Are Neuroscientists any Closer to Understanding a Microprocessor?

Nicholas K. Branigan

March 19, 2020

1 Introduction

In the past 20 years, neuroscientists have developed remarkable tools for acquiring neural data. In fact, methods like modern multielectrode arrays, calcium imaging, and optogenetics have placed in view a future in which we may no longer be limited by the data we can collect. But neuroscientists are interested in more than peering increasingly clearly onto increasingly great sections of neural territory. We want to *understand*. We want to understand the structure of information processing being carried out by brains, and we want to understand well enough, at least, to fix brain functions when they break. Of course, this cannot be accomplished without beautiful datasets. But, are we ready for them? If we had in hand "a matrix of the spatially and temporally resolved activity of all neurons in an animal performing an interesting behavior", what would we do with it? [3].

Jonas and Kording think that neuroscientists are not ready [1]. In their 2017 paper, they argue that current methods will fail to deliver the kind of understanding we want from the datasets we want. To make the case, they apply major analytical tools in neuroscience to the MOS 6502 microprocessor. The results are not encouraging. Of course, Jonas and Kording's argument succeeds only if there are important similarities between the MOS 6502 and the brains we are interested in. It is well known that the analogy between computers and brains can be taken too far. The systems differ in major ways, including in their stochasticity, adaptability, and complexity. Moreover, analogies of this sort have a long history, and their track record is not promising. Before computers, brains were thought to be like thermostats, weaving machines, and steam engines [2]. Nonetheless, the commonalities here appear more than trivial:

> "Both systems consist of interconnections of a large number of simpler, stereotyped computing units. They operate on multiple timescales. They consist of somewhat specialized modules organized hierarchically. They can flexibly route information and retain memory over time." [1]

My appraisal is that Jonas and Kording's argument succeeds. Importantly, this is not an acknowledgment that the neuroscientific project is hopeless. As Jonas and Kording were pointing out the shortcomings of current methods, they were pointing to the microprocessor as an exciting testbed for developing and testing better techniques. In this spirit, I apply methods we have developed in this class to the MOS 6502 to see if they offer a clearer path to understanding the microprocessor.

2 Data

Eric Jonas has made available in the `neuroproc data` repository of his GitHub much of the data used in the paper. Available are the lesion experiments, connectomics data, and full microprocessor time series. I think this last category is the most interesting component of the dataset, so that is where I focused my efforts. There are three time series, one for each game the authors simulated on the MOS 6502: Donkey Kong, Space Invaders, and Pitfall. Each game's time series is a (12799999, 3510) array. The i'th column in this matrix is the time series of transistor switching activity for the i'th transistor. It is important to understand that the entries are not of transistor state, but of changes in state. 0 says that the transistor state remained the same from the previous time step (the transistor stayed on or stayed off), while 1 says that

Optimal # latent states	Transistor #	Log-likelihood train	Log-likelihood test
1	3488	1,061,282.85	-234,798,938,716.73
3	3146	1,061,282.85	1,061,282.85
	3328	-85,363.34	-85,363.34
	3498	-85,363.34	-85,363.34
5	2893	-85,363.34	-85,363.34
10	860	812,256.77	717,736.23
	2407	908,295.23	778,834.61
18	435	848,771.55	773,180.27
	507	757,225.18	687,698.61
	2314	918,076.98	815,873.26
	3108	882,263.45	799,490.94
	3109	884,050.75	786,740.73
25	223	800,290.74	677,861.70
	752	886,952.32	800,534.44
	2294	928,625.67	804,556.76
50	880	716,821.79	605,760.03
	2924	1,055,558.44	1,008,675.60

Table 1: Summary of HMM models.

the transistor's state changed from the previous time step (it went from on to off, or off to on). The authors consider these transistor switches to be analogous to spikes and refer to this data as the chip's spiking time series. Following Jonas and Kording, I collapsed the observations into time bins of 100 time steps giving three (127998, 3510) arrays. The i'th time step for transistor j in this binned array is the total number of spikes seen on transistor j between time $i * 100$ and $(i + 1) * 100 - 1$, inclusive.

3 Methods

My hypothesis is that for some state space model, the number of latent states that best explains the spiking time series of a transistor can reveal the role of the transistor in information processing. If transistors have in common this hyperparameter setting under one of these models, I want to see whether they also work in similar functional capacities. The state space models I examined were Hidden Markov Models (HMMs) and Autoregressive Hidden Markov Models (ARHMMs). I randomly selected 20 transistors on which to conduct my analysis. I then fit HMMs and ARHMMs to the binned time series for each of the 20 transistors. I set out to find the number of latent states that best explained the behavior of each transistor under the HMM, and separately under the ARHMM. I chose Donkey Kong as a training set and Space Invaders as a test set. The latent states I explored were

$$[1, 3, 5, 8, 10, 18, 25, 50].$$

I fit the HMMs and ARHMMs using the Linderman Lab's State Space Models package. I used 300 iterations and declared convergence if the absolute value of the change in log likelihood between iterations was less than 1. 300 is a fairly large number of iterations, but I found that it was often necessary, especially on some of the models with higher numbers of latent states. I used a Gaussian model for the observations and the Expectation Maximization (EM) algorithm. Since it is well known that EM is susceptible to local maxima, for each candidate number of latent states, I fit the model three times using a random initialization each time. I chose the best result of the three initializations by log-likelihood on the train data to compare against the models fit with different numbers of latent states. Then, I compared the models with the eight different latent state settings and chose the winner by log-likelihood on the test data. Early experiments I ran suggested that over-fitting was possible on these data, so I use the comparison on the test data to guard against that.

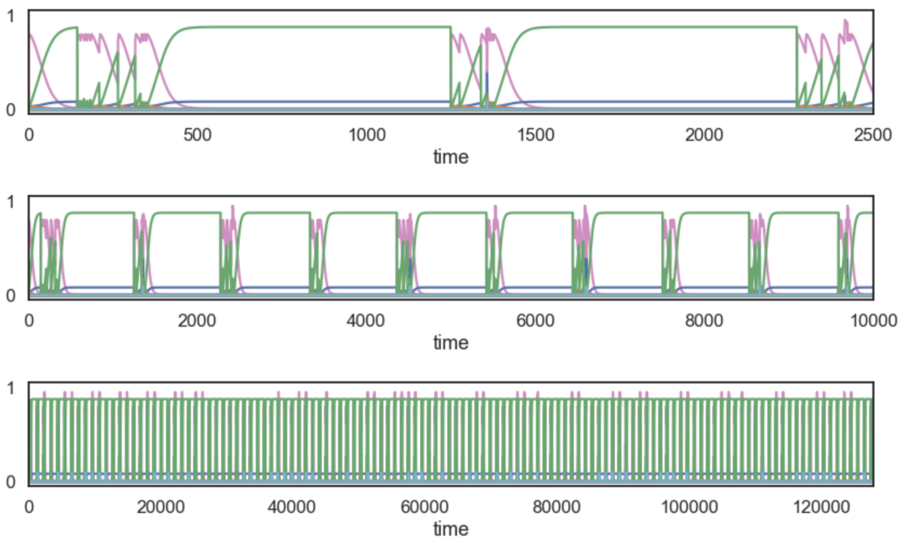

Figure 1: Posterior on latent states for transistor 2924 for three time windows.

4 Discussion

From Table 1, it is apparent that under the HMM, with respect to the number of latent states, one size does not fit all. In fact, we see that seven out of the eight latent states explored in my hyperparameter search were optimal for at least one of the twenty transistors. The story is not much different under the ARHMMs. There too, seven out of the the eight states were optimal for at least one transistor (see Table 2 in the appendix). The similarity between the HMM and ARHMM results is a general theme. Comparing Tables 1 and 2, we see that many of the transistors that were clustered together under the HMM (by optimal latent state) are also clustered together under the ARHMM. Moreover, the number of optimal latent states for a specific transistor tends not to change much between the two models. Sometimes there is movement between adjacent buckets (from 3 to 5 latent states), but large jumps are rare. An exception is transistor 3488, which jumps from 1 latent state under the HMM to 50 under the ARHMM. But this outlier is not especially worrying. This transistor is completely inactive under Donkey Kong and slightly active under Space Invaders. The total inactivity in the training data leads to a very poor fit, which is obvious from the difference in log-likelihoods for this transistor between the train and test data. As a result of the substantial similarity between the two modeling approaches, I focus on the HMM results.

The spiking on these transistors tends to exhibit substantial regularity over time. In Figure 1, we can see that this is nicely reflected in the posterior for transistor 2924's 50 state HMM. The lowermost component of the figure is the posterior over the full 127998 bin time series. The component above zooms in on this, and the component above that zooms in further. The inferred latent states for 2924 can be found in Figure 4 in the appendix. Interestingly, from Figure 1, we see that only a handful of states are occupied with high probability. Nonetheless, the 50 state model for 2924 outperformed every lower state model on the test data. Other transistors have similar posteriors to 2924. In Figures 5 and 6 in the appendix, we can see that transistor 3109 exhibits similar regularity in the behavior of the posteriors over time. Moreover, while 3109 also seems only to occupy a small number of states with substantial probability, it is best explained on the test data with 18 states.

Figure 2 provides so far the best test of my hypothesis. A few of the transistors I have randomly selected turn out to be physically near each other in the architecture of the MOS 6502. There are two pronounced clusters of transistors by chip location, one in the top center and the other in the top right of Figure 2. The three transistors in the first physical cluster are all best modeled by 25 latent states, and two of the three in the second cluster are best modeled by 18. Moreover, the two transistors that are the closest to each other

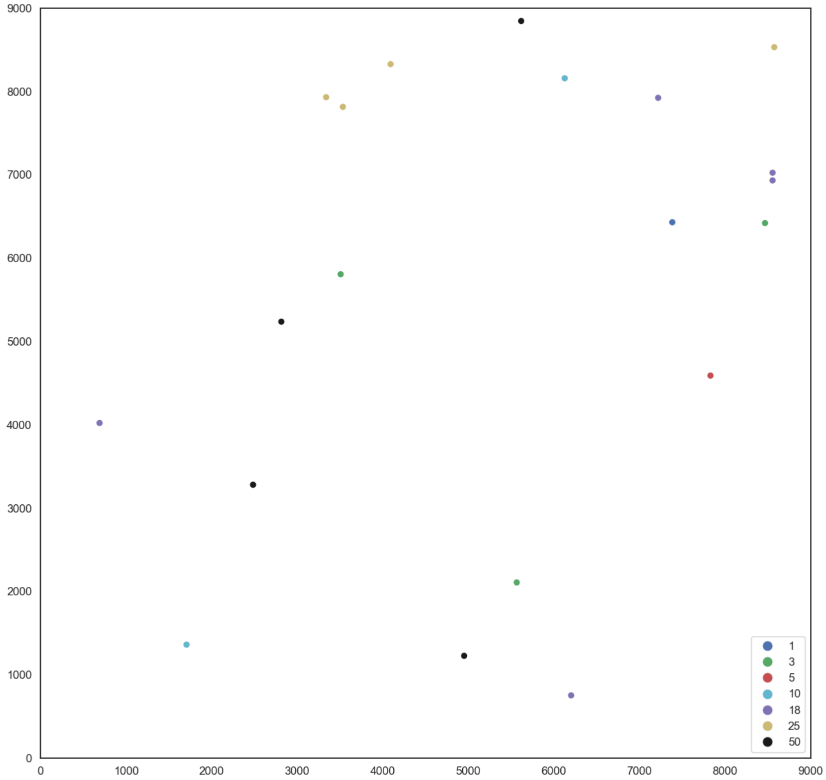

Figure 2: Locations in the microprocessor of each transistor in my analysis, color coded by the number of latent states that optimally explains its spiking under the HMM.

among the twenty (located in the second cluster) are best modeled by the same number of states (18). For these observations to support my hypothesis, one must suppose that transistors that are near each other share a functional role. My understanding of the MOS 6502 is that this assumption is plausible, sometimes. There does exist some functional modularity in this and other microprocessors. Although, there is quite a bit less of that in the chip than there is (believed to be) in the brain.

This test of my hypothesis is neither as precise nor as rigorous as I would like it to be. But I think that my results are at least suggestive. Based on Eric Jonas's publicly available data, it should be possible to determine the functional role(s) played by each of the 3510 transistors. This would allow for a conclusive test of my hypothesis, or any other that someone would like to form about the character of a transistor in the chip's information processing architecture. However, I believe that this determination will need to be done by hand and will be quite time consuming. I did not have time to attempt this.

5 Next Steps

Because of the computational expense of training models on this dataset, I was only able to train on 20 transistors. I would like to train on a far larger subset of the 3510 than this. I would also like to try other models for the observations and other state space models. Since the spikes are count data, a Poisson observation model seems natural, and it would be fun to try Switching Linear Dynamical System and Recurrent Switching Linear Dynamical System models in place of the HMM and ARHMMs. Most importantly, I need a better way to evaluate my success at picking out the functional role of transistors than examining their proximity to other transistors.

6 References

[1] Eric Jonas and Konrad Paul Kording. “Could a Neuroscientist Understand a Microprocessor?” en. In: *PLOS Computational Biology* 13.1 (Jan. 2017), e1005268. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1005268. URL: https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005268 (visited on 03/20/2021).

[2] John C. Marshall. “Minds, Machines and Metaphors”. In: *Social Studies of Science* 7.4 (1977), pp. 475–488. ISSN: 0306-3127. URL: https://www.jstor.org/stable/284716 (visited on 03/20/2021).

[3] L. Paninski and J. P. Cunningham. “Neural data science: accelerating the experiment-analysis-theory cycle in large-scale neuroscience”. eng. In: *Current Opinion in Neurobiology* 50 (June 2018), pp. 232–241. ISSN: 1873-6882. DOI: 10.1016/j.conb.2018.04.007.

7 Appendix

Optimal # latent states	Transistor #	Log-likelihood train	Log-likelihood test
1	3146	1,056,454.92	1,056,454.92
	3328	-67,536.12	-67,536.12
3	2893	-67,536.12	-67,536.12
	3498	-67,536.12	-67,536.12
8	2294	885,064.86	784,372.81
10	2407	890,194.51	781,425.96
18	860	730,746.38	542,194.67
	2314	889,760.89	775,834.48
	2924	1,048,389.46	978,453.32
	3081	844,524.72	795,896.47
25	223	743,446.98	623,155.61
	435	801,380.28	708,613.60
	507	717,085.10	638,256.08
	752	840,363.48	735,093.04
	2190	912,065.72	868,403.45
	2609	899,924.79	846,600.66
	3108	845,462.07	753,450.00
	880	640,927.00	557,651.71
50	3488	1,056,459.41	-412,305,394,479.99

Table 2: Summary of ARHMM models.

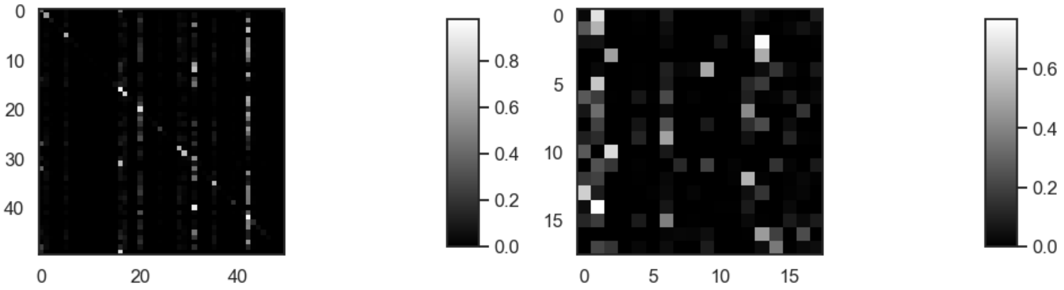

Figure 3: Inferred transition matrix for transistor 2924 (left) and transistor 3109 (right).

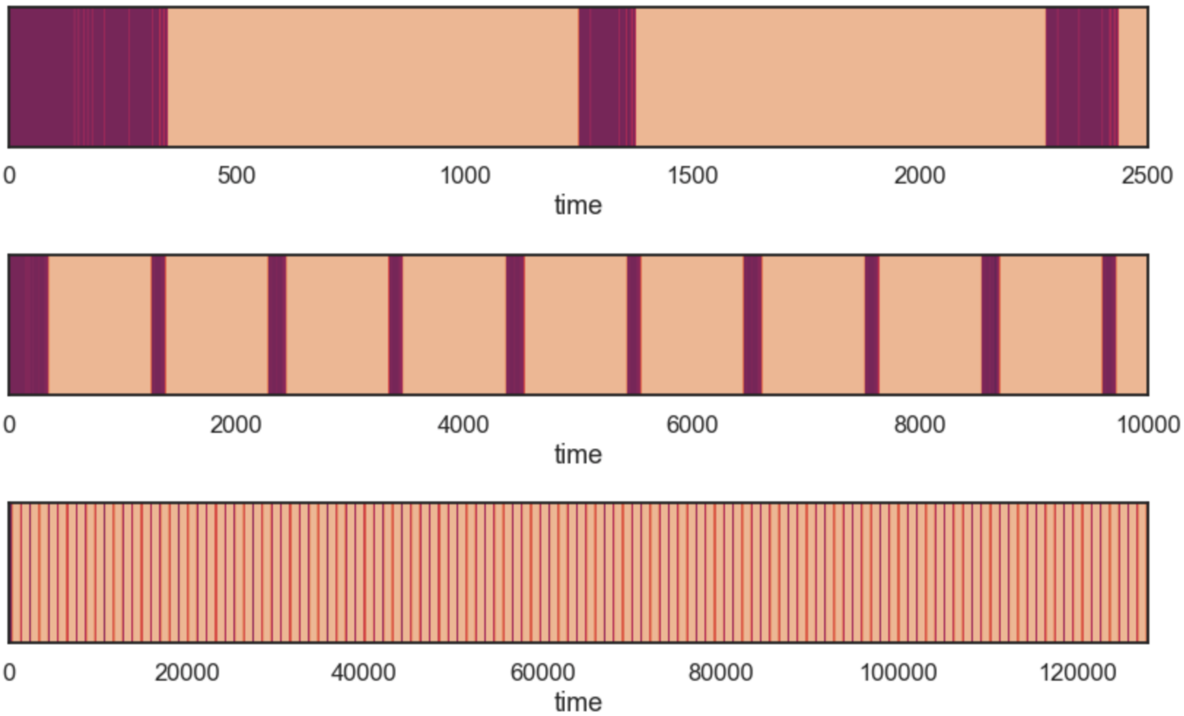

Figure 4: Inferred latent states for transistor 2924 for three time windows.

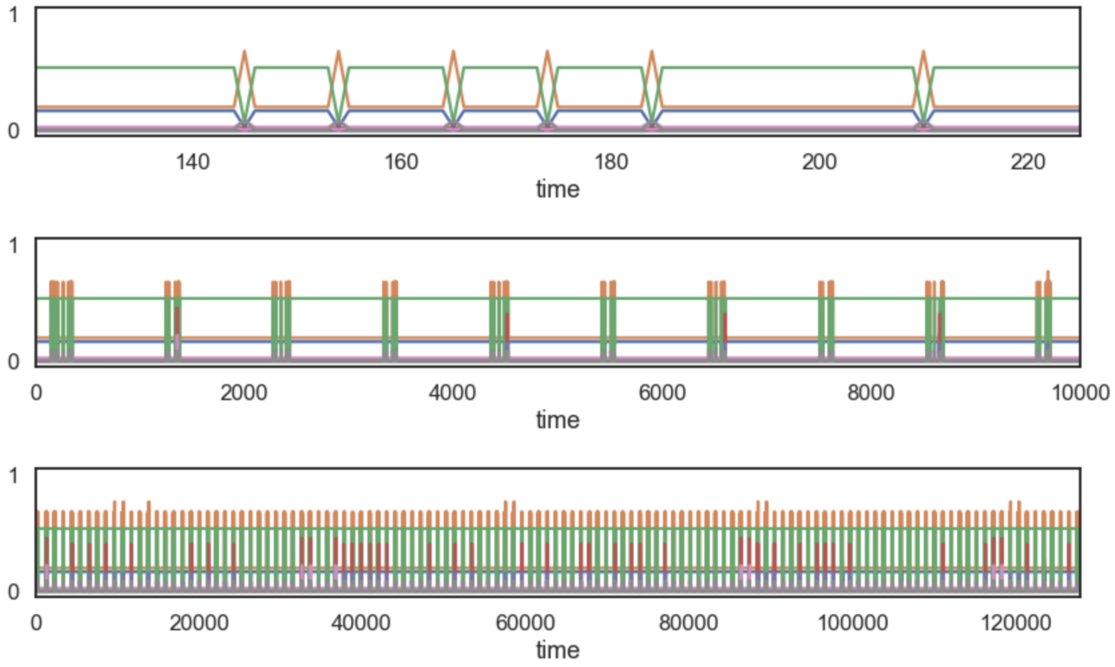

Figure 5: Posterior on latent states for transistor 3109 for three time windows.

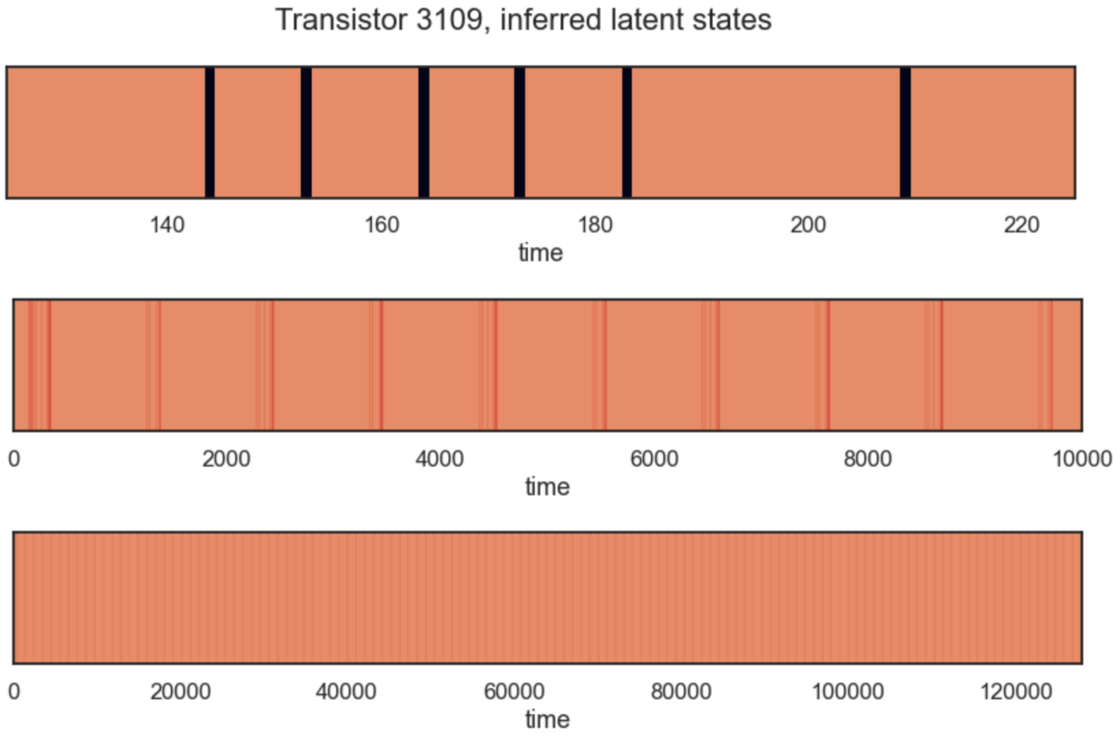

Figure 6: Inferred latent states for transistor 3109 for three time windows.

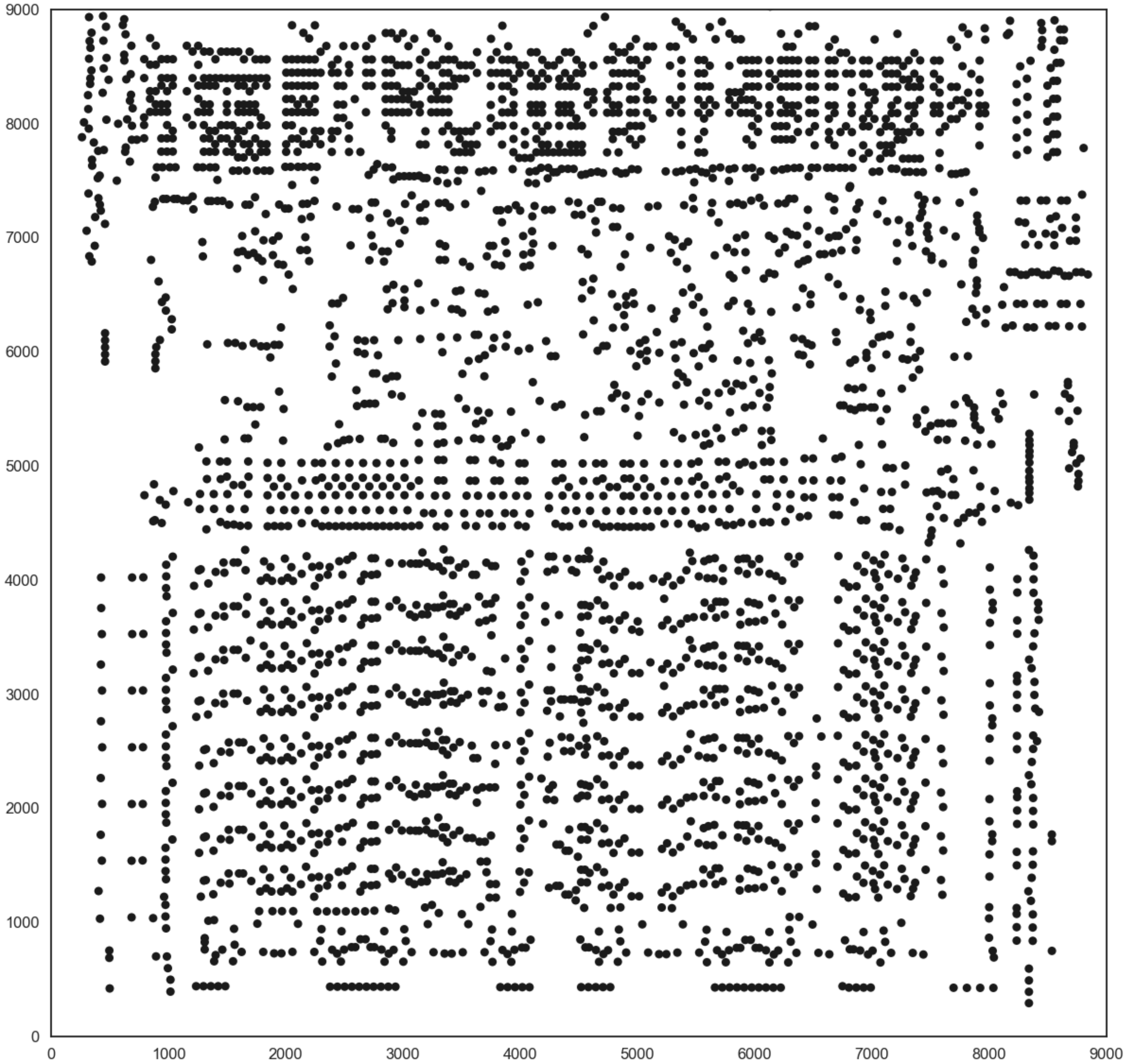

Figure 7: Locations of all 3510 transistors in the MOS 6502.