Proposed video structure

|  |  |
| --- | --- |
| **Section** | **Files** |
| Hello and welcome to the video walkthrough of Team 9’s digital systems group project – the glorified thermometer. In this demonstration, we will start with the overall structure of the design and then examine in detail some of the interfaces between components. | Audio\_intro  General Context Diagram  Some random graphics of rpi/basys/python? |
| The system consists of three hardware components: the raspberry pi zero (or RPI zero), the Basys3 Board, and a breadboard with an analog temperature sensor attached. In addition, a local web server is hosted on the RPI zero. The end user may interact with the Basys3’s onboard switches, make requests to the web server, and visually read the temperature data via the seven segment display. | Audio\_overview  General Context Diagram  Some snapshots of video? |
| Let’s take a closer look at the hardware components, beginning with the analog temperature sensor. The temperature sensor is wired to the analog PMOD connector (or the XADC) through a breadboard. The 3.3v and ground signals of the XADC are connected to the corresponding wires of the temperature sensor. Pins 7 through 10 are connected to ground, and pin 1 is connected to the difference between the 5 and 10K resistors in order to obtain the output. This is done to restrict the output to a maximum of 1 volt, which is the upper limit of the XADC. | Audio\_temperature\_sensor  Temp sensor stuff  Linear Temp Sensor schematic from onQ |
| Now it’s time to examine the basys3 board. There are three inputs and two outputs. The first input is the previously discussed analog signal from the temperature sensor. The second is a digital signal from the onboard switches, and the third is a digital signal from the general-purpose input/out (or GPIO) pins of the RPI zero.  Of the two outputs, one is a serial connection to the RPI zero and the other is a digital output to the onboard seven segment display.  The specifics of the serial connection to the Rpi zero as well as the GPIO connection to the digital PMOD port will be detailed later on. | Audio\_basys\_overview  basys schematic |
| The last of the three hardware components is the Raspberry Pi Zero. Its inputs consist of web requests made by the enduser to the locally hosted web server, and data from the serial connection to the Basys3 board. The outputs of the RPI zero consist of POST requests to the web server and a set of digital outputs to the Basys3 board via the GPIO pins.  The RPI zero has a python script using the PySerial module in order to format the data from the basys3 board. The web server is also hosted on the rpizero via a python script using the Flask module. These two scripts are executed on parallel threads to facilitate real time relaying of data from the basys3 board to the web server. | Audio\_rpi\_overview |
| The serial transmission from the basys3 to the rpi zero consists of three data values: the analog output from the temperature sensor, and the digital outputs from two onboard switches. The raw analog value is normalized to a temperature in Celcius or Fahrenheit, depending on the state of the first onboard switch. The second onboard switch is used to arm and disarm the system. The state of the two switches as well as the ambient temperature reading are uploaded to the webserver. When the system is armed, if the ambient temperature exceeds a user-defined value, an alert will be displayed on the web server. | Audio\_serial\_interface |
| The GPIO pins are connected to the JB pmod ports on the basys3. These values are used to drive the seven segment display, which provides a visual indication of the ambient temperature. Eight GPIO pins are used in total, enabling the transmission of one byte of data that can be decoded into two digits on the seven segment display. The python script controls the transmission of this data by selectively asserting GPIO pins on the rpizero. | Audio\_gpio\_interface |
| Finally, lets have a look at the system as a whole. (cue music?)  Thank you for your time, and we hope that you have enjoyed this short video demonstration. | Audio\_outro |