

ISAT 300

Spring Semester 2025, Section 1

Lab Instructor: Dr. Chris Bachman

Experiment #4: Ultrasonic Sensors

Date: February 26th, 2025

**Authors: Joseph Gros, Alex Chizmadia, Sebastian Erb,
Jaylon Taylor**

Project Leader: Sebastian Erb

**Honor Pledge: On our honor, we (Joseph Gros, Alex
Chizmadia, Sebastian Erb, Jaylon Taylor) have neither given
nor received unauthorized help on this assignment**

Abstract:

Ultrasonic sensors are widely used in various technological applications, particularly in autonomous vehicle systems, where precise distance measurements are critical for navigation, obstacle avoidance, and overall safety. This lab was designed to investigate the performance, accuracy, and practical applications of the HC-SR04 ultrasonic sensor when integrated with a Raspberry Pi microcontroller, with a specific focus on its ability to measure distances, assess precision, and calculate velocity and acceleration based on detected motion. The experimental setup involved wiring the ultrasonic sensor to the Raspberry Pi, followed by the implementation of Python-based code to trigger the sensor, record data, and analyze results. Calibration tests were conducted to determine whether the sensor performed within the manufacturer's specifications, with repeated trials conducted at various fixed distances to establish consistency and accuracy in measurements. Additionally, data was analyzed to identify the sensor's minimum and maximum effective range, evaluate potential sources of uncertainty, and implement a secondary feature in which an LED warning system was activated when an object approached a specific threshold distance. The sensor's response time and data acquisition rate were also analyzed by modifying the code to determine the fastest possible sampling rate that still maintained reliable results, allowing for an evaluation of how efficiently the sensor could track rapid movement. Furthermore, motion-based trials were conducted where an object was moved at different speeds in front of the sensor, enabling calculations of velocity and acceleration using time-stamped distance readings, which provided insights into the sensor's capabilities for real-time motion tracking. The experimental results demonstrated that the ultrasonic sensor was able to produce reasonably accurate distance measurements within its expected operational range, with deviations observed at extreme distances that suggested limitations in either the sensor's precision or environmental factors affecting wave reflection. The data collected also indicated that while the sensor was effective for basic velocity and acceleration calculations, its inherent latency and sensitivity to environmental conditions, such as surface reflectivity and angle of incidence, introduced variability in motion-tracking accuracy, making it less suitable for high-speed applications requiring extremely precise measurements. Despite these limitations, the findings from this lab reinforce the practical applications of ultrasonic sensors in low-risk environments where moderate accuracy is sufficient, while also highlighting the necessity of careful calibration, statistical analysis of uncertainties, and consideration of hardware limitations when implementing such sensors in more advanced systems, such as robotics and autonomous navigation technologies.

Introduction:

The emergence of the Raspberry Pi has significantly impacted educational tools and hobbyist projects, offering an affordable and accessible platform for bridging technology and creativity. However, this lab places a particular emphasis on ultrasonic sensors and their application in real-world scenarios, especially in the context of autonomous vehicles. Ultrasonic sensors, renowned for their accuracy in distance measurement, play a pivotal role in autonomous systems, where their ability to detect objects and measure distances with precision is critical for navigation, obstacle avoidance, and safety features. This lab aims to explore the integration of ultrasonic sensors with the Raspberry Pi to measure distances and then apply these measurements to control external systems or devices, mimicking the functionality of sensors used in autonomous vehicles. [1]

Autonomous vehicles rely heavily on sensors such as ultrasonic devices to perceive their environment and make real-time decisions, including determining the proximity of obstacles, detecting parking spaces, and navigating tight spaces. The use of ultrasonic sensors allows these vehicles to measure distances accurately, enabling safe navigation in various conditions. By utilizing the Raspberry Pi as an interface, this lab demonstrates how students can engage with real-world technologies that are essential to autonomous systems. Through Python programming, participants can program the Raspberry Pi to process sensor inputs, such as triggering an alarm or activating a braking system when an object is detected too close, which mirrors functions in autonomous systems.[2]

In the context of autonomous vehicles, ultrasonic sensors offer a cost-effective solution for close-range object detection and are often used in tandem with other sensor technologies, such as cameras and LiDAR, to create a comprehensive perception system. This lab not only enhances students' understanding of ultrasonic sensors and Python programming but also provides insight into the practical applications of these sensors in modern technology. Students will learn how to integrate sensors with physical systems, analyze data, and apply it to control devices, bridging theoretical knowledge with practical, real-world applications in robotics and autonomous vehicle systems. The project's focus on the measurement precision and reliability of ultrasonic sensors reflects their critical role in ensuring the safety and effectiveness of autonomous systems, where accurate distance sensing is vital for vehicle navigation and interaction with the environment.

Methods and Materials:

The purpose of this lab was to gain familiarity with operating an ultrasonic sensor with a computer for data acquisition and processing, then determine if the sensor was within the manufacturer's specifications. With the advancement of technology in the 21st century, ultrasonic sensors have become a commonly used device in a wide array of applications with continued hopes of advances, highlighted in autonomous vehicles, climate control systems, industrial robotics, and much more.. In this experiment, a HC-SR04 5V ultrasonic sensor was wired to a Raspberry Pi 4B via a voltage divider constructed on an electronics breadboard with additional wires, one 1000 Ω resistor, and one 1500 Ω resistor. Then, python codes were inputted that instructed the Raspberry Pi and sensor to

perform certain commands under specific conditions. Calibration tests were done prior to any experimentation to ensure precise and accurate recordings before any legitimate experiments were done. During experimentation, using an meter stick and a large enough object (like a box or white board), the maximum and minimum recordings were taken along with their mean and standard deviations from Microsoft Excel. From there, the uncertainty of the sensor was calculated with recordings of 100+ individual measurements, and the fastest speed of the sensor by manipulating various factors like the timing in the code and the object's distance from the sensor while considering the unchangeable factors like the speed of the processor and memory card. Then, a LED circuit that illuminated and alerted the system was installed to signal to the operator when an object got too close to the sensor. Finally, the velocity and acceleration were determined between two distances at two different times.

Ultrasonic Sensor and Raspberry Pi Setup

The ultrasonic sensor was electronically connected to the Raspberry Pi through a voltage divider. This was necessary because the Raspberry Pi operates at a maximum voltage of 3.3V, while the ultrasonic sensor requires 5V for accurate measurements. To bridge this voltage difference, the sensor was powered by the 5V supply from the Raspberry Pi's GPIO, and a voltage divider was built on a breadboard using additional wires and resistors. This setup split the sensor's output signal between the Raspberry Pi and the ground of the ultrasonic sensor, ensuring the voltage remained below 3.3V. The inputs and outputs of the ultrasonic sensor are illustrated in **Image 1: Ultrasonic Sensor Inputs/Outputs**, while the complete sensor and Raspberry Pi setup is depicted in **Image 2: Ultrasonic Sensor and Raspberry Pi Circuit Diagram**, both sourced from the digital laboratory manual [3]. According to these images, the ultrasonic sensor's +5V input was linked to the Raspberry Pi's 5V pin, the 'Trigger' pin was connected to 'GPIO 21 SCLK (SPI 1),' and the 'Echo' pin was split—leading to 'GPIO 20 MISO (SPI 1)' and 'Ground'—through a voltage divider consisting of a 1000 Ω resistor and a 1500 Ω resistor.

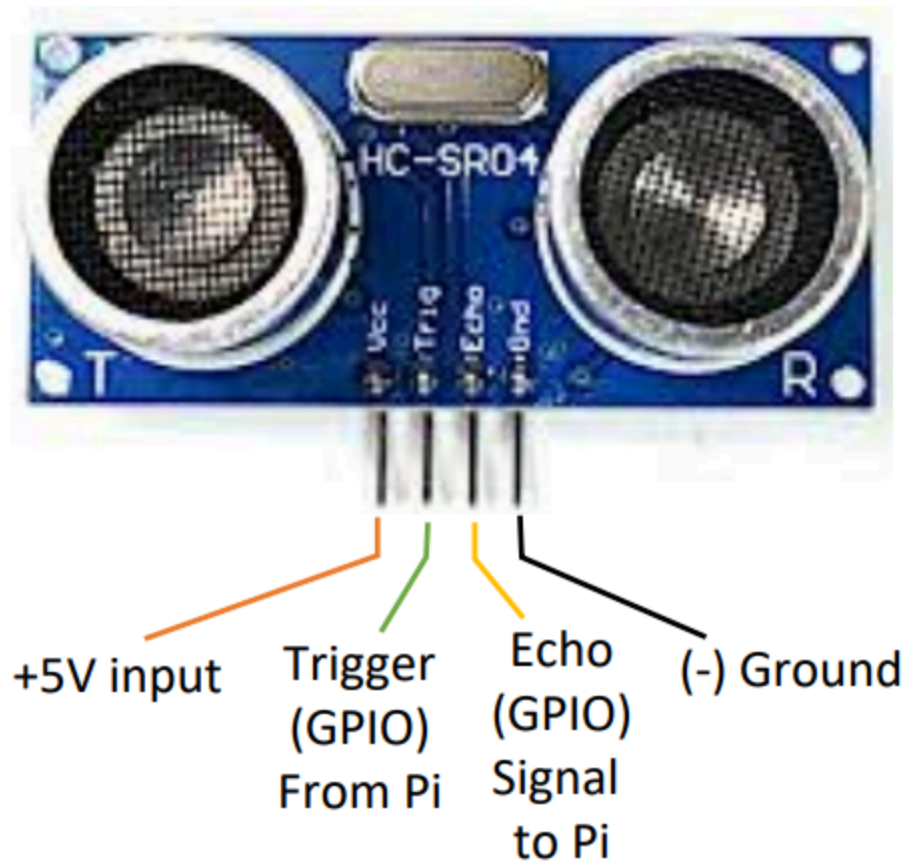


Image 1: Ultrasonic Sensor Inputs/Outputs

Image 1 provides a clear visualization of the ultrasonic sensor's inputs and outputs, facilitating easier identification of its connections with the voltage divider and the Raspberry Pi.

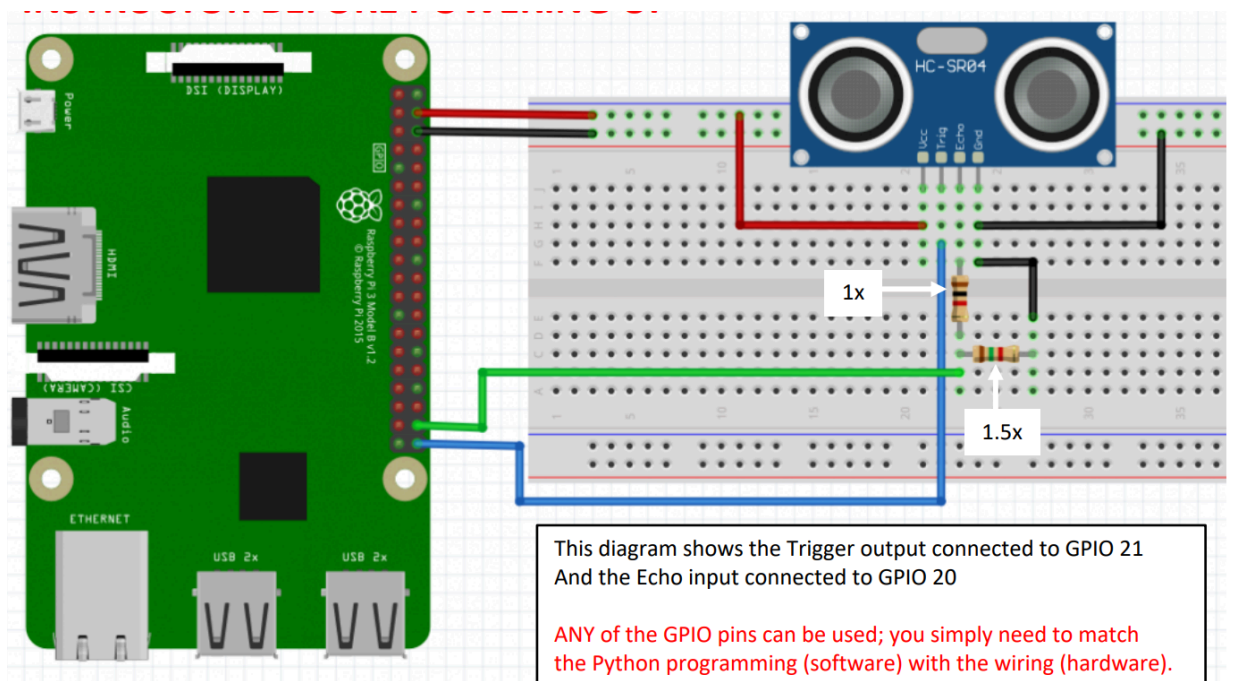


Image 2: Ultrasonic Sensor and Raspberry Pi Circuit Diagram

Image 2 shows a circuit diagram of the final setup for the ultrasonic sensor connected to the voltage divider to the Raspberry Pi should look like.

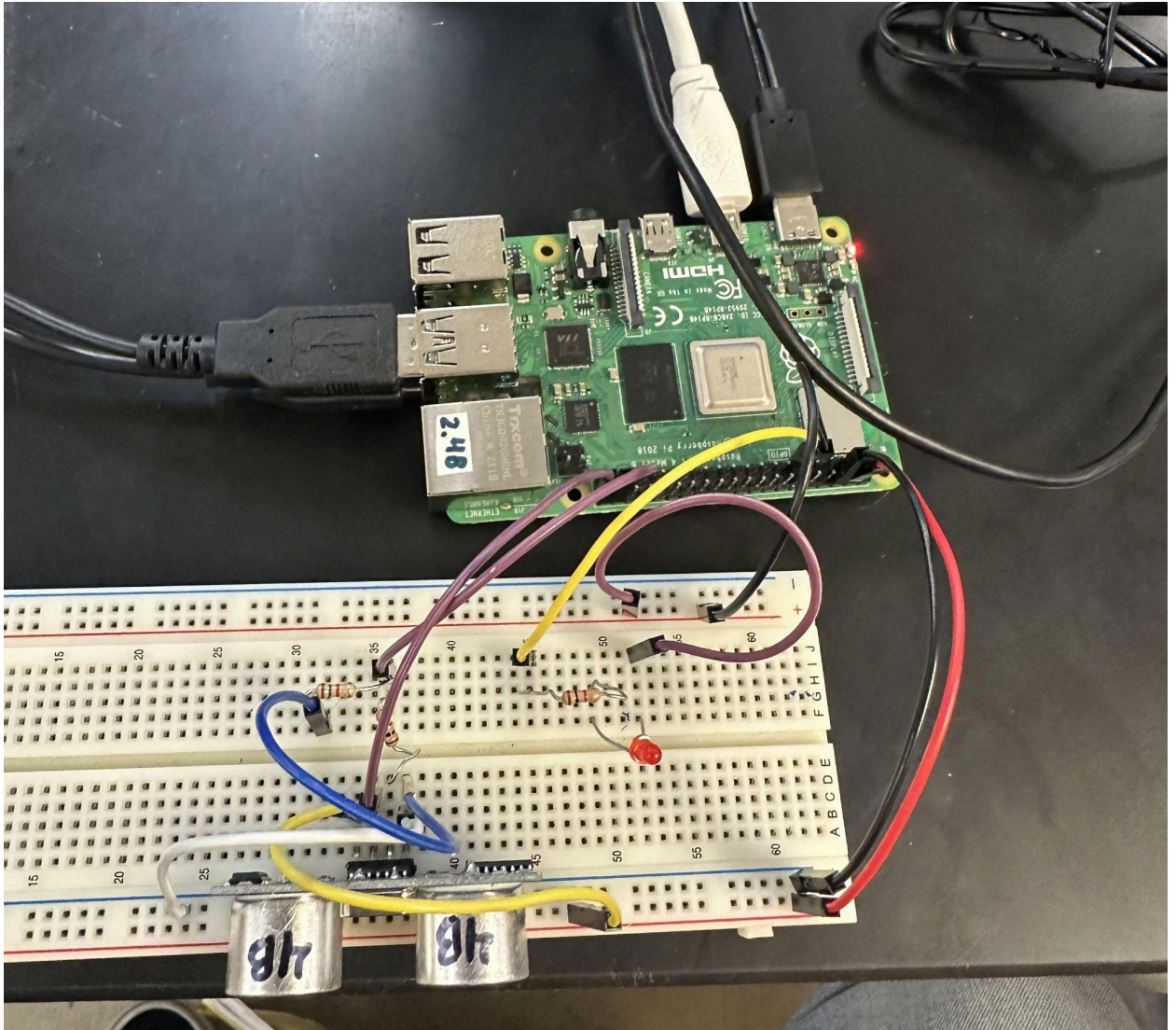


Image 3: Final Setup of Ultrasonic Sensor

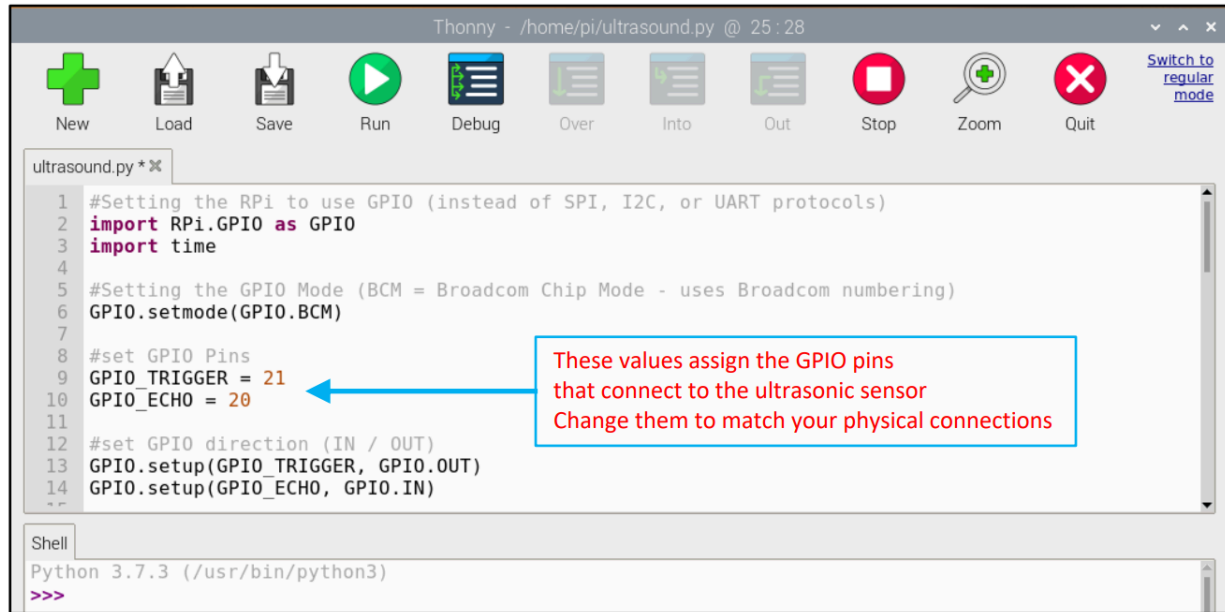
Image 3 showcases the final setup after assembling the ultrasonic sensor, voltage divider, and Raspberry Pi, clearly reflecting the design outlined in Image 2.

Next, the cables were connected to the power outlet, computer monitor, and Raspberry Pi keyboard to input codes for operating the ultrasonic sensor.

Python Coding Setup

After assembling the hardware and logging into the Raspberry Pi, Python code was entered into the Thonny

Python Integrated Development Environment (IDE). The initial step involved defining the GPIO input and output pins. **Image 4: Python Codes to Establish GPIO Input/Output Pins** displays a screenshot of the first 14 lines of code from Step 1: Setting the GPIO input/output pins, as outlined in the digital laboratory manual [3]. Lines 2-3 instruct the Raspberry Pi to utilize GPIO, while Line 6 designates Broadcom Chip numbering for the GPIO. Lines 9-10 assign pin 21 to the Trigger and pin 20 to the Echo, while Lines 13-14 specify that the Trigger functions as an output from the Raspberry Pi and the Echo as an input to it.



```
Thonny - /home/pi/ultrasound.py @ 25:28
New Load Save Run Debug Over Into Out Stop Zoom Quit Switch to regular mode

ultrasound.py *
1 #Setting the RPi to use GPIO (instead of SPI, I2C, or UART protocols)
2 import RPi.GPIO as GPIO
3 import time
4
5 #Setting the GPIO Mode (BCM = Broadcom Chip Mode - uses Broadcom numbering)
6 GPIO.setmode(GPIO.BCM)
7
8 #set GPIO Pins
9 GPIO_TRIGGER = 21
10 GPIO_ECHO = 20
11
12 #set GPIO direction (IN / OUT)
13 GPIO.setup(GPIO_TRIGGER, GPIO.OUT)
14 GPIO.setup(GPIO_ECHO, GPIO.IN)
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613

```

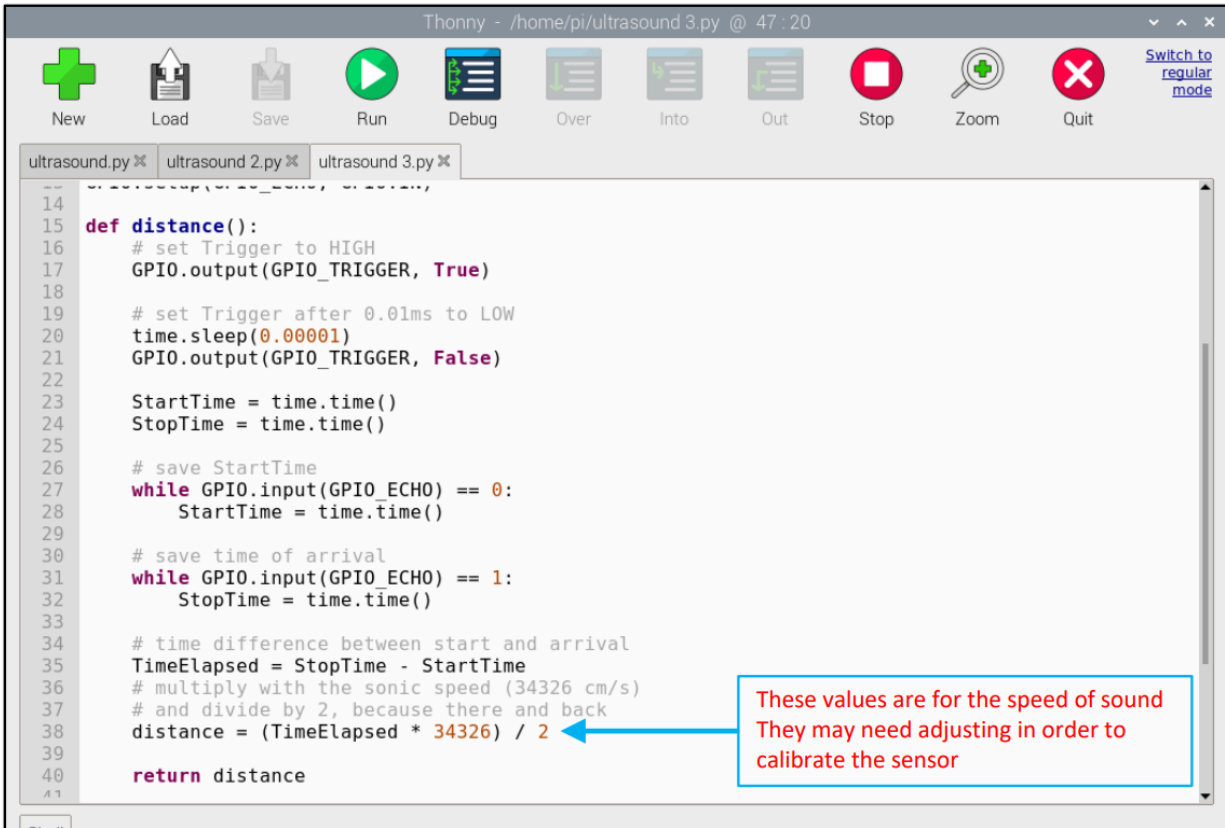
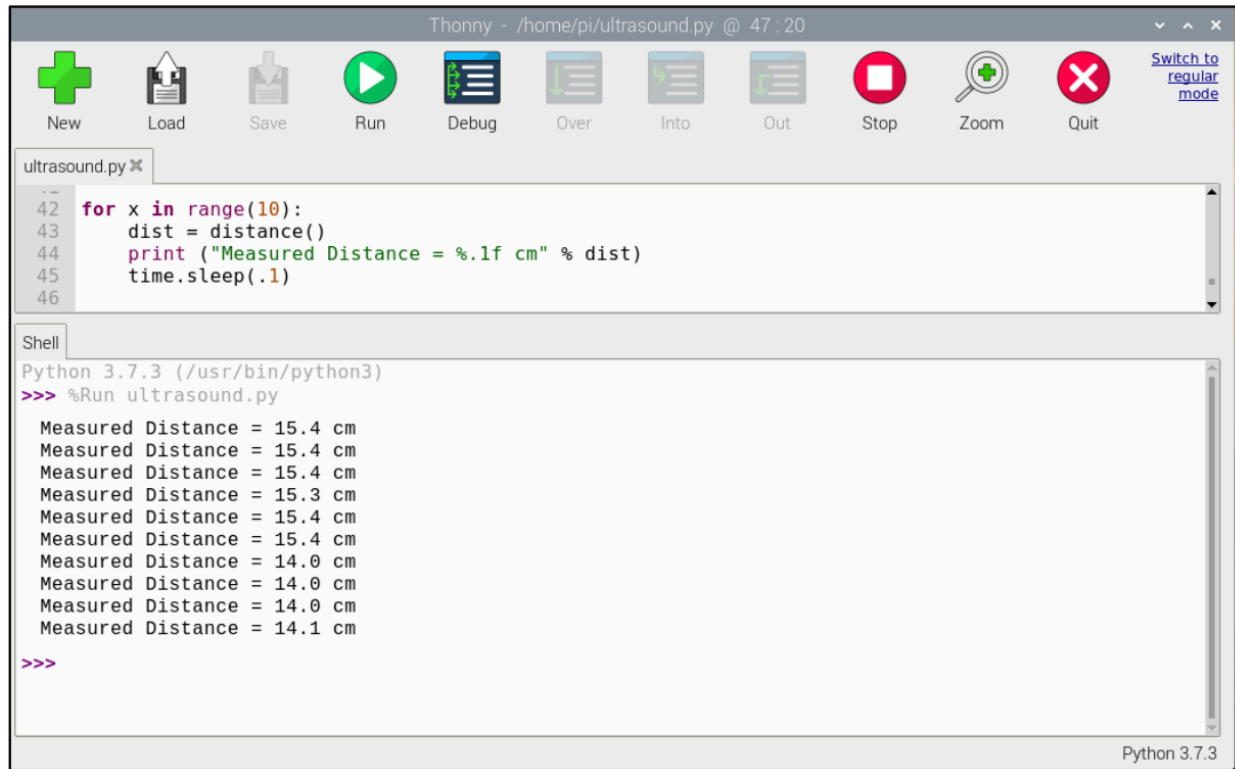


Image 5: Python Code to Activate the Ultrasonic Sensor

Image 5 displays a screenshot of the next 25 lines of code (lines 15-40) from Step 2: Activating the Ultrasonic Sensor, as outlined in the digital laboratory manual.

In the third step, code was inputted to allow the user to view the measurements while they were being recorded. **Image 6: Python Code to Print Measured Distances** shows the next three lines of code (lines 42-45) from Step 3: Printing the Measured Distance on the screen, as presented in the digital laboratory manual [3]. Line 42 initiates a for loop to collect a sample of 10 data points, which can be adjusted for a different sample size later. Line 43 assigns the variable “dist” to represent the distance, Line 44 prints the distance on the screen with “%.1f” later replaced by “%dist,” and Line 45 ends the measurement.



```
Thonny - /home/pi/ultrasound.py @ 47:20
New Load Save Run Debug Over Into Out Stop Zoom Quit Switch to regular mode

ultrasound.py
42 for x in range(10):
43     dist = distance()
44     print ("Measured Distance = %.1f cm" % dist)
45     time.sleep(.1)
46

Shell
Python 3.7.3 (/usr/bin/python3)
>>> %Run ultrasound.py
Measured Distance = 15.4 cm
Measured Distance = 15.4 cm
Measured Distance = 15.4 cm
Measured Distance = 15.3 cm
Measured Distance = 15.4 cm
Measured Distance = 15.4 cm
Measured Distance = 15.4 cm
Measured Distance = 14.0 cm
Measured Distance = 14.0 cm
Measured Distance = 14.0 cm
Measured Distance = 14.1 cm
>>>
```

Image 6: Python Code to Print Measured Distances

Image 6 displays a screenshot of lines 42 to 45 from Step 3: Printing the Measured Distance on the screen, as outlined in the digital laboratory manual.

The final lines of code are used to write the data into a .CSV file. **Image 7: Python Code to Write Data to .CSV File** presents a screenshot of lines 42 to 47 from Step 4: Writing the Data to a .CSV file, as shown in the digital laboratory manual [3]. Line 42 initiates a for loop for a sample size of 100 data points. Line 43 defines “dist” to represent the distance. Line 44 prints the distance measurement on the screen. Line 45 creates a .csv file named “ultrasounddata.csv” (which can be renamed for better identification), with the “a” indicating that the file should append data rather than overwrite or restart data collection. Line 46 specifies how the measurements should be recorded in the .csv file, such as in centimeters, inches, or feet. Line 47 halts the measurement. During experimentation, the units will need to be adjusted as necessary.

```
42 for x in range(100):
43     dist = distance()
44     print ("Measured Distance = %.1f cm" % dist)
45     f = open("ultrasounddata.csv", "a")
46     f.write('\n'+ "Actual Distance =, 10cm, Measured Distance, %.2f" % dist)
47     time.sleep(0.01)
```

Image 7: Python Codes to Write Data to .CSV File

Image 7 displays a screenshot of the updated set of codes from lines 42 to 47 from Step 4: Writing the Data to a .CSV File.

After setting up the software for the hardware, it was time to do the measurement and calculations, the first step being essential which is to establish the starting point for the measurements as 0 cm. After discussion, it was decided to set the starting point for all measurements at the edge of the sensor, as seen in **Image 8: Starting Point of Measurements**

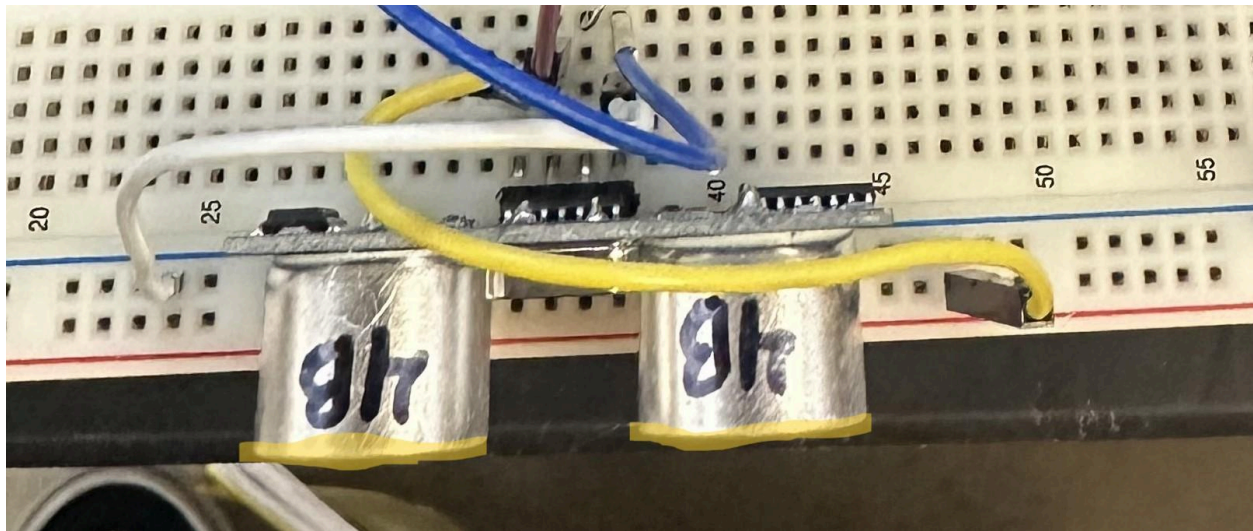


Image 8: Starting Points for Measurements

Image 8 represents the starting point of measurements, 0cm, indicated by the highlighted yellow market at the end of the sensor.

After setting a reference point, calibration tests were conducted to ensure the ultrasonic sensor provided accurate and precise readings. To assess the calibration, objects were placed at fixed distances from the sensor to create a baseline reference, and the sensor was activated. These distances included 20 cm, 50 cm, 100 cm, 200 cm, 300 cm, and 350 cm. Calibration runs were performed three times for each distance to ensure consistency. Following calibration, the maximum recordable distance of the ultrasonic sensor was tested by gradually increasing the distance of the object and measuring it with a meter stick. A large object, such as a whiteboard, was used to ensure the ultrasonic waves could reflect back to the sensor. After each run, the data was saved into a designated file. Once the maximum distance was determined, three trial runs were conducted to demonstrate consistent readings. The data was then transferred to a Microsoft Excel spreadsheet, where the mean and standard deviation of each data set were calculated. Next, the minimum recordable distance was assessed. Similar to the maximum distance, the distance was gradually reduced until the sensor could no longer provide accurate readings or produce errors. Measurements started at 20 cm and were shortened by 2-4 cm until the sensor could no longer generate precise readings. This phase also required three trial runs with consistent minimal distances, followed by calculations of the mean and standard deviation in Microsoft Excel.

The next experiment involved connecting a small LED to the Raspberry Pi and the already constructed circuit. **Image 9: LED Circuit Diagram** presents a screenshot from the digital laboratory manual [4], showing how the LED was to be attached to the preexisting circuit. It was important to understand that the LED could only operate in one direction: the shorter prong indicated negative (-), while the longer prong indicated positive (+). The negative terminal of the LED was wired to the common ground, while the positive terminal was connected to GPIO 17 on the 3.3V row, with a 1000 Ω resistor.

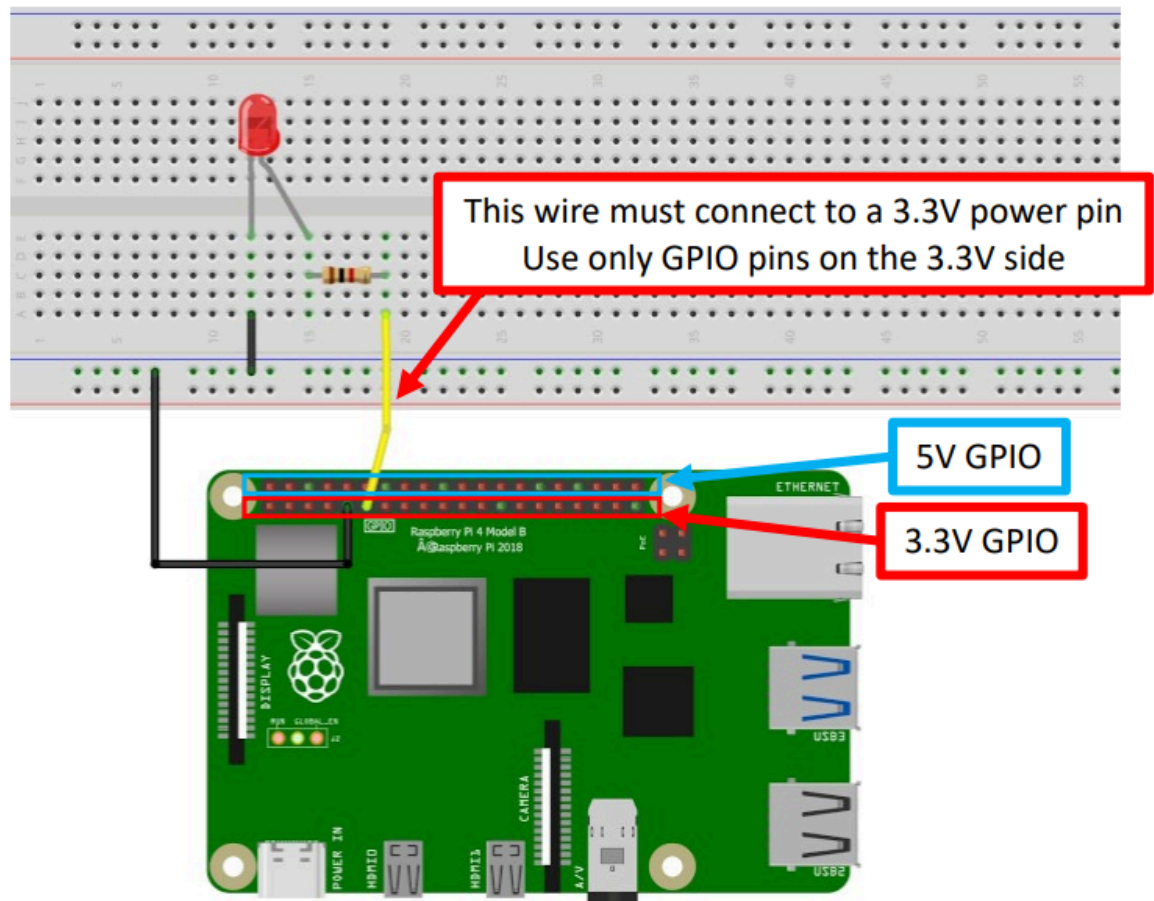


Image 9: LED Circuit Diagram

Image 9 displays a screenshot from the digital laboratory manual, showing how the LED was to be connected to the preexisting circuit.

Additional lines of code were then added to the existing Python script. **Image 10: Python Codes for Alert System** shows a screenshot from the digital laboratory manual, highlighting the newly added code [4]. These additions enabled the sensor to trigger the ultrasonic sensor and illuminate the LED when an object comes too close, which is a common feature in modern technology. The first line of code assigns GPIO 17 to the LED pin, in

line with the wiring setup. The second line ensures the LED remains off when not activated. The rest of the code consists of an if-else statement, where the target distance is defined first. The following lines activate the LED and display the “WARNING!!!!” message on the software. The else statement turns the LED off, returning it to its initial state.

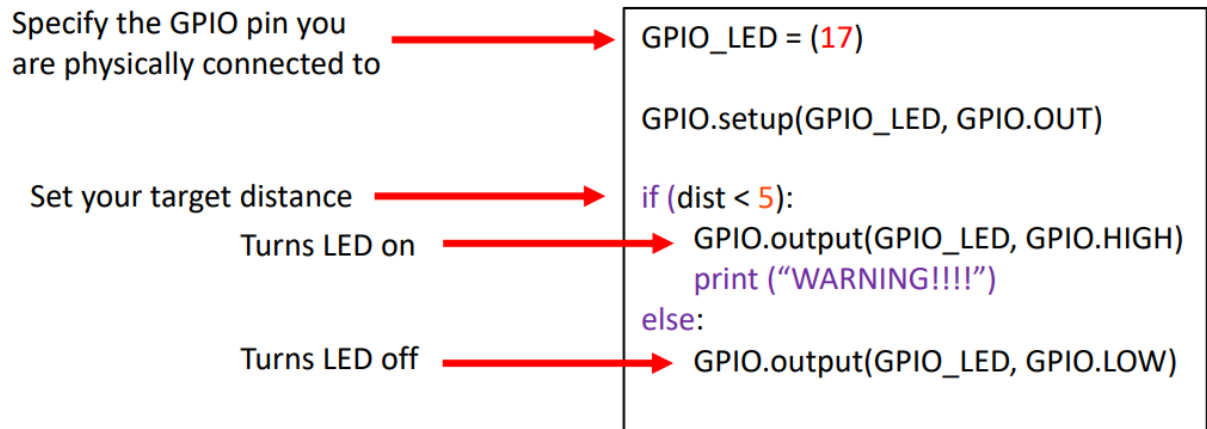


Image 10: Python Codes for Alert System

Image 10 displays the collection of Python code that was added to the preexisting lines.

The final experiment involved modifying the existing Python code or manually calculating the velocity of the object based on the ultrasonic sensor's measurements between two successive readings. The following formula was crucial for calculating velocity:

$$Velocity (m/s) = \frac{\Delta Distance}{\Delta Time} = \frac{(Distance\ 2 - Distance\ 1)}{(Time\ 2 - Time\ 1)}.$$

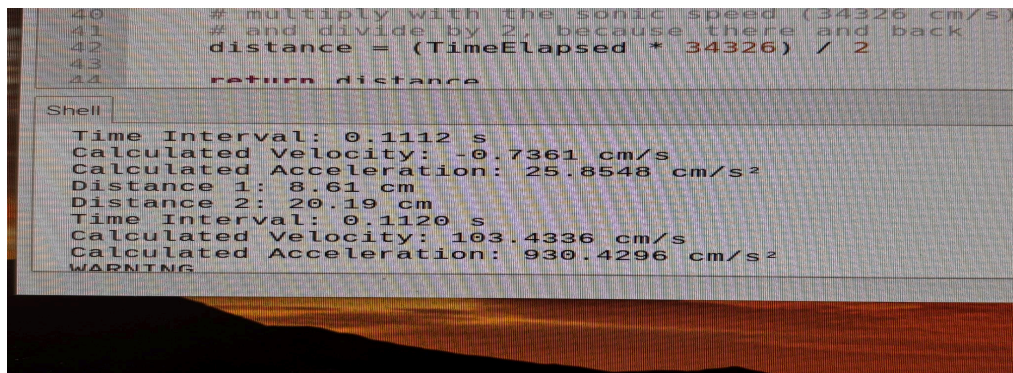
The data collected from evaluating the ultrasonic sensor's fastest processing speed (when the object was moved while the sensor was collecting measurements, as described in the "Uncertainty and Fastest Processing Speed" section) was used to calculate the velocity. Following this, the formula below was applied to calculate the object's acceleration during the same experiment:

$$Acceleration (m/s^2) = \frac{\Delta Velocity}{\Delta Time}$$

Results:

The purpose of this lab exercise is to provide hands-on experience in controlling sensors using a computer and utilizing the computer for data acquisition, processing, and control of physical systems. In this lab, an ultrasonic sensor is used to measure distance, controlled by a Raspberry Pi computer interfaced through the General Purpose Input and Output (GPIO) pins. The lab consisted of two 50 sampling rates which equates to 100 trials. The data collected is intended to investigate the optimal sampling rate, minimum distance, maximum distance, velocity, and acceleration using the ultrasonic sensor, which is stated to detect distances of up to 400 cm.

The optimal sampling rate came out to be in the range of 50. This is due to the sensor being able to collect data at a slower but more accurate rate compared to the given 100 in the lab. This could be also due to the processor of the raspberry pi as a better processor would make it run more smoothly giving more accurate data points over time.



```
40 # multiply with the sonic speed (34326 cm/s)
41 # and divide by 2, because there and back
42 distance = (TimeElapsed * 34326) / 2
43
44 return distance

Shell
Time Interval: 0.1112 s
Calculated Velocity: -0.7361 cm/s
Calculated Acceleration: 25.8548 cm/s²
Distance 1: 8.61 cm
Distance 2: 20.19 cm
Time Interval: 0.1120 s
Calculated Velocity: 103.4336 cm/s
Calculated Acceleration: 930.4296 cm/s²
WARNING
```

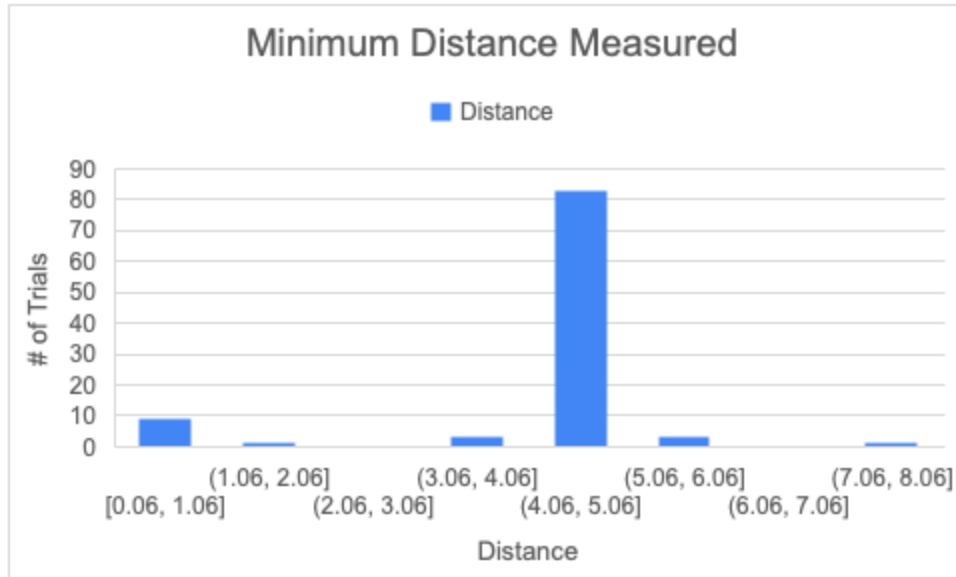
Image 11: Shows the calculated velocity and calculated acceleration of an object being dropped on the floor.

The minimum distance measured by the ultrasonic sensor came out to be 0.06cm during the two 50 sample rates in trials. This was done in 100 trials as the range for the data was from 0.06cm to 7.17cm. The uncertainty of the measured minimum distance for one standard deviation or 68% confidence interval is between 3.09cm to 5.75cm.

Minimum Distance

Minimum	Maximum	Mean	STD
0.06cm	7.17cm	4.42cm	1.33cm

Table 1: Table describes the min, max, mean, and standard deviation of the ultrasonic sensor measuring the minimum distance.



Graph 1: This graph shows over a 100 trials how the distance mostly stays around 4–5 cm range. There are multiple outliers where the distance drops to nearly 0 cm, indicating sudden reductions and measurements in the 7cm - 8cm range. The outliers suggest fluctuating measurements, possibly due to an object moving back and forth or sensor inconsistencies.

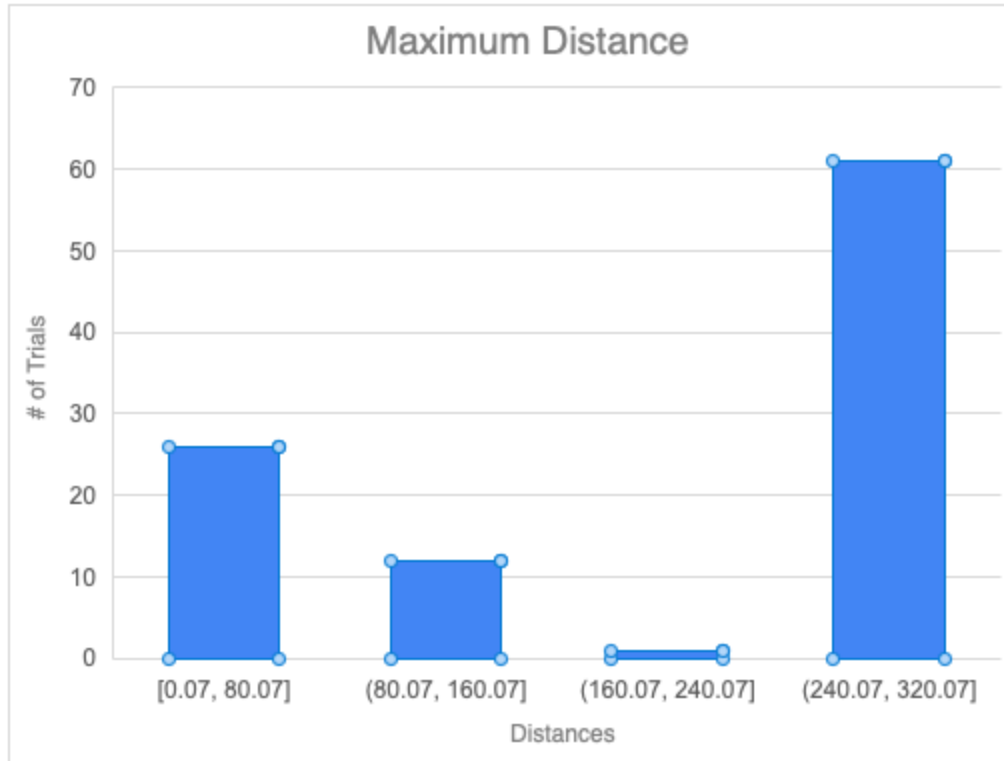
The maximum distance measured by the ultrasonic sensor came out to be 302.03 cm during the two 50 sample rates in trials. This was done in 100 trials as the range for the data was from 0.07cm to 302.03 . The uncertainty of the measured maximum distance for one standard deviation or 68% confidence interval is between 82.2cm to 294.9cm.

Maximum Distance

Minimum	Maximum	Mean	STD
.07cm	302.03cm	188.5cm	106.3cm

Table 2: Showing the min, max, mean, and STD for the maximum distance measured using the ultrasonic sensor.

The minimum distance measured by the ultrasonic sensor came out to be 0.06cm during the two 50 sample rates in trials. This was done in 100 trials as the range for the data was from 0.06cm to 7.17cm. The uncertainty of the measured minimum distance for one standard deviation or 68% confidence interval is between 3.09cm to 5.75cm.



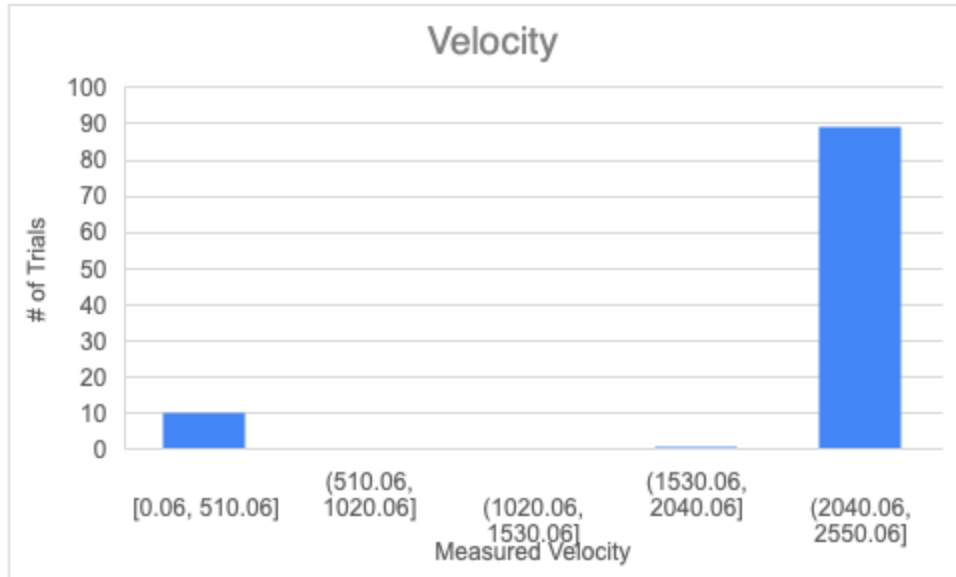
Graph 2: This graph shows over a 100 trials how the maximum distance was mostly in the 240.07-320.07 range. The average was 188cm even though the trials itself doesn't have many data points in that range indicating there were many outliers in the measurements. This can be due to the sensor and the calibration of the ultrasonic sensor.

The mean velocity measured by the ultrasonic sensor came out to be 202 m/s during the two 50 sample rates in trials. This was done in 100 trials as the range for the data was from 0.06m/s to 225 m/s. The uncertainty of the measured velocity for one standard deviation or 68% confidence interval is between 134.5m/s to 269.5m/s.

Measured Velocity

Mean	Maximum	Minimum	STD
202m/s	225m/s	0.06m/s	67.5m/s

Table 3: Showing the mean, max, min, and STD of the measured velocity using the ultrasonic sensor.



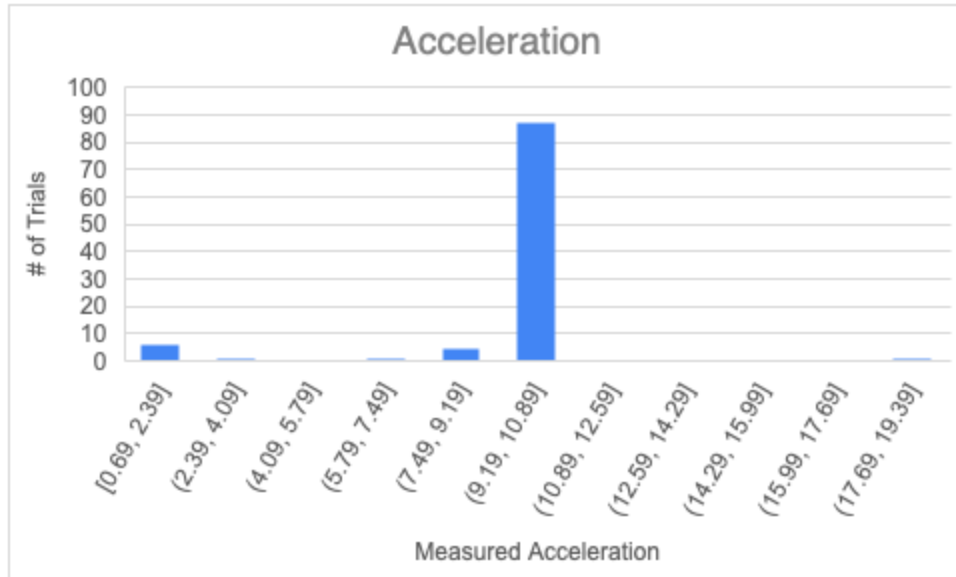
Graph 3: This graph shows over a 100 trials how the velocity measured was mostly in the 240.1-255.1 range. The average was 202m/s which validates the graph as most of the data points were in that range. There were many data points outside that range which were outliers. This can be due to the sensor not working properly for a few of the measurements and the calibration of the ultrasonic sensor being off.

The mean acceleration measured by the ultrasonic sensor came out to be 202 m/s during the two 50 sample rates in trials. This was done in 100 trials as the range for the data was from 0.69m/s² to 17.94m/s². The uncertainty of the measured acceleration for one standard deviation or 68% confidence interval is between 6.61m/s² to 11.15m/s².

Measured Acceleration

Mean	Maximum	Minimum	STD
8.88m/s ²	17.94m/s ²	0.69m/s ²	2.27m/s ²

Table 4: Showing the mean, max, min, and STD of the measured acceleration using the ultrasonic sensor.



Graph 4: This graph shows over a 100 trials how the acceleration measured was mostly in the 9.2m/s^2 - 10.9m/s^2 range. This makes sense as the acceleration on Earth equates to 9.81m/s^2 . The average was 8.88m/s^2 which validates the graph as most of the data points were near that range set. There were many data points outside that range which were outliers. This can be due to the sensor not working properly for a few of the measurements and the calibration of the ultrasonic sensor being off.

Two-Point Calibration

The ultrasonic sensor system measuring acceleration and velocity was calibrated using two reference points: 9.37 m/s^2 at 203 m/s and 7.7 m/s^2 at 224 m/s , with a true acceleration of 9.81 m/s^2 . Measured values deviated by 0.44 m/s^2 and 2.11 m/s^2 , respectively, indicating a scaling error in distance readings or calibration error in the ultrasonic sensor. This makes sense as the data wasn't giving both the acceleration and velocity at the same time during the measurements which can lead to inaccurate data. A correction factor of 1.16 was applied to raw distances, improving acceleration outputs toward 9.81 m/s^2 . The calibration addresses systematic underestimation, though velocity-dependent errors.

Discussion:

The purpose of this lab was to familiarize oneself with the operation of an ultrasonic sensor for data acquisition and processing, as well as to determine if the sensor operated within the manufacturer's specifications. The primary goals were to perform calibration tests, evaluate the maximum and minimum measurable distances of the HCSR04

5V ultrasonic sensor, calculate the uncertainty, and assess its accuracy through the movement of an object. Additionally, the sensor was re-programmed to measure velocity and acceleration, which required adjustments to the software. This process involved multiple measurements to minimize errors and ensure the reliability of the acquired data. The average of these trials provided a linear relationship between the measured distance and the actual distance, demonstrating the sensor's capability to provide calibrated and accurate readings. The minimal uncertainty observed is crucial for applications requiring precise distance measurements, such as robotics or autonomous vehicles, where even small errors could lead to significant issues.

When calculating the uncertainty for the minimum distance, it was found to be approximately 0.11 cm. This value is a result of repeated measurements and reflects the sensor's reliability at short distances. For the maximum measurable distance, the sensor demonstrated an uncertainty of 298.58 cm, determined through similar methods. The maximum distance achieved in the lab was 2241.61 cm, well within the manufacturer's specification of a 0.9 cm range. This uncertainty for maximum distance emphasizes the sensor's limitations, particularly in long-range detection. While larger sensors may be necessary for applications such as satellite-based missile tracking, this ultrasonic sensor performs adequately for low-risk applications like basic robotics or home automation.

The sensor was also tested to determine the maximum sampling rate at which it could reliably capture data. The maximum sample rate achieved was 100 samples per second, with an associated uncertainty of 10 samples per second. This rate was determined by modifying the Python code to increase the frequency at which data was collected. The uncertainty in the sample rate indicates potential limitations of the hardware and software, as the Raspberry Pi may not process data fast enough at higher sampling rates. In real-world applications, this may impact the ability to track fast-moving objects, such as in autonomous vehicles or high-speed industrial robotics. However, this sample rate is sufficient for many low-risk applications like fitness trackers or simple proximity sensors for smart device

A major goal of this lab was to modify the software to measure velocity and acceleration in addition to distance. To measure velocity, the sensor was programmed to take two distance measurements at different timestamps and calculate the difference in distance divided by the time interval, using the formula:

$$Velocity (m/s) = \Delta distance / \Delta time = (Distance\ 2 - Distance\ 1) / (Time\ 2 - Time\ 1)$$

This method successfully calculated velocity by determining the rate of change in position over time, providing useful data for tracking moving objects. Similarly, acceleration was determined by calculating the change in velocity over time using the formula:

$$Acceleration = \Delta Velocity / \Delta Time \quad \{Acceleration\} = \{\{\Delta Velocity\} / \{\Delta Time\}\} \quad Acceleration = \Delta Time \Delta Velocity$$

In this case, velocity readings were calculated at regular intervals, and the differences in velocity were used to calculate acceleration, providing insight into the sensor's ability to track changes in motion.

To ensure the accuracy of the velocity and acceleration measurements, controlled experiments were conducted with a moving object. The object was moved a known distance from the sensor, and the velocity and acceleration were hand-calculated based on the sensor's readings. Despite some human error in the moving object experiment, which likely led to inaccuracies in distance measurement, the calculations yielded velocity and acceleration values of 103.43 cm/s and 930 cm/s², respectively. These calculations provided a baseline for evaluating the sensor's capabilities in tracking moving objects.

The primary source of error in the moving object experiment likely stemmed from inconsistent object movement, as human error in manually moving the object could have introduced discrepancies. Additionally, the sensor may have lost its calibration during the experiment, further contributing to the inaccuracies. As a result, the velocity and acceleration calculations are not fully reliable and would require further refinement, particularly in more precise applications where high accuracy is necessary.

These results have real-world implications in several fields. For instance, in autonomous vehicles, precise velocity and acceleration measurements are critical for collision avoidance and navigation, and a high degree of accuracy is needed. While the ultrasonic sensor is not suitable for high-stakes applications such as these, its use in lower-risk applications like fitness trackers, home automation devices, and basic robotics is more appropriate. Sensors like this are ideal for tracking objects nearby, with minimal risk of catastrophic failure in the event of a miscalculation. However, sensors with greater accuracy and reliability would be required for more critical applications, such as missile tracking or industrial robots.

The ability to measure velocity and acceleration expands the potential applications of the ultrasonic sensor, particularly in scenarios where object movement is involved. However, the uncertainty in both distance and movement measurements suggests that this sensor is not a suitable choice for high-speed tracking or environments requiring pinpoint accuracy.

Conclusion:

The results of this experiment demonstrate that the ultrasonic sensor effectively measures distance, velocity, and acceleration, but with notable limitations in accuracy and precision. The minimum and maximum distances measured fell within the expected range, though inconsistencies and outliers suggest potential sensor calibration issues and environmental factors affecting performance. The measured velocity and acceleration values were reasonably accurate when compared to theoretical expectations, but some discrepancies highlight the limitations of using this sensor for high-precision motion tracking. The data further supports the conclusion that while the sensor is suitable for general proximity sensing and basic motion tracking, it may not be ideal for applications requiring highly accurate distance or speed measurements.

These findings largely align with the initial hypothesis that the ultrasonic sensor would provide reliable data within the manufacturer's specified range, though with some degree of uncertainty. The observed variability in measurements, particularly at extreme distances, suggests that while the sensor is generally accurate, its reliability diminishes in certain conditions. The calculation of uncertainty and standard deviation reinforces the importance of repeated trials and statistical analysis in evaluating sensor performance. Furthermore, the experiment confirms that software modifications can enable the sensor to measure velocity and acceleration effectively, but real-world applications may require more sophisticated calibration techniques to reduce errors.

This lab emphasizes the significance of understanding sensor limitations, calibration, and error analysis in data acquisition. The experiment highlights the fundamental trade-offs between measurement accuracy, precision, and environmental influences, demonstrating the necessity of statistical tools to quantify uncertainty. These concepts are critical in fields such as robotics, automation, and scientific research, where sensor reliability directly impacts decision-making and system performance. [5] While the ultrasonic sensor proved useful for basic measurements, its limitations emphasize the importance of selecting appropriate sensors for specific applications and implementing rigorous calibration procedures to ensure data integrity.

References:

- [1] Raspberry Pi. (n.d.). <https://www.raspberrypi.com/> date accessed 3/5/25
- [2] Yeong J, Velasco-Hernandez G, Barry J, Walsh J. Sensor and Sensor Fusion Technology in Autonomous Vehicles: A Review. *Sensors* (Basel). 2021 Mar 18;21(6):2140. doi: 10.3390/s21062140. PMID: 33803889; PMCID: PMC8003231.
- [3]. Bachmann, C., & Rudmin, J. (n.d.). *Measuring Distance with an Ultrasonic Sensor*. College of Integrated Science and Engineering James Madison University.
- [4]. Bachmann, C., & Rudmin, J. (n.d.). *Measuring Distance with an Ultrasonic Sensor Physical Computing*. College of Integrated Science and Engineering James Madison University.
- [5] Tripathy, D., R. Gottumukkala, and D. Kim. "Exploring the Nexus Between Sensor Reliability and System Performance: A Comprehensive Analysis". *Annual Conference of the PHM Society*, vol. 16, no. 1, Nov. 2024, doi:10.36001/phmconf.2024.v16i1.3888. Date accessed: Mar. 7th, 2025