|  |
| --- |
| Import time |
|  | import sys |
|  |  |
|  | EMULATE\_HX711=False |
|  |  |
|  | referenceUnit = 1 |
|  |  |
|  | if not EMULATE\_HX711: |
|  | import RPi.GPIO as GPIO |
|  | from hx711 import HX711 |
|  | else: |
|  | from emulated\_hx711 import HX711 |
|  |  |
|  | def cleanAndExit(): |
|  | print("Cleaning...") |
|  |  |
|  | if not EMULATE\_HX711: |
|  | GPIO.cleanup() |
|  |  |
|  | print("Bye!") |
|  | sys.exit() |
|  |  |
|  | hx = HX711(5, 6) |
|  |  |
|  | # I've found out that, for some reason, the order of the bytes is not always the same between versions of python, numpy and the hx711 itself. |
|  | # Still need to figure out why does it change. |
|  | # If you're experiencing super random values, change these values to MSB or LSB until to get more stable values. |
|  | # There is some code below to debug and log the order of the bits and the bytes. |
|  | # The first parameter is the order in which the bytes are used to build the "long" value. |
|  | # The second paramter is the order of the bits inside each byte. |
|  | # According to the HX711 Datasheet, the second parameter is MSB so you shouldn't need to modify it. |
|  | hx.set\_reading\_format("MSB", "MSB") |
|  |  |
|  | # HOW TO CALCULATE THE REFFERENCE UNIT |
|  | # To set the reference unit to 1. Put 1kg on your sensor or anything you have and know exactly how much it weights. |
|  | # In this case, 92 is 1 gram because, with 1 as a reference unit I got numbers near 0 without any weight |
|  | # and I got numbers around 184000 when I added 2kg. So, according to the rule of thirds: |
|  | # If 2000 grams is 184000 then 1000 grams is 184000 / 2000 = 92. |
|  | hx.set\_reference\_unit(113) |
|  | #hx.set\_reference\_unit(referenceUnit) |
|  |  |
|  | hx.reset() |
|  |  |
|  | hx.tare() |
|  |  |
|  | print("Tare done! Add weight now...") |
|  |  |
|  | # to use both channels, you'll need to tare them both |
|  | #hx.tare\_A() |
|  | #hx.tare\_B() |
|  |  |
|  | def loop(): |
|  | try: |
|  | # These three lines are usefull to debug wether to use MSB or LSB in the reading formats |
|  | # for the first parameter of "hx.set\_reading\_format("LSB", "MSB")". |
|  | # Comment the two lines "val = hx.get\_weight(5)" and "print val" and uncomment these three lines to see what it prints. |
|  |  |
|  | # np\_arr8\_string = hx.get\_np\_arr8\_string() |
|  | # binary\_string = hx.get\_binary\_string() |
|  | # print binary\_string + " " + np\_arr8\_string |
|  |  |
|  | # Prints the weight. Comment if you're debbuging the MSB and LSB issue. |
|  | val = hx.get\_weight(5) |
|  | print(val) |
|  | return val |
|  |  |
|  | # To get weight from both channels (if you have load cells hooked up |
|  | # to both channel A and B), do something like this |
|  | #val\_A = hx.get\_weight\_A(5) |
|  | #val\_B = hx.get\_weight\_B(5) |
|  | #print "A: %s B: %s" % ( val\_A, val\_B ) |
|  |  |
|  | hx.power\_down() |
|  | hx.power\_up() |
|  | time.sleep(0.1) |
|  |  |
|  | except (KeyboardInterrupt, SystemExit): |
|  | cleanAndExit() |
|  |  |
|  |  |