

### Mini NPU Design How to Test:

#### **Post-silicon testing:**

Connect an ADXL345 accelerometer with I2C to the appropriate pins as outlined in chip.sv. Note that io\_in and io\_out should be tied together for correct bidirectional functionality. Press the start button to begin motion detection. Continuously move the connected accelerometer in a single direction for at least two seconds, and the connected LEDs will light up with binary values corresponding to the direction you swiped. Press the stop button to end motion detection.

#### **Testbench testing:**

If testing with the testbench accelerometer\_tb.py, run 'make -Bf testbench.mk' on the design. Having python3 and cocoTB installed is required. The testbench acts as the accelerometer device, and checks that the design performs all I2C data interactions correctly, as in accordance with the *UM10204 I2C-Bus Specification and User Manual*, and checks that the design correctly detects motions based on the data that was given.

#### **FPGA testing:**

If testing with an FPGA, switch out SDA\_in for 1'bz (high-z) in the tri-state driver for SDA in the I2C module. Run yosys on the design with 'read\_verilog -sv mini\_npu.sv', then 'synth\_ecp5 -json synth\_out.json -top Mini\_NPU'. Exit yosys, then run 'nextpnr-ecp5 --12k --json synth\_out.json --lpf constraints.lpf --textcfg pnr\_out.config' with the provided constraints.lpf file (edit this as needed). Then run 'ecppack --compress pnr\_out.config bitstream.bit' and then 'fujprog bitstream.bit' to load bitstream onto the FPGA.