our job, our social network, and even more as the days go by. That, in and of itself, is not bad.

But these days, due to the casual nature of computer use, and the frequency of our use, as well as the emotional connection to the things we are doing and consuming on the device, we project our brain/self/essence onto the existence of the phone. The user is in full control of a large amount of instantaneous power and gratification. That’s kind of like an extension of our free will. In everyday life, you see this same concept in your free will. You have the power to do anything within the limitations of the operation of your body. However, due to social rules and self care, there are certain things that are typically required of you (going to work, driving a car, brushing your teeth, etc). Beyond the surface, though, there are additional “para-behaviors” that we are expected to obey and exhibit. These are not the same thing as social rules/expectations, but para-behaviors ARE largely influenced by those. The other determiner of para-behaviors is your inner voice.

Para-behaviors are what allows people to appear “normal” in any setting. These would include things like keeping eye contact while talking, using language appropriate to the people around you, and other social expectations. These ALSO include all the things that you’re NOT expected to do (i.e. take off your clothes in the middle of a meeting), as well as things that you’re specifically expected NOT to do as a functional human (i.e. become a terrorist, other crimes).

My point is, these behaviors are not required of the user of a computer/smartphone by default. It is definitely necessary in more professional environments, just like in real life too, but there is no equivalent of needing to brush your teeth on the internet, or do the laundry, and there’s no requirement to make any sort of sense or have any thought whatsoever when talking online. There is no way to know the true intentions of another person online unless you know them, because it is very possible that that person doesn’t even know their own intentions; they could have just been messing around/trolling, or it could have typos that change the intended meaning.

Typing something out and sending it takes a lot less thought and effort than speaking a sentence. On top of that, you are largely anonymous, so it is very easy to desensitize yourself to shame or anxiety or morals. Even if you ARE known by your online audience, there is still a very reduced intake of direct attention as a result of your own actions, making it easier to do something like break up with someone or confess your feelings.

Basically, those things make us feel good and comfortable and safe indirectly because there is a lack of many of the things we dislike or are anxious about that are present in everyday life. It is an escape for humanity, digital but also truly a world in its own right. The device extends the brain of the person using it, and gives them a playground of pleasure and entertainment, with a constant yet never-satisfied promise for more.

Due to these facts, we are not human when we use our devices. At least, people who use the internet for anything more than just work (which is almost no-one at this point, even boomers use Facebook) are not humans. Much of the mundanity of life is abandoned when the screen turns on, but the mundane makes us human, it gives us context for our existence by creating a ground between the highs and the lows. Both negative and positive emotions are piped to our brain differently when we use the internet, and I feel like that is a fundamental and undeniable shift in executive integrity. It’s not for better or for worse, it’s just different. I think currently it negatively affects people much more than positively, but that can change if we change our habits.

Without further ado, here’s my fucking computer.

Major Components

1. Logic Emulation:
   1. Goal: accurately emulate MOS 6502 logic, input, output, data processing, instructions
   2. 151 legal opcodes. Illegal opcodes currently are NOP
2. Hardware Emulation:

Standard Memory Map

|  |  |  |  |
| --- | --- | --- | --- |
| **Section** | **Address Range** | **Width** | **Usage** |
| Zero Page | *$0000 - $00FF* | **256 B** | **Reserved** |
|  | *$0000 - $000F* | **16 B** | **Local variables, function arguments** |
|  | *$0010 - $00FF* | **240 B** | **Global variables, pointer tables** |
| Stack | *$0100 - $01FF* | **256 B** | **Reserved** |
|  | *$0100 - $019F* | 160 B | Nametable data to be copied during VBLANK |
|  | *$01A0 - $01FF* | 96 B | Stack memory |
| Other internal RAM | *$0200 - $07FF* | **1,536 B = 1.5 KB** | **Global and system data, variables** |
|  | *$0200 - $00FF* | 256 B | OAM data to be copied during VBLANK |
|  | *$0300 - $00FF* | 256 B | Sound player and other variables |
|  | *$0400 - $07FF* | 1024 B = 1 KB | Arrays, global variables that are used less often |

C. Binary/Machine Code File Format (.ctr)