



A GUI PROBLEM



DESIGNING AND IMPLEMENTING A GRAPHICAL USER INTERFACE THAT CREATES FINITE STATE MACHINES

AYLA SOLOMON ADVISOR: BILL DUNBAR

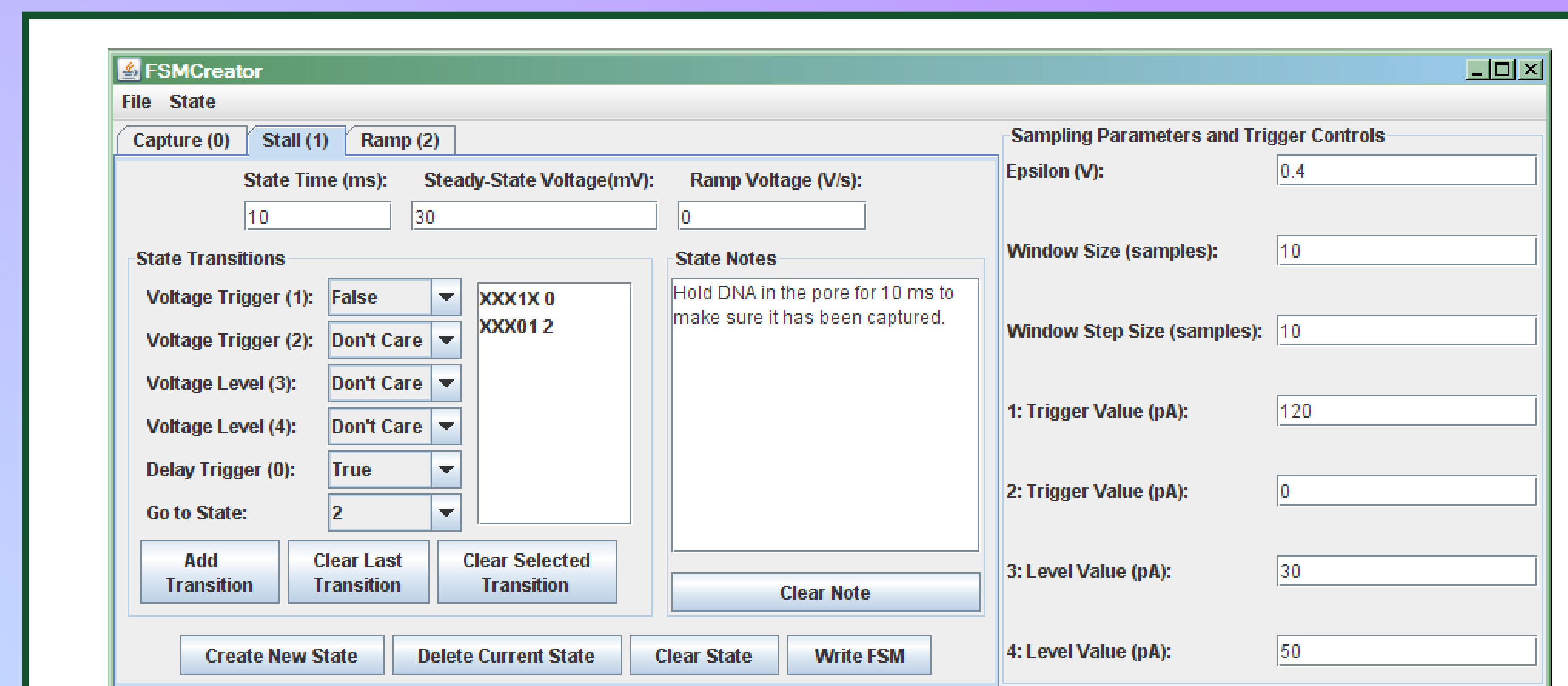
Motivation



The GUI seeks to simplify and expedite programming a finite state machine (FSM). An FSM is used to control capture and translocation of DNA through a biological nanopore. The FSM is used to program a field programmable gate array (FPGA) that can quickly detect voltage changes.

Design

Goal: Create a user interface that is simple, intuitive, robust, and extensible.

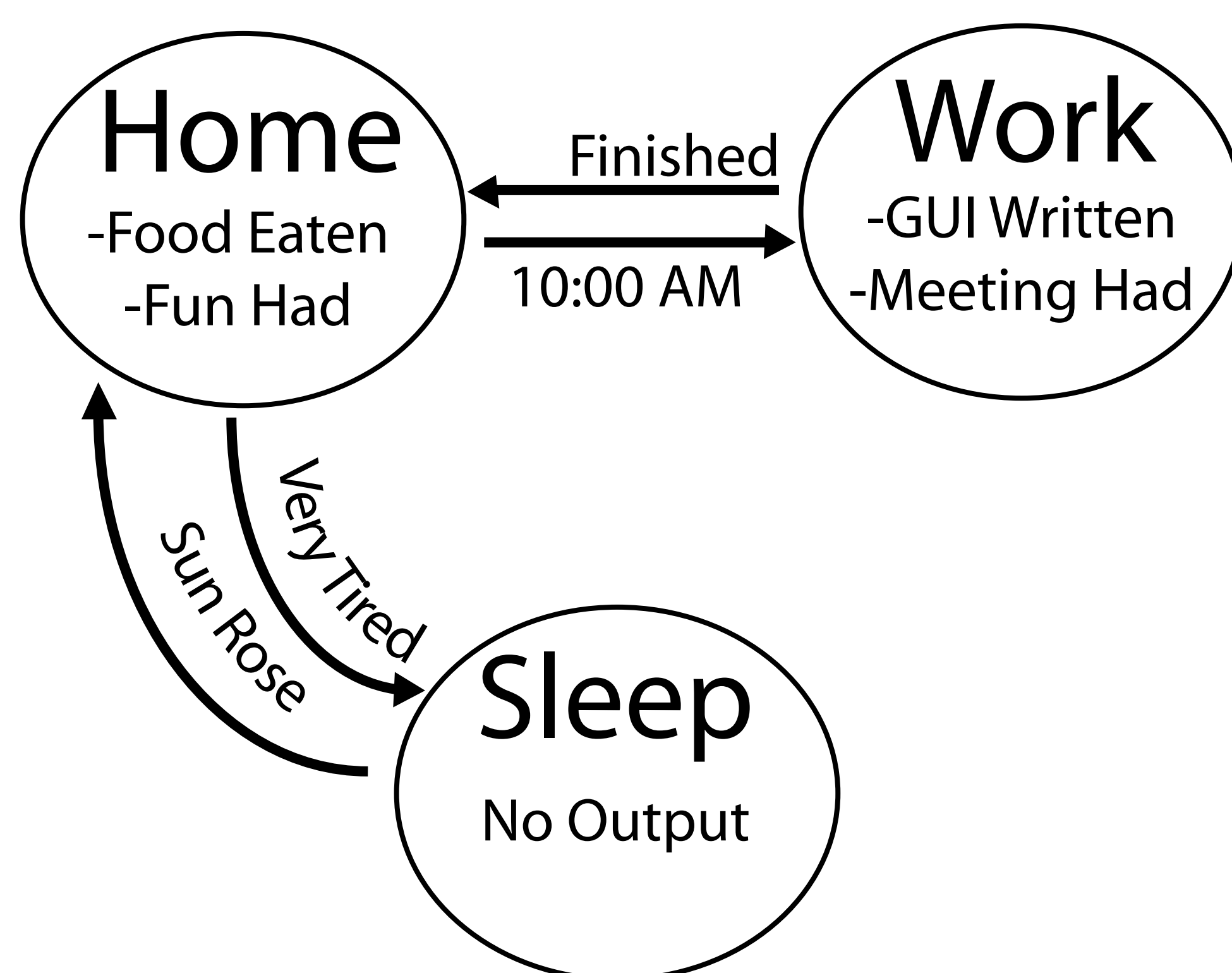


The GUI with a simple 3-state machine in the process of being built.

What is an FSM?

An FSM is a simple logic construct, algorithm, or program that models the behaviors of an object or system as a series of states. Each state has:

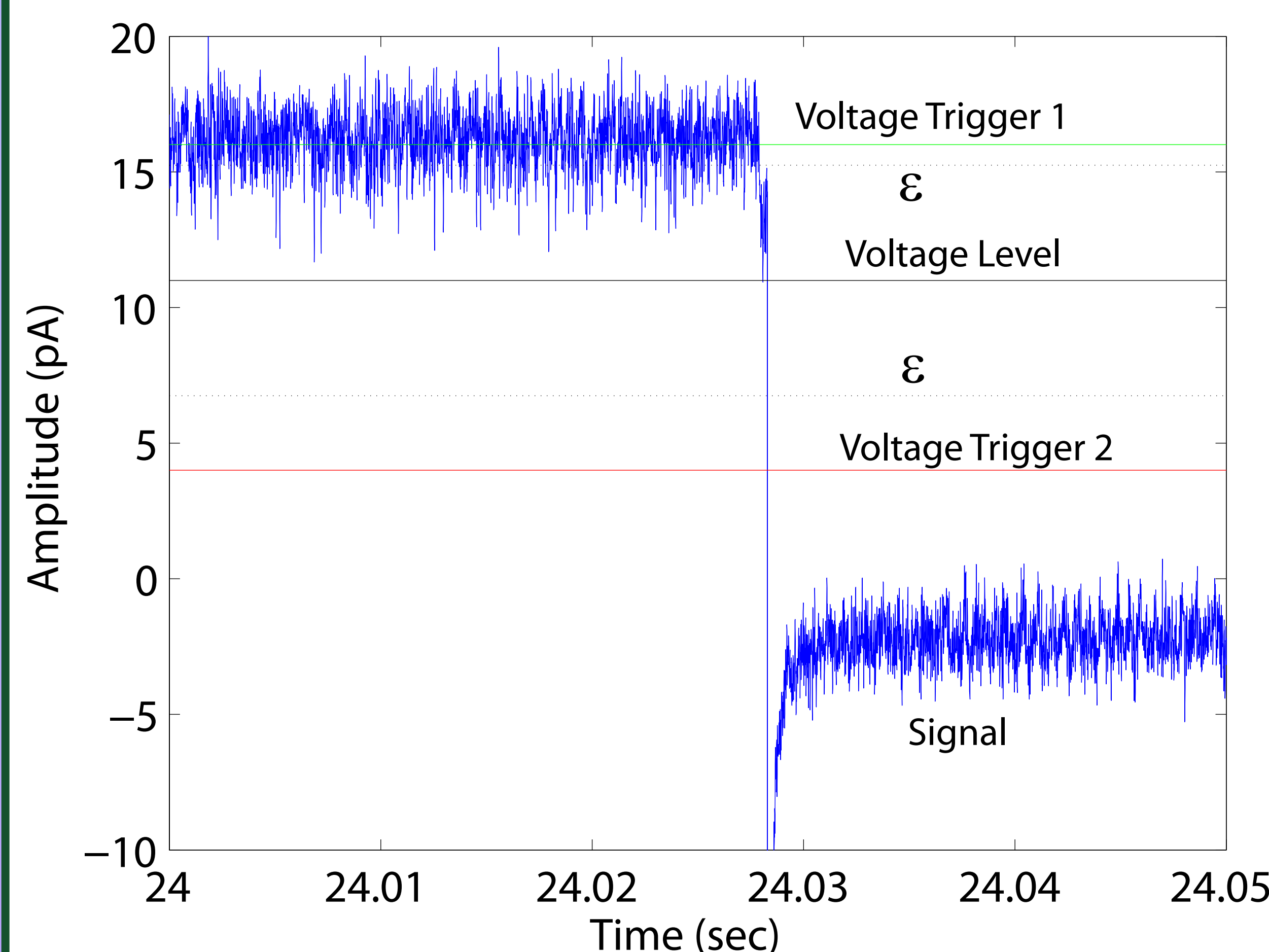
- Outputs
- Transitions to other states



A simplified example of an FSM: My typical day.

FSM Control of the Nanopore System

The FSMs used to control the nanopore system have two kinds of inputs: voltage levels, and voltage triggers. Voltage triggers detect when the signal crosses a set voltage. Voltage levels detect when the signal stays within the range (+/- ε) of a set voltage.



A signal from the nanopore system with example voltage triggers and voltage level.

Features

- Creates an FSM that is ready to export to the FPGA
- Disallows illegal values and actions
- Buttons and fields are self-explanatory
- No programming knowledge required to use the GUI
- Saves and opens unfinished FSMs
- Easy to extend

Improvements

Previously, text files had to be written that represented the FSMs. They were difficult for people without a programming background to write and understand. The GUI does incorporate some elements of the text files, but labels and limited fields make it easier to use.

```

begin0 10 180 0
      XX0X1 1
      1XXX0 2
end0
begin1 0 30 0 ...
  
```

A sample of a text file that would be written to create an FSM. These files required more compilation steps before they could be loaded onto the FPGA. Additionally, only a single grad student would write them.

Acknowledgements

Funding: The National Science Foundation Original FPGA conversion code and help: Noah Wilson