

Modeling Complex Random Memristor Networks

Daniel Teal
Mech. Eng. & Math, UT Austin

Dr. Tomonobu Nakayama
Nano Functionality Integration Group at MANA, NIMS



Novel Electronics

Computers aren't getting much faster.
(2D photolithographic silicon/CMOS)



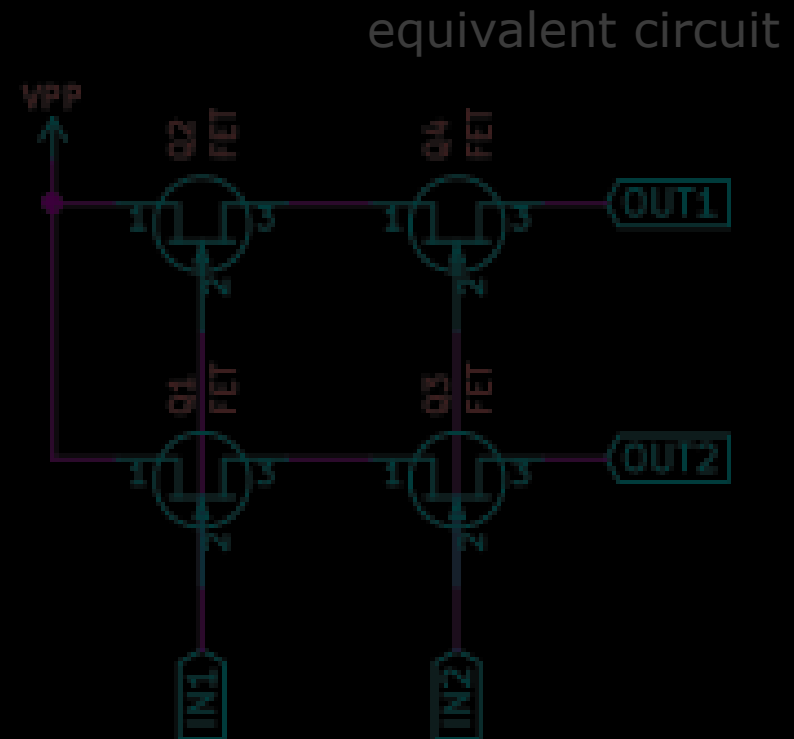
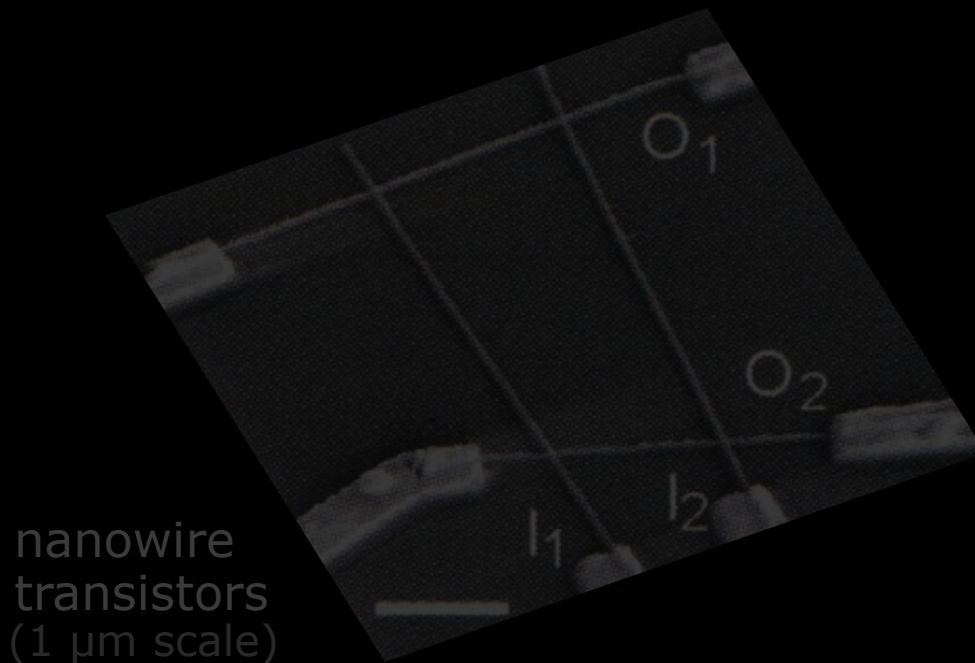
search for new computing methods

- 3D circuitry
- quantum computing
- programmable hardware (FPGA)
- photonic computing
- neuromorphic computing
(nanowire electronics)

Nanowires for Electronics

Nanowire (NW): 1D cylinder of Ag, Si, Au...

- NW-NW junction = circuit element



PVP@Ag Nanowire Networks

collection of Ag nanowires with PVP coating

- model properties, understand potential
- similar to neural network?

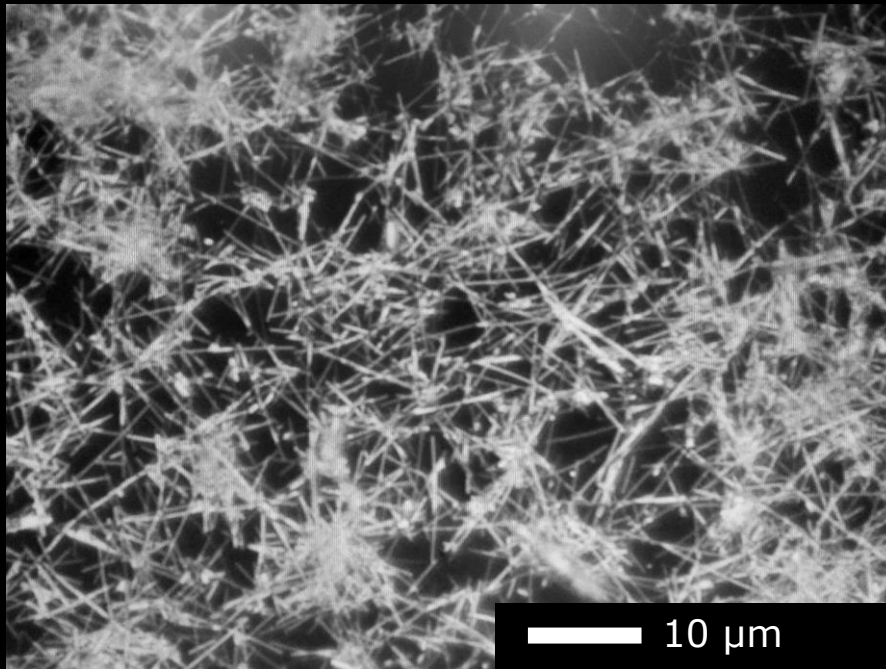
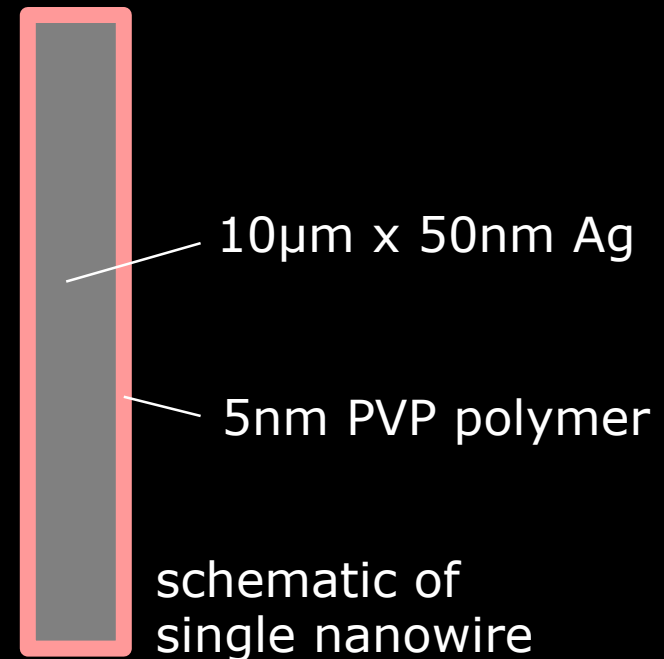
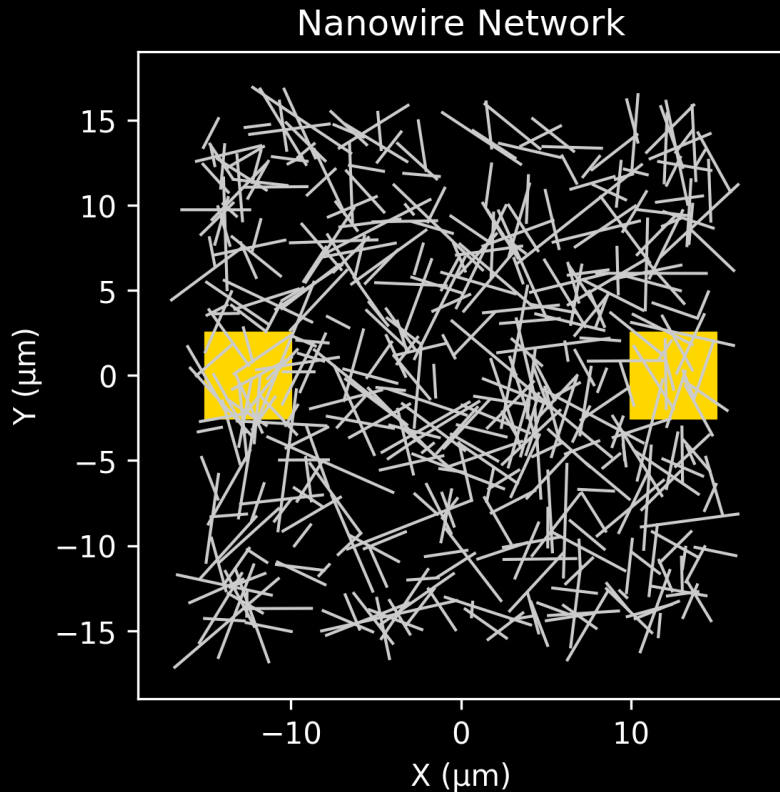


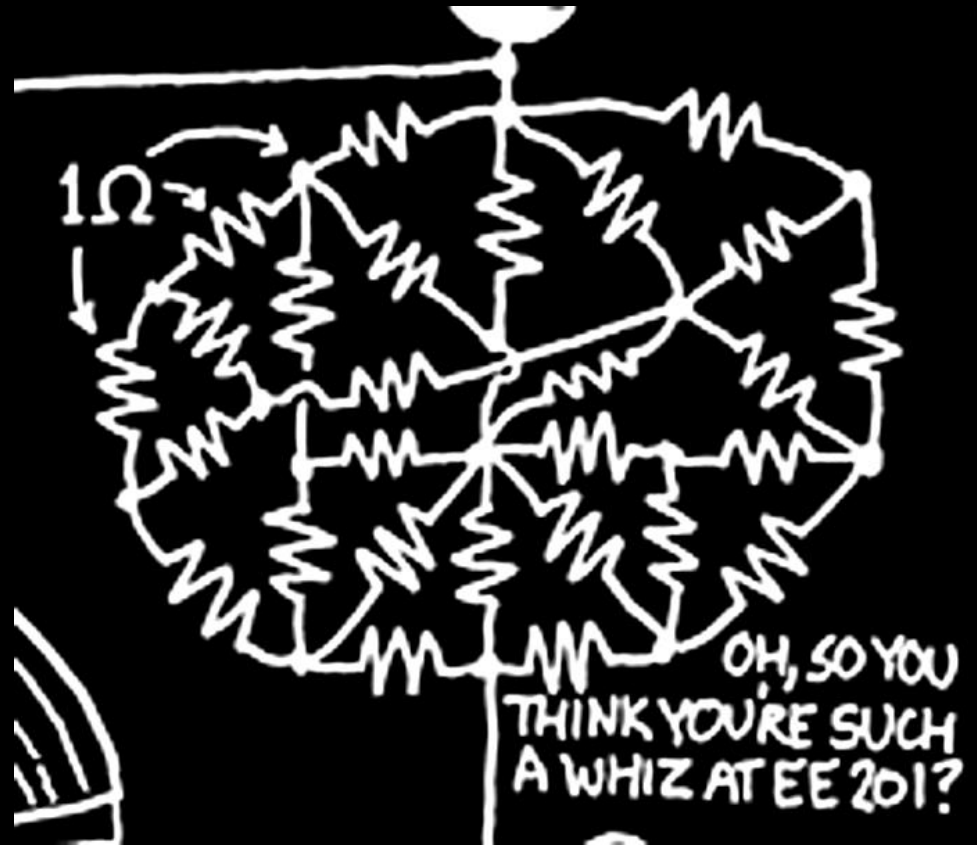
image of a PVP@Ag network



Step 1: Create Simulation



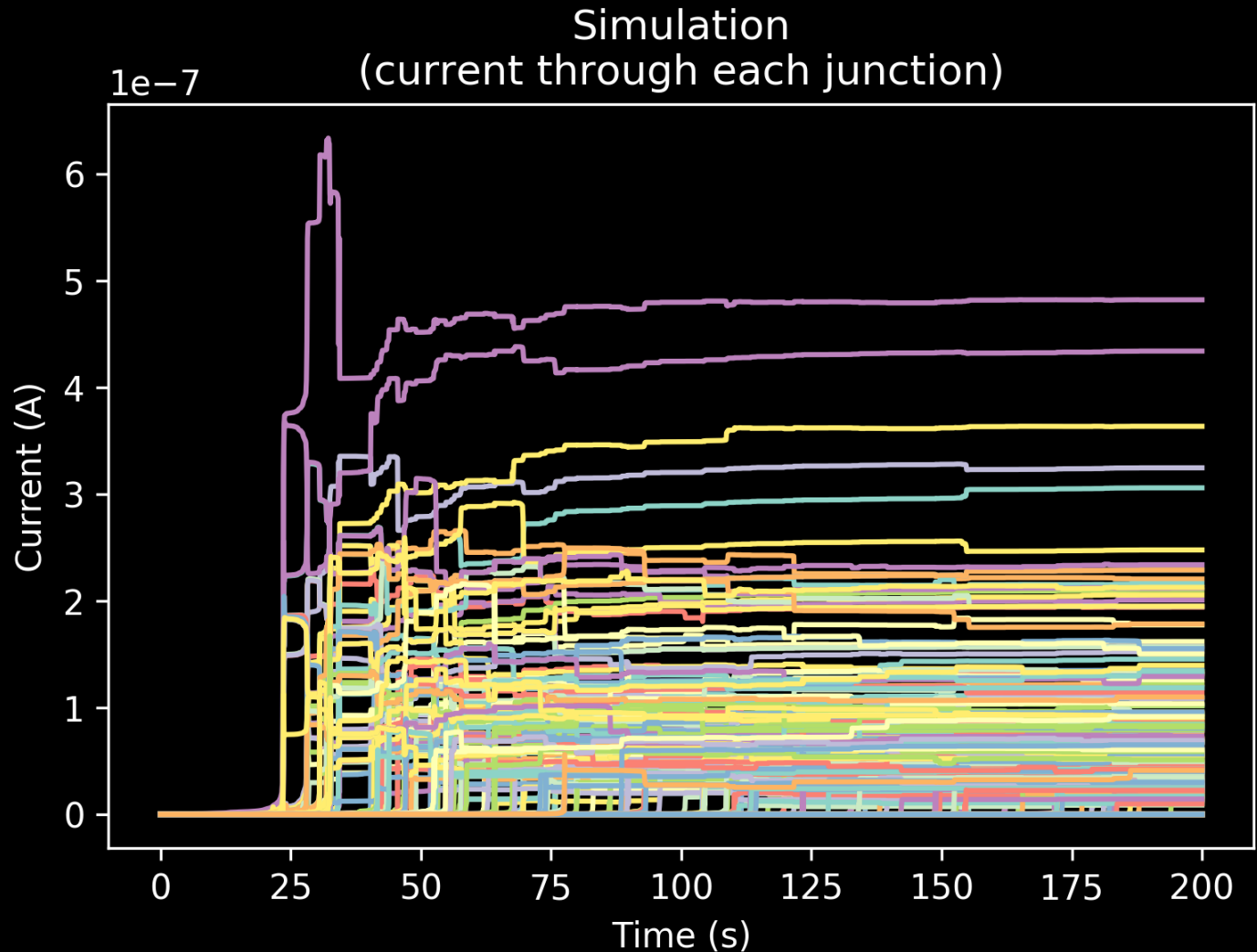
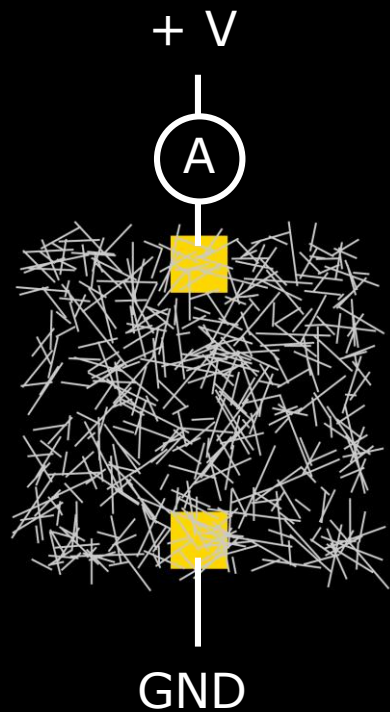
a complete simulated network



equivalent circuit

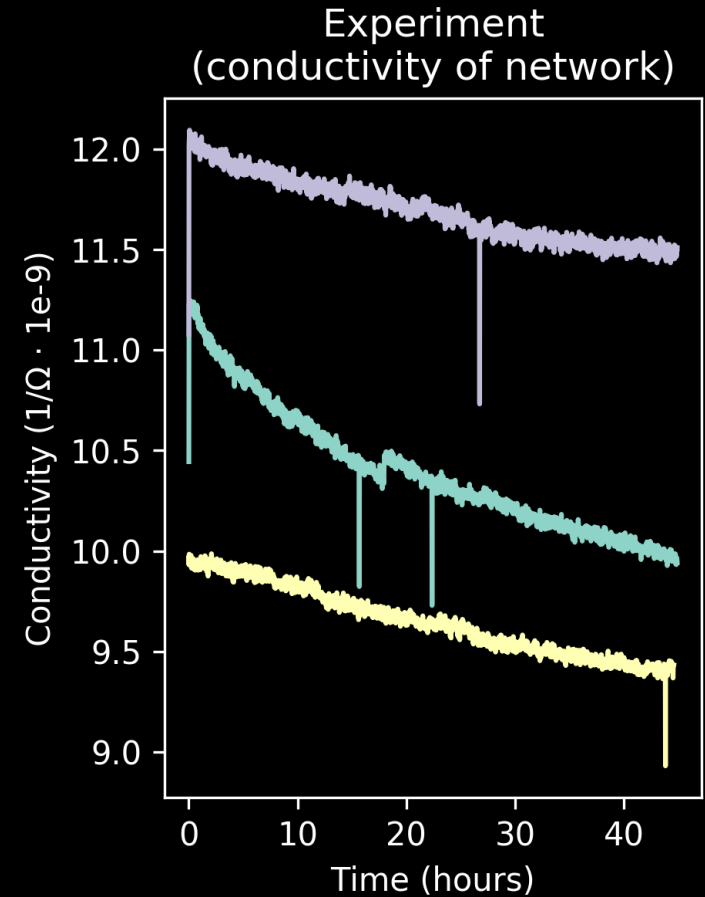
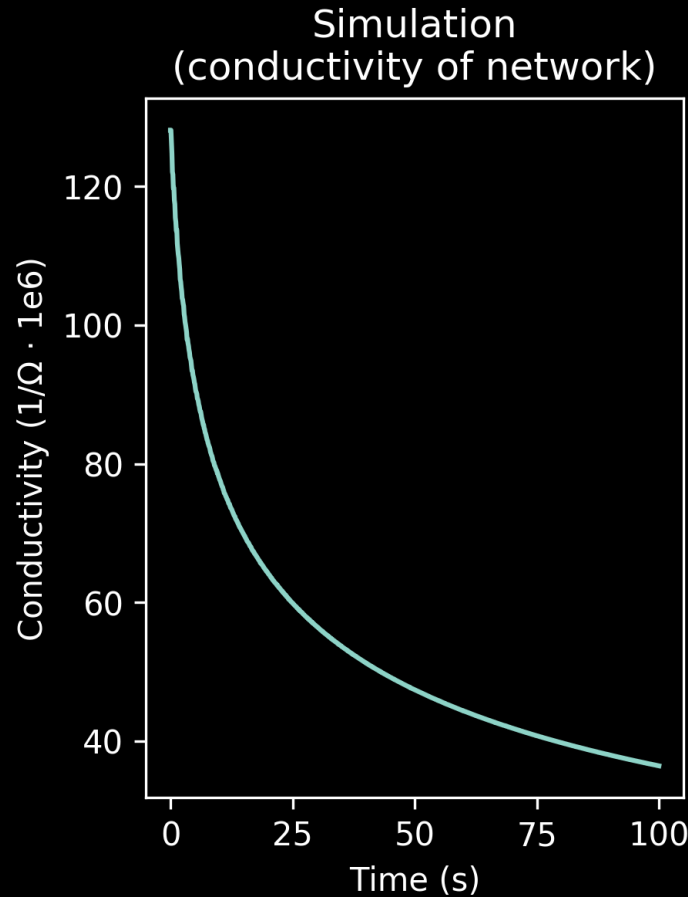
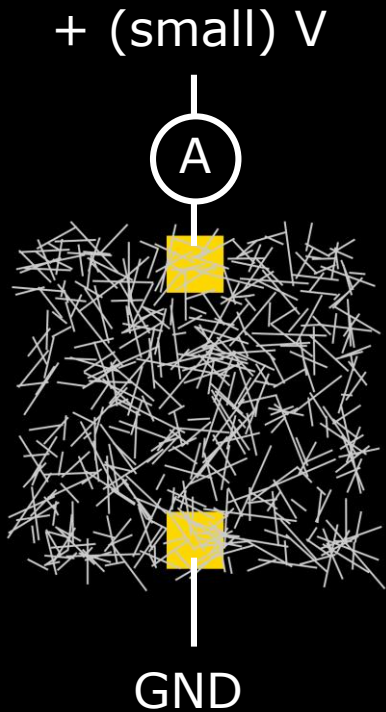
Munroe, R. Circuit Diagram. XKCD 730.

Step 2: Response to Constant Voltage



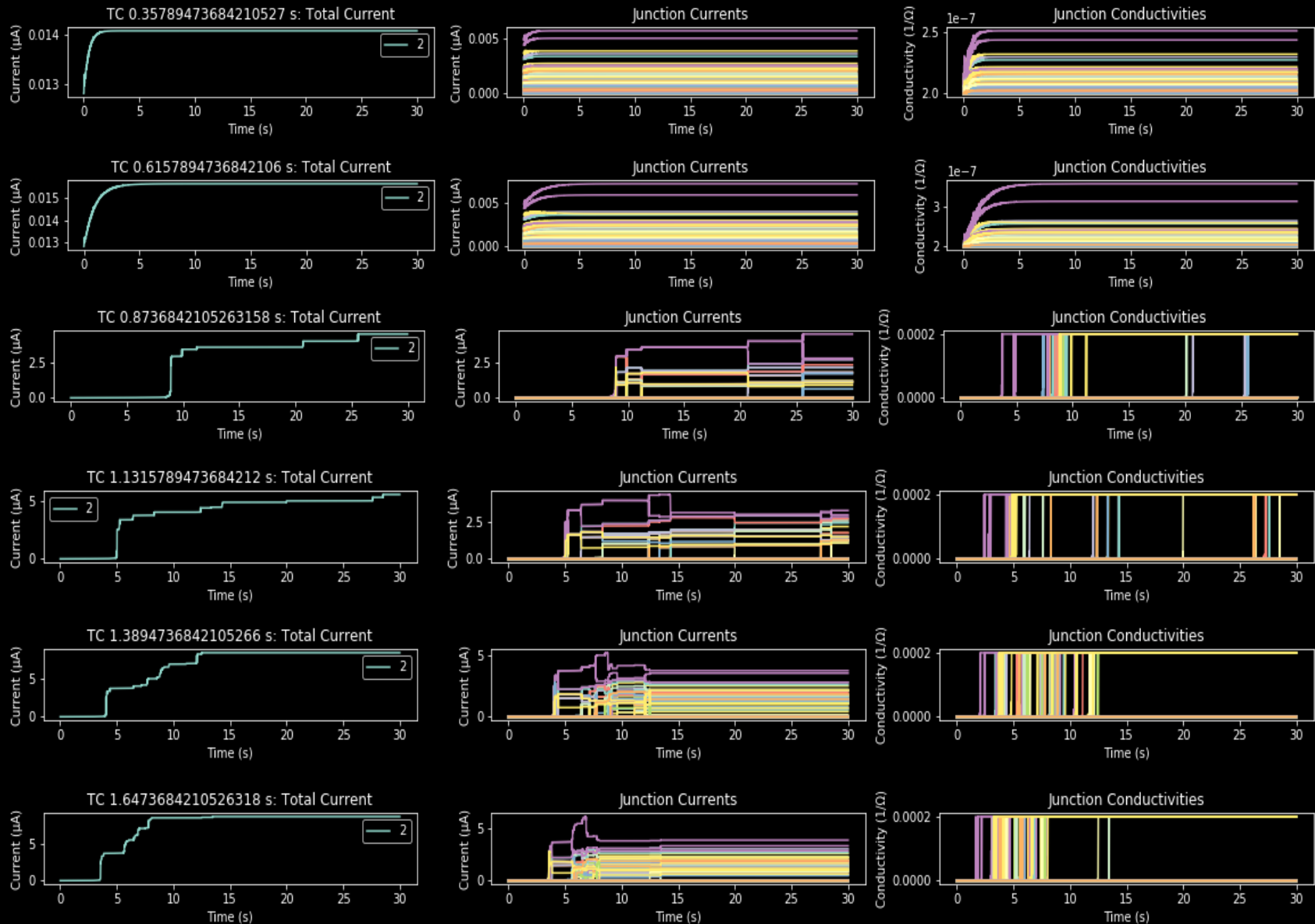
relative current density over time

Step 3: Resistance Decay with Time



full decay is very slow- weeks!

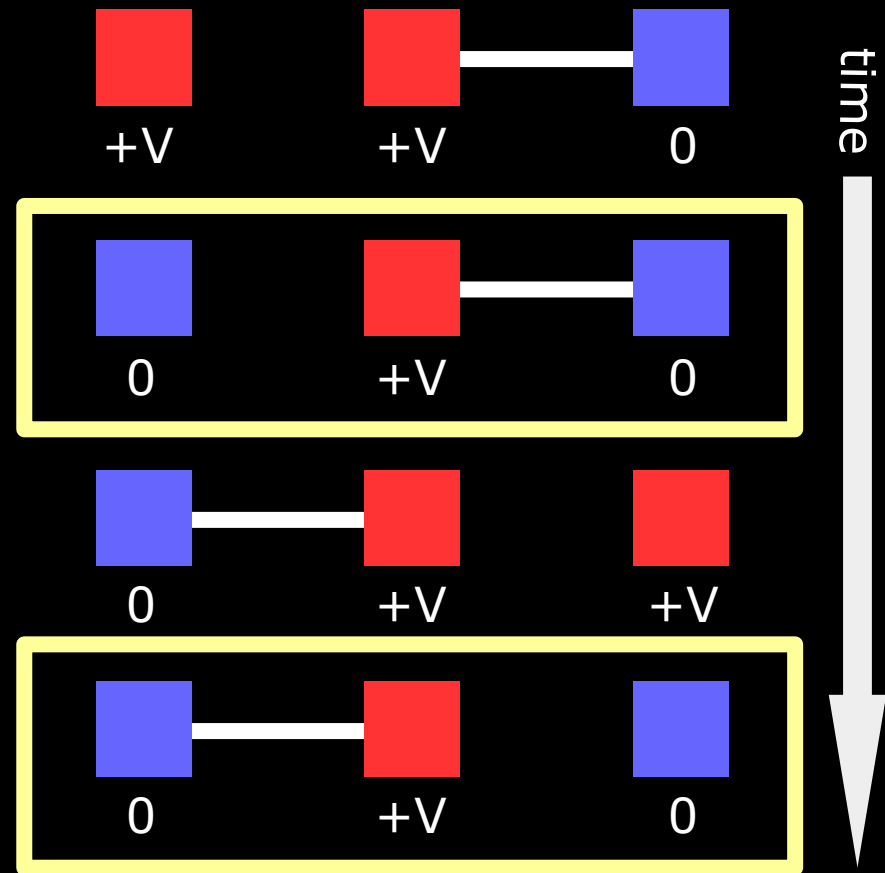
Parameter sweep found nothing else



One use: current switch



switch simulation



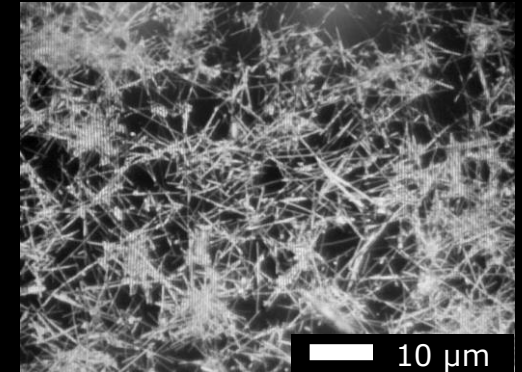
white indicates current flow

Conclusions

- Simulation predicts behavior
- Junction memristor hypothesis supported
- Network acts like planar memristor

Applications:

- Switch for a finite state machine
- Ephemeral memory: information that disappears after a week
- Custom non-CMOS electronics
- Planar memristance



Future work:

- Attempt to control memristor parameters (decay, resistance)
- Add capacitance for higher-order behavior

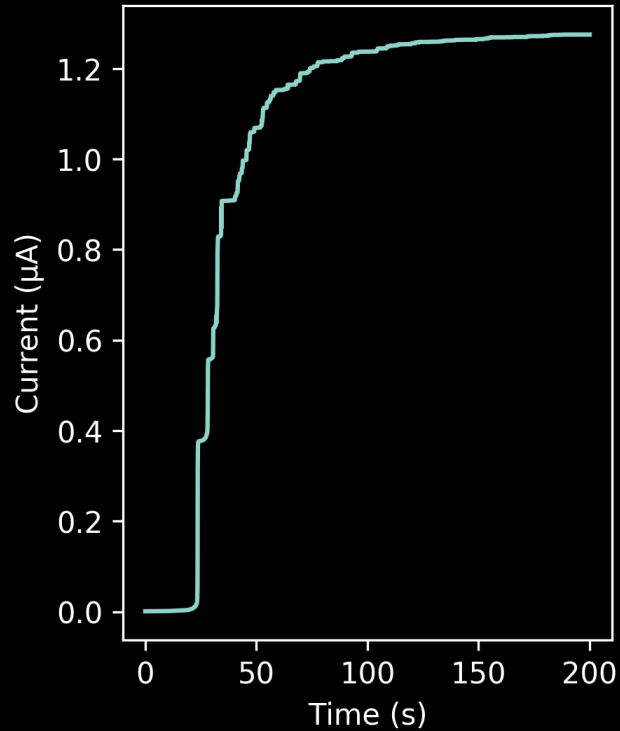
Acknowledgements

- NIMS and MANA
- National Science Foundation
 - NSF grant no. OISE-1559368
- National Nanotechnology Coordinated Infrastructure
- PI: Dr. Tomonobu Nakayama
- Nano Functionality Integration Group: Qiao Li, Adrian, Sasaya...
- Dr. Lynn Rathbun and other program coordinators

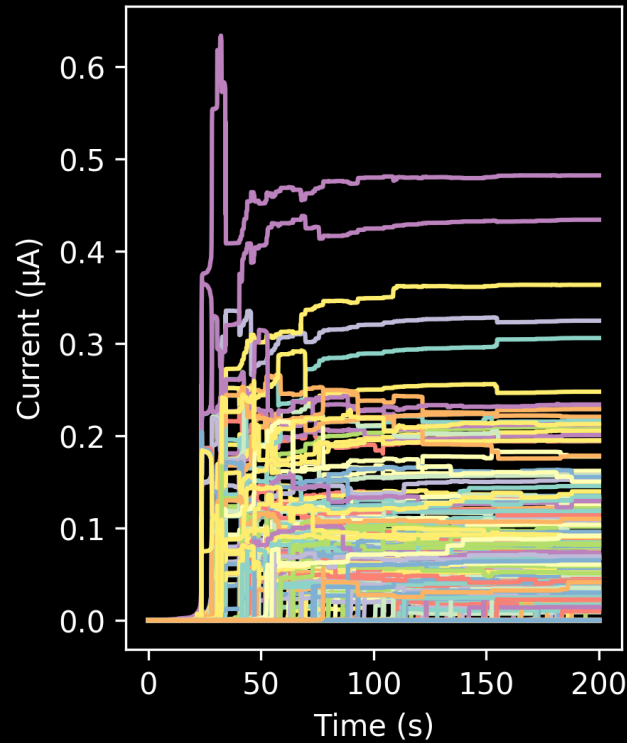


Junction Behavior

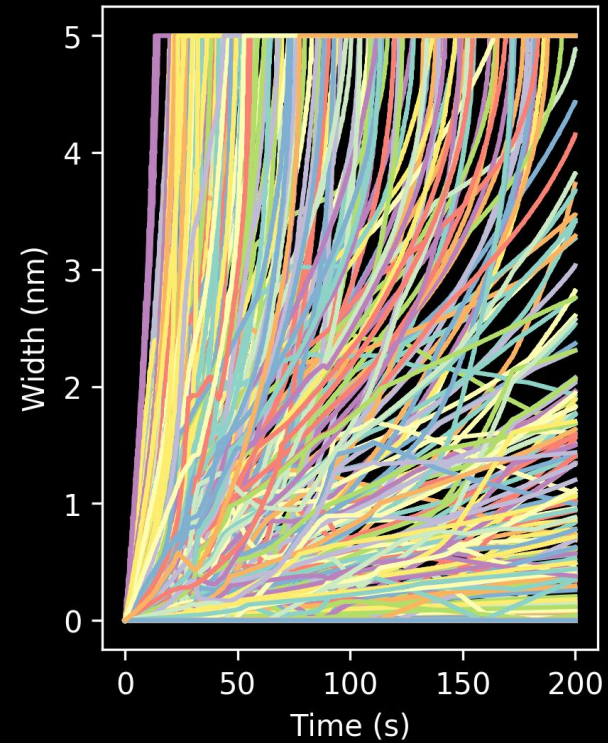
Simulation
(current through network)



Individual Junction Currents



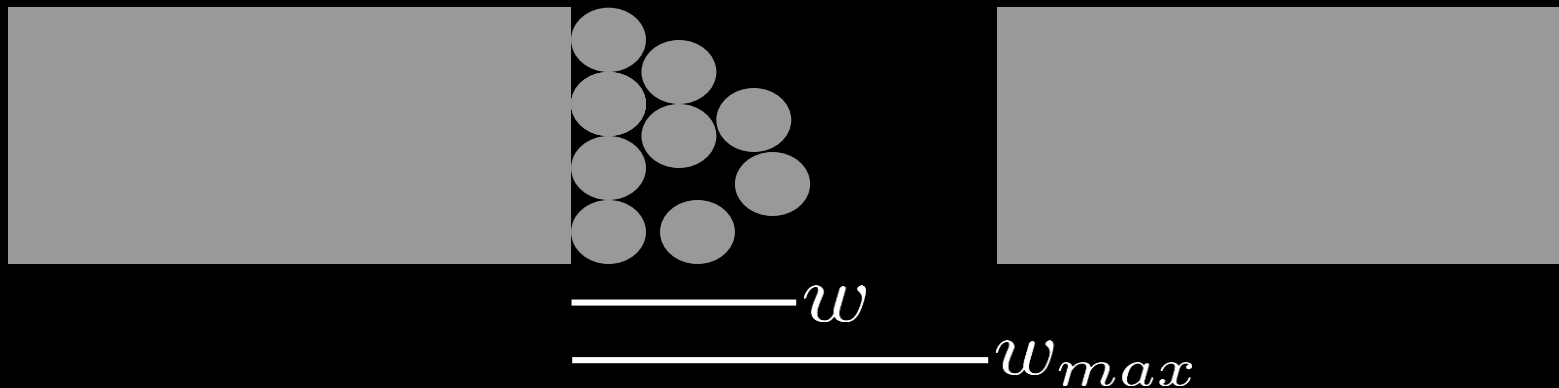
Individual Junction Widths



NW-NW Junction Electrical Model

$$V = \left[\frac{w}{w_{max}} \cdot R_{on} + \left(1 - \frac{w}{w_{max}} \right) \cdot R_{off} \right] \cdot I$$

$$\frac{dw}{dt} = \mu_v \cdot \frac{R_{on}}{w_{max}} \cdot |I| - \frac{w}{\tau}$$

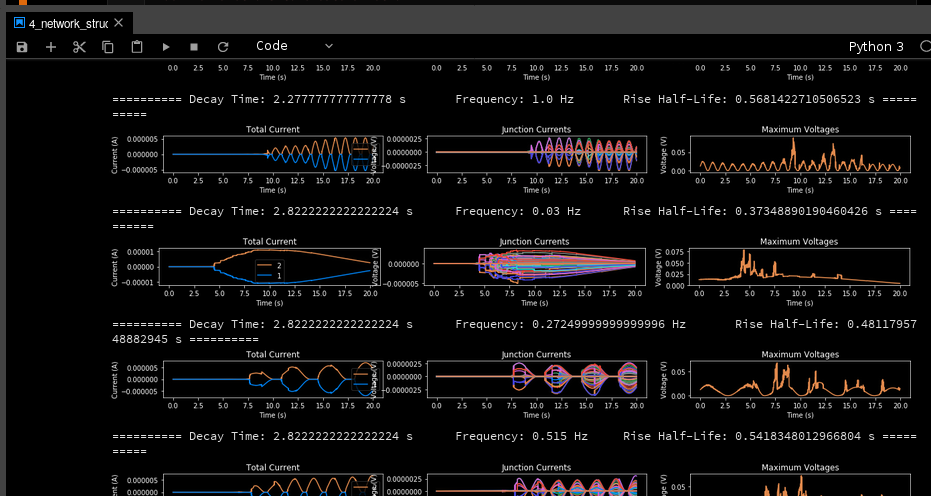


Programming Environment

```

File Edit View Run Kernel Tabs Settings Help
nanowire.py
21 create a new nanowire.
22
23 Keyword arguments:
24 start, end -- endpoint coordinates in um as 2 element numpy array
25 end -- same
26 ***
27 return {'start':start, 'end':end, # endpoint coordinates in um
28        'min':(min(start[0], end[0]), # for bounding box checks
29              min(start[1], end[1])),
30        'max':(max(start[0], end[0]),
31              max(start[1], end[1])),
32        'electrodes':[], 'junctions':[]} # indices of connected items
33
34 def new_junction(position=(0, 0), start=0, end=0):
35     """Create a new junction.
36
37     Keyword arguments:
38     position -- coordinates in um as 2-tuple
39     start, end -- nanowire indices
40     ***
41     return {'position':position, # tuple that stores coordinates in um
42           'start':start, 'end':end, # nanowire indices
43           'nanowires':[start, end]} # for connectivity algorithm
44
45 def new_electrode(x_pos=0, y_pos=0, width=0, height=0):
46     """Create a new electrode.***
47     return {'x':x_pos, 'y':y_pos, 'w':width, 'h':height}
48
49 # define functions to create a network

```



NETWORK CONNECTIVITY

Next, we will determine properties of the network based on the above estimated parameters.

According to said parameters, the nanowires occupy a square area about 3000 μm wide on a side. This would contain about two million nanowires. Because this would be difficult to simulate in a reasonable timeframe, consider instead a 30 μm wide square area with a total of about 200 nanowires.

Scaling Behaviors

As shown below, as the number of nanowires in a given area increases, the total number of junctions increases quadratically while the number of junctions per nanowire increases linearly.

This is straightforward in theory: consider any two nanowires. Because they are independently, randomly, and uniformly positioned, there is some constant (albeit hard to calculate) (proportional to the nanowire length squared) probability they will touch to form a junction. Thus, the number of junctions should be proportional to the number of pairs of nanowires, which, for a number of nanowires N , is

$$\frac{N^2 - N}{2}.$$

For large N , $N^2 \gg N$, so this means the number of junctions J has, approximately, $J \propto N^2$.

Divide the total number of junctions by the number of nanowires to find that the number of junctions per nanowire, J_n , has about $J_n \propto N$.

```

In [22]: # assume some experiment constants
e_size = 5 # um, electrode size
area_size = 30; area = area_size**2 # um

# create arrays to hold results
nw_density = []; j_density = []; j_per_n = []; percent_nw_connected = []; not_connected = []

# iterate through nanowire densities
min_density = 0.01; max_density = 0.4; density_step = .001
current_density = min_density
bar = FloatProgress(min=min_density, max=max_density); display(bar) # progress bar!
while current_density < max_density:

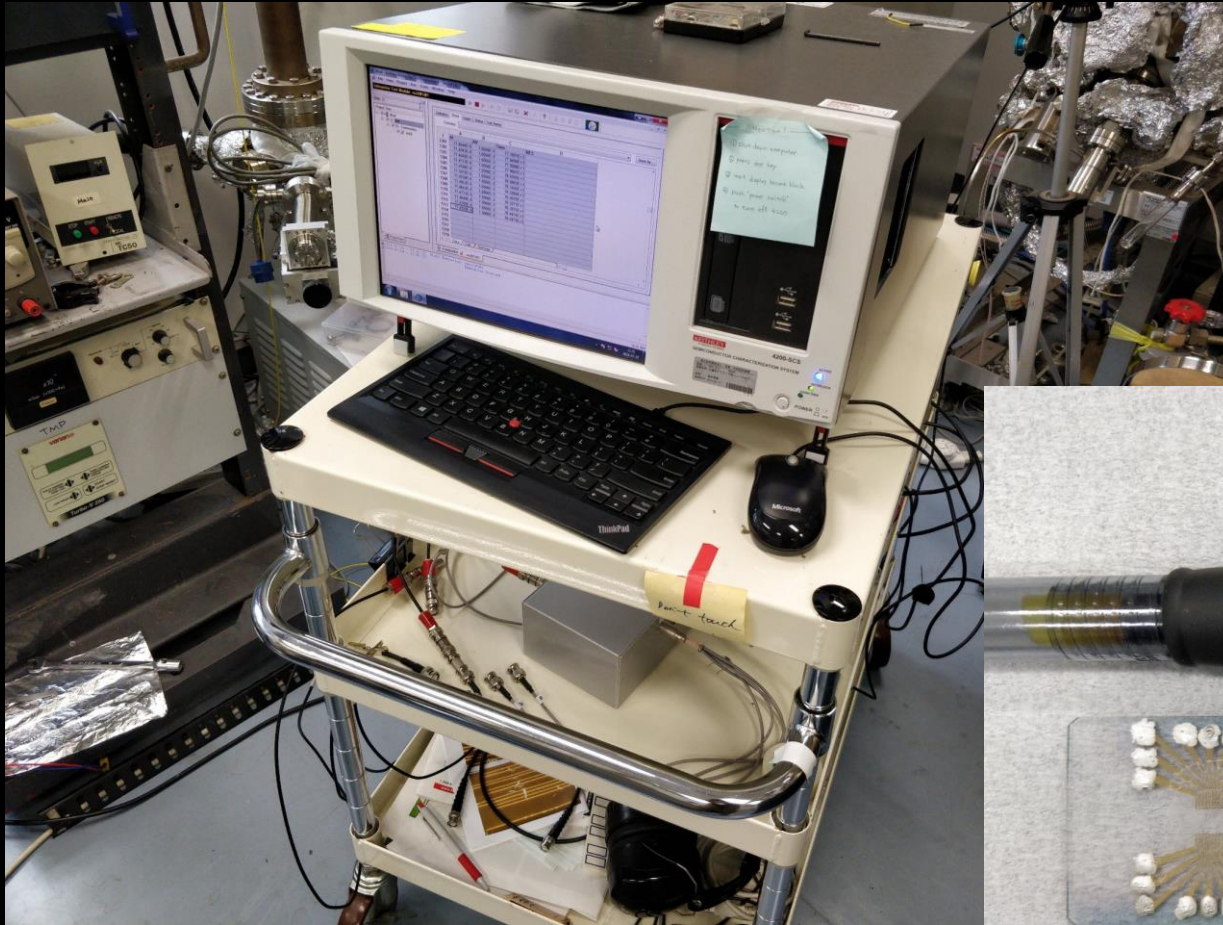
    # create a random network with the given nanowire density
    network = create_network(params={
        'width':area_size, 'height':area_size,
        'inputs':1, 'outputs':1,
        'e_width':e_size, 'e_height':e_size,
        'density':current_density,
        'lalpha':5.77, 'lloc':-0.65, 'lbeta':0.87,
        'seed':random.randint(0,1e6)
    })

    # store values
    nw_density.append(current_density)
    j_density.append(len(network['junctions'])/area_size)
    j_per_n.append(np.mean([len(n['junctions']) for n in network['nanowires']]))
    percent_nw_connected.append((len([n for n in network['nanowires'] if n['connected']])/len(network['nanowires'])))

```



Test Apparatus



Keithley SCS 4200

Test devices

