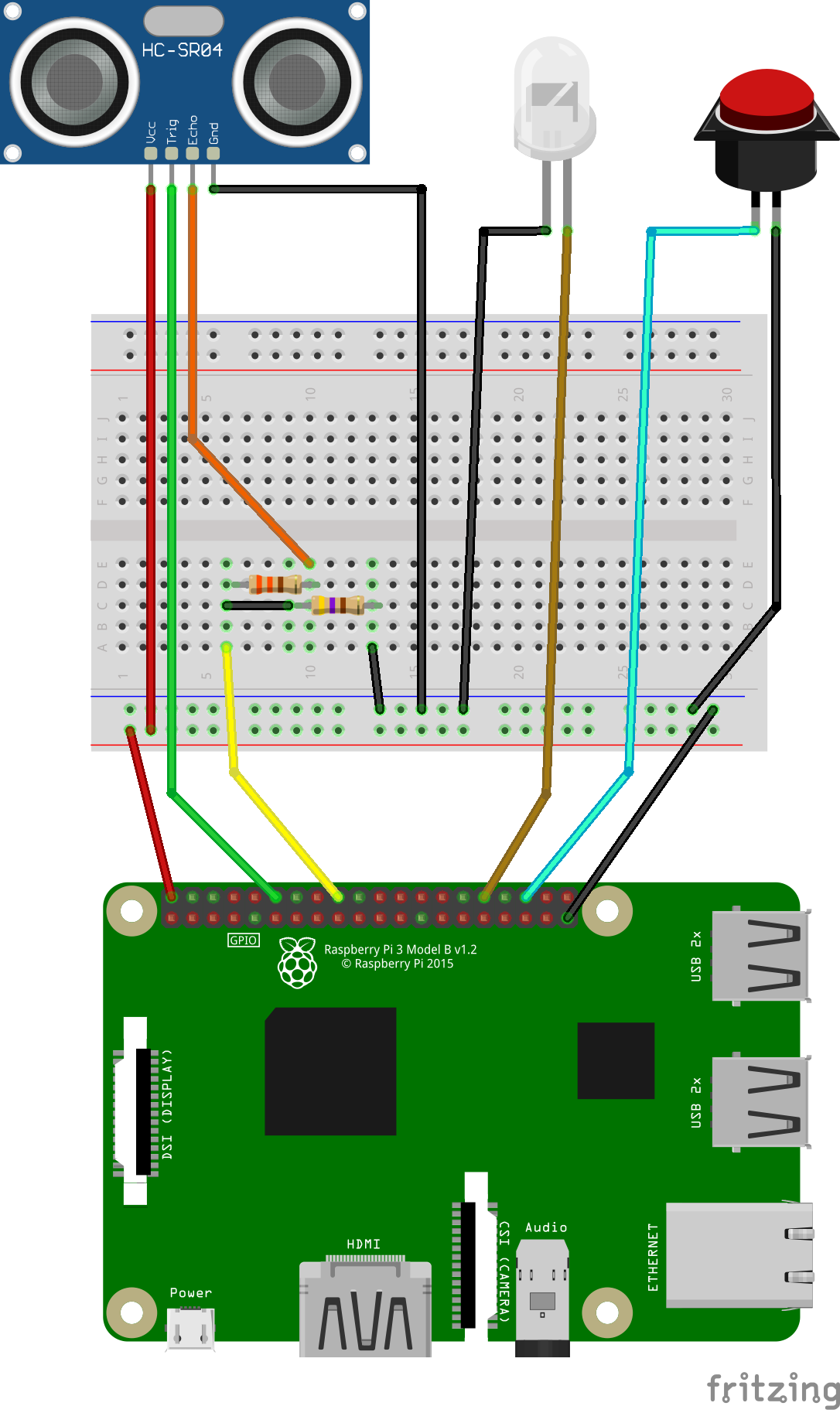
# Handout for Polling vs Interrupt RPi Weekly

## Parts List

* HC-SR04 Ultrasonic Distance Sensor
* LED
* Pushbutton
* 330Ω Resistor
* 470Ω Resistor
* Any Raspberry Pi with GPIO pins
* Breadboard
* Jumper Wires

## Circuit Diagram



## Circuit Description

* Connect the Breadboard
  + RPi board pin 2 (5vdc) to Breadboard Power Rail
  + RPi board pin 39 (gnd) to Breadboard Ground Rail
* Connect the HC-SR04
  + Breadboard 5v to HC-SR04 vcc
  + RPi board pin 12 (GPIO18) to HC-SR04 trigger
  + RPi board pin 18 (GPIO 24) to breadboard (I’ll call this BB:*A*)
  + 330Ω resistor from BB:A to HC-SR04 echo
  + 470Ω resistor from BB:A to breadboard ground rail
  + HC-SR04 gnd to breadboard ground rail
  + Use test\_hc-sr04.py to check functionality. Wave a piece of cardboard in front of the sensor from three inches to three feet - the readout in the console should indicate the change in distance.
  + If the readout doesn’t change, check the value of the resistors. Also check to be sure you are using the correct GPIO pins.
* Connect the LED
  + Cathode to breadboard ground rail
  + Anode to RPi board pin 32 (GPIO 12)
  + Use test\_led.py to check functionality. The LED should blink three times; on for a second, off for a second.
  + If the LED doesn’t light up, try reversing the polarity of the LED by rotating the connections 180 degrees. Also check to be sure you are using the correct GPIO pins.
* Connect the Pushbutton
  + One side to breadboard ground rail
  + The other side to RPi pin 36 (GPIO16)
  + Use test\_button.py to check functionality. Pushing the button should change the message in the console.
  + If the message doesn’t change, check the jumper wires between switch and breadboard. If the message says the button is pressed when it is not, check if the pushbutton is normally-closed. If there is no change, check to be sure you are using the correct GPIO pins.