CS343S Project Proposal

Tejas Narayanan (`tejasn9`)

May 2, 2024

1 Introduction

Circuit designs are often created on paper or on specialized software. However, the initial designs of circuits are not always functional. For complex circuits that use hardware beyond simple resistors, wires, capacitors, and power sources, it is impossible to test and debug circuits on a simple simulator. This means that one must physically build the circuit in order to find out that it does not work, wasting time and potentially money/parts.

In this proposal, I will outline "pyvolt", a domain-specific language for circuit design specification. It will enable circuit designers to define custom components (e.g., Arduinos, phototransistors, etc.) that are often absent from circuit simulators, and connect them via standard components. With the programmed custom components, users will be able to test their circuits using the DSL to determine whether or not it works correctly before physically building the circuit.

2 Language design

Pyvolt has two primary classes: `Circuit` and `Component`. Each `Component` is associated with a `Circuit` (specified by `in circuit_name` when creating a component) and connects to two or more `Nodes` of the circuit. `Nodes` are created implicitly by specifying connections between components through the syntax `comp1 >> comp2`.

Pyvolt has a number of pre-created `Component`s, including `Resistor`, `VoltageSource`, and `Diode`.

Pyvolt `Circuit`s can be inspected at any of their `Nodes` for the node's voltage or current.

Here is an example program of a circuit powering an LED:

```
1  import pyvolt as pv
2  from pyvolt import components as comp
3
```

```
4  circuit = pv.Circuit()
5
6  # define the components of the circuit
7  voltage_source = comp.VoltageSource(v=5) in circuit
8  resistor = comp.Resistor(ohm=150) in circuit
9  diode = comp.Diode(v_f=2, i_f=20e-3) in circuit
10
11 # define the connections between components
12 voltage_source.vplus >> resistor.n1
13 resistor.n2 >> diode.anode
14 diode.cathode >> voltage_source.vminus
15
16 # set the circuit reference voltage
17 circuit.gnd >> voltage_source.vminus
18
19 # compile the circuit
20 circuit.compile()
21
22 # probe the circuit
23 print("V before diode:", circuit.inspect_voltage(diode.anode))
24 print("V after diode:", circuit.inspect_voltage(diode.cathode))
```

Pyvolt also supports creating custom Components by creating a class that inherits from the Component class. Here is the same LED example, but with an Arduino instead of a voltage source:

```
1  import pyvolt as pv
2  from pyvolt import components as comp
3
4  class Arduino(pv.Component):
5      def __init__(self, n_pins=5):
6          self.n_pins = n_pins
7          self.pin_voltages = [0 for _ in range(n_pins)]
8          self.pin_connections = [None for _ in range(n_pins)]
9
10         self.gnd_connection = None
11
12     def pin(self, pin):
13         return self.pin_connections[pin]
14
15     def gnd(self):
16         return self.gnd_connection
17
18     def pin_on(self, pin):
19         self.pin_voltages[pin] = 5
20
21     def pin_off(self, pin):
22         self.pin_voltages[pin] = 0
23
24 circuit = pv.Circuit()
25
26 # define the components of the circuit
27 arduino = Arduino() in circuit
28 resistor = comp.Resistor(ohm=150) in circuit
29 diode = comp.Diode(v_f=2, i_f=20e-3) in circuit
30
31 # define the connections between components
```

```
32  arduino.pin(0) >> resistor.n1
33  resistor.n2 >> diode.anode
34  diode.cathode >> arduino.gnd()
35
36  # set the circuit reference voltage
37  circuit.gnd >> ardunio.gnd()
38
39  # CASE 1: pin 0 is on
40  arduino.pin_on(0)
41  circuit.compile()
42  print("V before diode:", circuit.inspect_voltage(diode.anode))
43  print("V after diode:", circuit.inspect_voltage(diode.cathode))
44
45  # CASE 2: pin 0 is off
46  arduino.pin_off(0)
47  circuit.compile()
48  print("V before diode:", circuit.inspect_voltage(diode.anode))
49  print("V after diode:", circuit.inspect_voltage(diode.cathode))
```

Finally, pyvolt can draw circuit diagrams using `circuit.draw()`:

```
1   import pyvolt as pv
2   from pyvolt import components as comp
3
4   circuit = pv.Circuit()
5
6   # define the components of the circuit
7   voltage_source = comp.VoltageSource(v=5) in circuit
8   resistor = comp.Resistor(ohm=150) in circuit
9   diode = comp.Diode(v_f=2, i_f=20e-3) in circuit
10
11  # define the connections between components
12  voltage_source.vplus >> resistor.n1
13  resistor.n2 >> diode.anode
14  diode.cathode >> voltage_source.vminus
15
16  # set the circuit reference voltage
17  circuit.gnd >> voltage_source.vminus
18
19  # draw the circuit
20  circuit.draw()
```