Linear Pendulum Rangefinder for Education

By: Mathew Pula
April 22, 2014
Physics 2P32, Brock University

Abstract

The apparatus produced in this project can measure physical quantities of a pendulum bound by small amplitudes. This device is intended as an educational model for SHM, specifically that of a linear pendulum. The device produced can measure distances and periods of an oscillating pendulum. The data collected by the device can then be graphed.(e.g., by physica)

The device utilizes a rangefinder. The rangefinder emits ultrasonic pulses which are received again by the rangefinder. Then, the rangefinder creates a high on its output; proportional to the time between the emission and detection of the ultrasonic pulse. By measuring the time the output stays high, one can determine the distance the pulse had to travel, and thus the distance of an object.

To measure the period, a simple stopwatch was implemented. The stopwatch initially records the distance of a given object, then records the time until the stop button is pushed. This creates data of amplitude vs time.

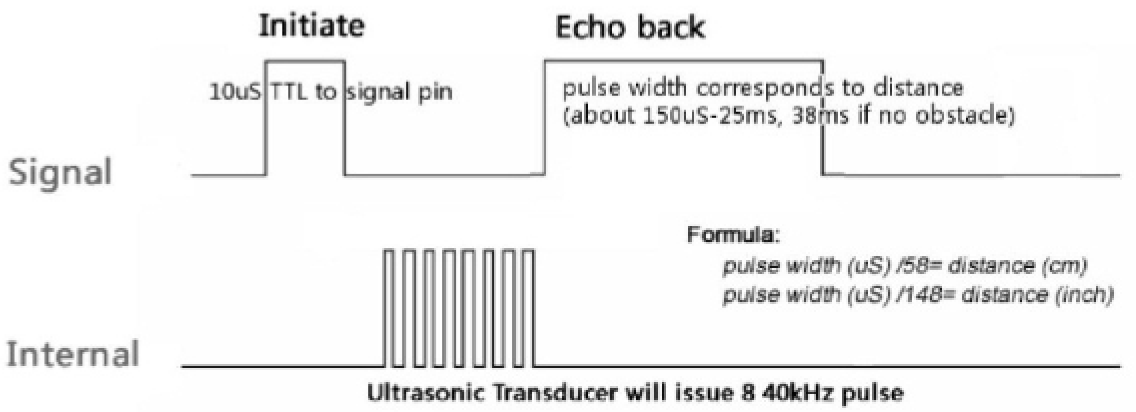
This device was created as an educational model for SHM. A linear pendulum was chosen because basic physical properties are observable in the model(e.g., the amplitude should never increase due to conservation of energy), as well as some properties specific to a pendulum. However, the device will also operate on other apparatus, such as a spring oscillator.

Contents **Page**

1.1:Introduction:

The sensor used in the project is a HC-SR04 Ultrasonic rangefinder. The device operates by emitting a bundle of ultrasonic pulses from an emitting device. This is done by setting the trigger high for a certain amount of time (~20μs was used in this project). Then, upon receiving the bundle of pulses in the receiving device, the sensor outputs a high signal on the echo output. The duration of the signal is equal to the time between the emission of the bundle and the reception of said bundle. From this, one can determine the distance by using a calibrated counter. In this project, the counter is calibrated to cycle ever 5.8μs. This gives the distance in millimeters. A timing diagram is shown below in figure 1.1.0.

Figure 1.1.0 HC-SR04 Timing Diagram



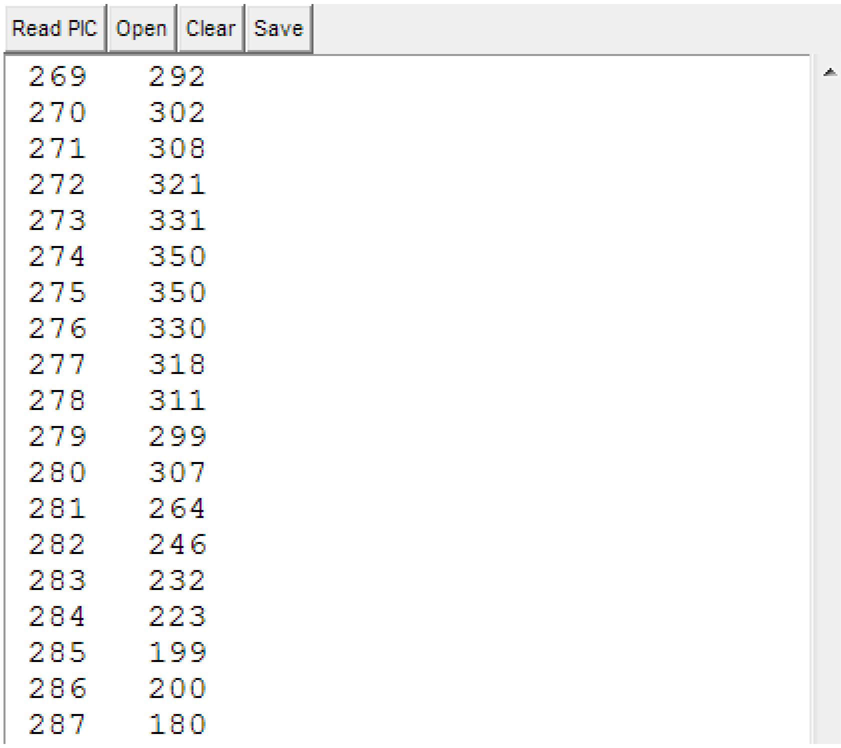
Taken from Cytron Technologies: *HC-SR04 User's_Manual*

The purpose of the device constructed in this project is to model the SHM of a linear pendulum. The device produced can model the position, x(t), of a pendulum or spring oscillator by providing data points; which can then be graphed. In addition, a stopwatch function has been included. This is to measure the period of oscillation, however it requires manual control.

This device measures distance fairly precise and accurately and could be used for quantitative analysis. However the period measurements, hence relying on the judgment of the user, should not be used quantitatively. Instead, the period should be determined graphically.

The device produced in this project is a distance measuring device. The device measures distance in millimeters, and outputs the data, along with a trial number, to the PICLab output. The trial number is added for graphing purposes. An example of this is shown below in figure 1.1.1. This process is automated. This mode is selected by pressing any button excluding reset and switch 2 on the Picboard.

Figure 1.1.1 Example Data Mode 1; left column = trial number, right column=distance(mm)



269	292
270	302
271	308
272	321
273	331
274	350
275	350
276	330
277	318
278	311
279	299
280	307
281	264
282	246
283	232
284	223
285	199
286	200
287	180

In addition to the distance measuring technique mentioned above; this device can also measure the period of an oscillation. The program first measures the distance between the object in question and the Picboard. Then, a counter is run until a button (switch 2) is released. The countermeasures in milliseconds. This mode is selected by pressing switch 2 and is manual. An example of the data collected via this mode of operation is shown in figure 1.1.2.

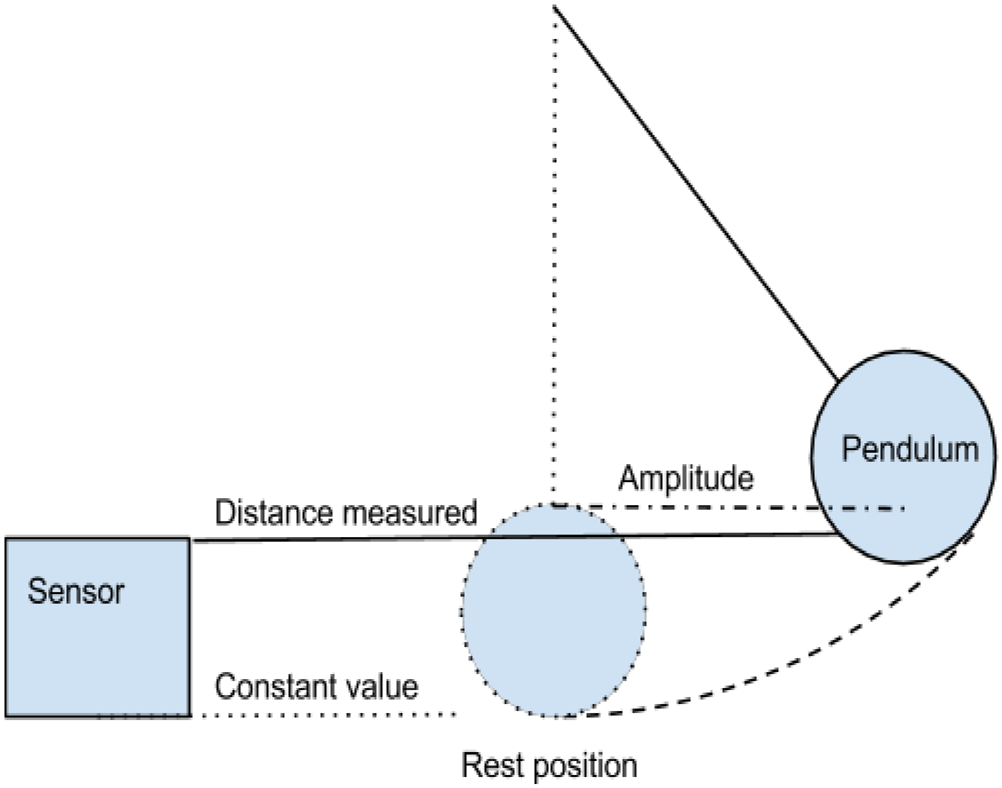
Figure 1.1.2 Example Data Mode 2; left column=distance(mm), right column=time(ms)

Read PIC | Open | Clear | Save

87	1143
121	497
448	479
109	465
159	555
311	400
2636	426
454	453
337	529
232	530
139	433
105	2579
88	610
90	113
86	109
90	96
86	95
86	101
90	3115
202	852
336	544
453	529
162	1296
57	464
50	353
170	4769

In terms of the oscillation of a pendulum, this distance will be the amplitude of oscillation, plus some constant distance (see figure 1.2.3). For the intended purposes of this mode of operation, it is not necessary to find the actual amplitude. It is only necessary to show that the amplitude may have changes between data points.

Figure 1.2.3 Constant Value Explanation



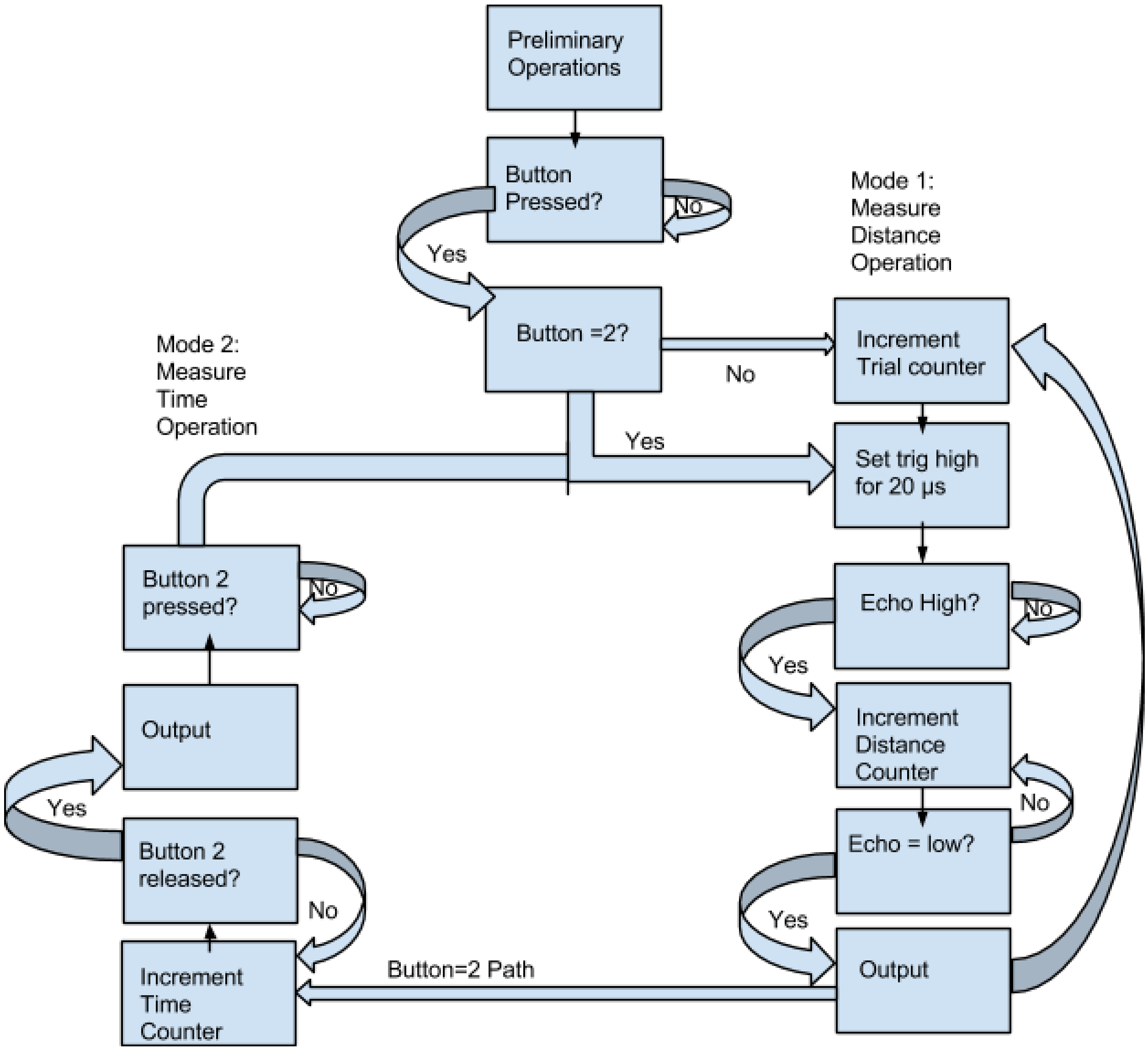
N.B Angle in this diagram is exaggerated.

The device can model the equation **x(t)=A*sin(ωt + φ) +C** i.e., the position equation for SHM; which is an important model in physics. The data has a small error, but not negligibly small. It also shows how much a simple measurement, such as the position, can do i.e., show conservation of energy, calculate amplitude, determine v(t), a(t), etc. This is important to those just introduced to physics, as it has a stigma of being difficult; in reality most things are fundamentally simple. The data output of the rangefinder synergies well with Physica. The device is also simple to use, having only a few operation buttons. For these reasons, this device makes an ideal educational tool for introductory mechanical physics.

1.2: The Program:

The program can be explained with the flowchart, figure 1.2.1. This device operates

Figure 1.2.1 Program Flowchart



N.B some checking and clearing steps are omitted for compactness.

The entire program appears as:

```
Rangefinder
;Effective angle 30 degrees
;Distance between 2cm-4m
;First, create a pulse


TRIG   equ     1; Define Trigger and
ECHO equu     2; Echo channels incase
        ; they change on board

Selector1       equ     0x26
THold  equ      0x20
EHold  equ      0x21
ECount1         equ     0x22
ECount2         equ     0x23
Graph1          equ     0x24
Graph2          equ     0x25
TimerC1         equ     0x27
TimerC2         equ     0x28

Bank1           ;Setting the Trigger
bcf     TRISE,TRIG ; set channel trigger to an output
bsf     TRISE,ECHO ; set channel Echo to input
Bank0; end of preliminary operations

Selector        call    Getkey; to select timer or distance
        movwf Selector1
        movf    Selector1,W
        sublw  7
        btfsc   STATUS,Z
        goto    Selector
        movf    Selector1,W
        sublw  2;                       pushing 2 selects timer
        btfsc   STATUS,Z
        goto    Timer


clrf    Graph1 ; clearing trial counters
clrf    Graph2
```

```
;mode 1
Triggermovlw 250 ; This is for points per seconds, It is just to delay between data points
        call    Wait


        incf    Graph1 ; increment LSD counter
        btfsc   STATUS,Z ;  if LSD = 0, increment
        incf    Graph2 ; MSD counter
        movf    Graph1 W
        iorwf   Graph2 W ; Test for 0 state, omitted from flowchart
        btfsc   STATUS,Z
        goto    END; end operation if 0 state
        movf    Graph1 W
        movwf WL      ; move counter 1 value to WL register
        movf   Graph2W
        movwf WH      ; move counter 2 value to WH register
        call    Bin2BCD
        movlw 4; this selects the amount of trials possible
        call    BCD2TCL
        movlw 0x20
        call    TxByte; at this point, the output is ready for a distance measurement


        bsf     PORTE,TRIG ; ;Need to set Trigger high for ~20us
        movlw 32 ; Delay device
        movwf THold

TWait   decfsz THold; to delay signal deactivation
        goto    TWait

        clrf    ECount1; clear counters
        clrf    ECount2
        bcf     PORTE,TRIG; Set trigger output low

;For Echo, we want distance to be in reasonable units
;As distance = 340/2 *t, if we want dist. in mm then t= 5.8823us
;therefore we want Echo clock to take 5880ns per cycle
;but cycles only in intervals of 200ns, so 5800ns
```

```
TIH      movlw 3; this is to test if the echo input is high from the sensor
         movwf EHold
DELAYdecfsz EHold
         goto    DELAY
         nop
         btfss   PORTE,ECHO; Test if high
         goto    TIH


;Distance counter begins now
Echo     movlw 5 ; For delay to make 5800ns
         movwf EHold; to store delay value
         nop; for delay
         incf    ECount1 ; increment LSD counter
         btfsc   STATUS,Z ;  if LSD = 0, increment MSD counter
         incf    ECount2 ; MSD counter
         movf    ECount1 W
         iorwf   ECount2 W ; Test for 0 state
         btfsc   STATUS,Z
         goto    END; end if 0 state


EWait    decfsz EHold ;
         goto    EWait ; for delay
         nop
         btfss   PORTE,ECHO; Test if Echo is high, skip if yes
         goto    Output ; go to output if ECHO is low
         goto    Echo ; go to counter function 'Echo' if Echo input is high

Output movf      ECount1 W
         movwf WL       ; move counter 1 value to WL register
         movf    ECount2         W
         movwf WH       ; move counter 2 value to WH register
         call    Bin2BCD
         movlw 5
         call    BCD2TCL
         movlw '\n
         call    TxByte; new line, output ready for new trial
         goto    Trigger; loop to trial counter

; end of mode 1
```

```
;mode 2
Timer   call    Getkey; test to see if 2 is pressed
        sublw   2
        btfss   STATUS,Z
        goto Timer
        clrf    TimerC1; clearing counters of data
        clrf    TimerC2;

; This is a repeat of the above distance counter
        bsf     PORTE,TRIG ; Set trigger output high
        movlw 32 ; Delay device
        movwf THold

TWW decfsz  THold; to delay signal deactivation
        goto    TWW

        clrf    ECount1; clear counters
        clrf    ECount2
        bcf     PORTE,TRIG; Set trigger output low
;total execution time between bsf and bcf is 10us. so x = (10000-(5*200))/(200*3)=15

;For Echo, we want distance to be in reasonable units
;As distance = 340/2 *t, if we want dist. in mm then t= 5.8823us
;therefore we want Echo clock to take 5880ns per cycle
;but cycles only in intervals of 200ns, so 5800ns

DLAY  movlw 3
        movwf EHold
DELAY1          decfsz EHold
        goto    DELAY1
        btfss   PORTE,ECHO; Test if high
        goto    DLAY



Echo1 movlw 5 ; For delay
        movwf EHold; to store delay value
        nop
        incf    ECount1 ; incriment LSD counter
        btfsc   STATUS,Z ;  if LSD = 0, incriment
        incf    ECount2 ; MSD counter
        movf    ECount1 W
```

```
        iorwf   ECount2 W ; Test for 0 state
        btfsc   STATUS,Z
        goto    END; error if 0 state


EWait1          decfsz EHold ;
        goto    EWait1 ; for delay
        nop
        btfss   PORTE,ECHO; Test if Echo is high, skip if yes
        goto    Output1 ; go to output if ECHO is set
        goto    Echo1 ; go to counter function 'Echo' if ECHO is low  ;14

Output1         movf    ECount1 W
        movwf WL        ; move counter 1 value to WL register
        movf    ECount2         W
        movwf WH        ; move counter 2 value to WH register
        call    Bin2BCD
        movlw 5
        call    BCD2TCL
        movlw 0x20
        call    TxByte; move amplitude to graphing output
        goto    TTT

TTT     call    Getkey; test to see if 2 is released, stop counter if 2 released
        sublw   7
        btfsc   STATUS,Z
        goto    Output2; if yes output time
        incf    TimerC1 ; incriment LSD counter 6
        btfsc   STATUS,Z ;  if LSD = 0, incriment
        incf    TimerC2 ; MSD counter
        movf    TimerC1 W
        iorwf   TimerC2 W ; Test for 0 state
        btfsc   STATUS,Z
        goto    END; error if 0 state
        nop ; for time delay
        movlw 89
        movwf THold; 15 instruction cycles
; We want nice time units 1ms,(16+ 89*56)200ns =5(200)us=1ms

TW      movlw 17
        movwf EHold
TWT     decfsz EHold
        goto    TWT;
```

```
        nop; for missing goto cycle
        decfsz THold
        goto    TW; one TW loop is 17*3 + 5 =56 instruction cycles
        goto TTT; on last instruction, additional cycle here so 16 +89*56 cycles

Output2     movf    TimerC1 W
        movwf WL      ; move counter 1 value to WL registar
        movf    TimerC2       W
        movwf WH      ; move counter 2 value to WH registar
        call    Bin2BCD
        movlw 5
        call    BCD2TCL
        movlw '\n
        call    TxByte; ready for new data
        goto    Timer; loop to timer
;end of mode two

END     return; for ending purposes
```

1.3: User Manual:

1. load the program into the Picboard
2. Select a mode of operation. Switch 2 selects mode 2, any other button selects mode 1
3. If mode one is selected, press the reset button when the desired amount of trials are conducted and/or before the upper limit of trials is reached.
4. If mode 2 selected, hold down button 2 until desired time period is achieved, then release button 2.
5. Press button 2 again and repeat instruction 4 to conduct another trial. Press the reset button when the desired amount of trials have been conducted.

*Device best operated within 30 degrees, operational range 2-400cm. For best performance, an object of area at least 0.5m^2, not made of an acoustically absorbent material, should be used for detection.

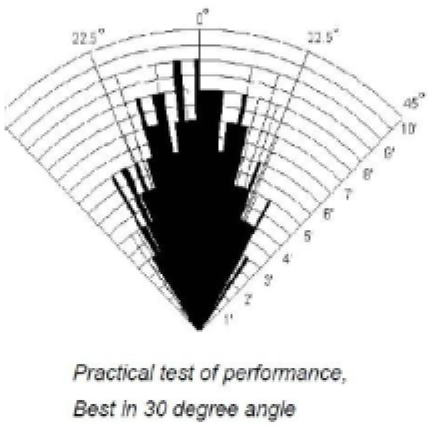
*Taken from Cytron Technologies: *HC-SR04 User's_Manual*

1.4: Device Properties and Limitations

When tested on a reasonably sized object(a picboard), the operational range of the rangefinder is greatly reduced. An object of similar size to a picboard should surpass no more than 50cm from the rangefinder and should not come closer than 2cm.

From the manufacturer, the performance vs angle measurement is shown in figure 1.4.1. It can be said that the device operates most effectively within 15^{o} from the axis of measurement.

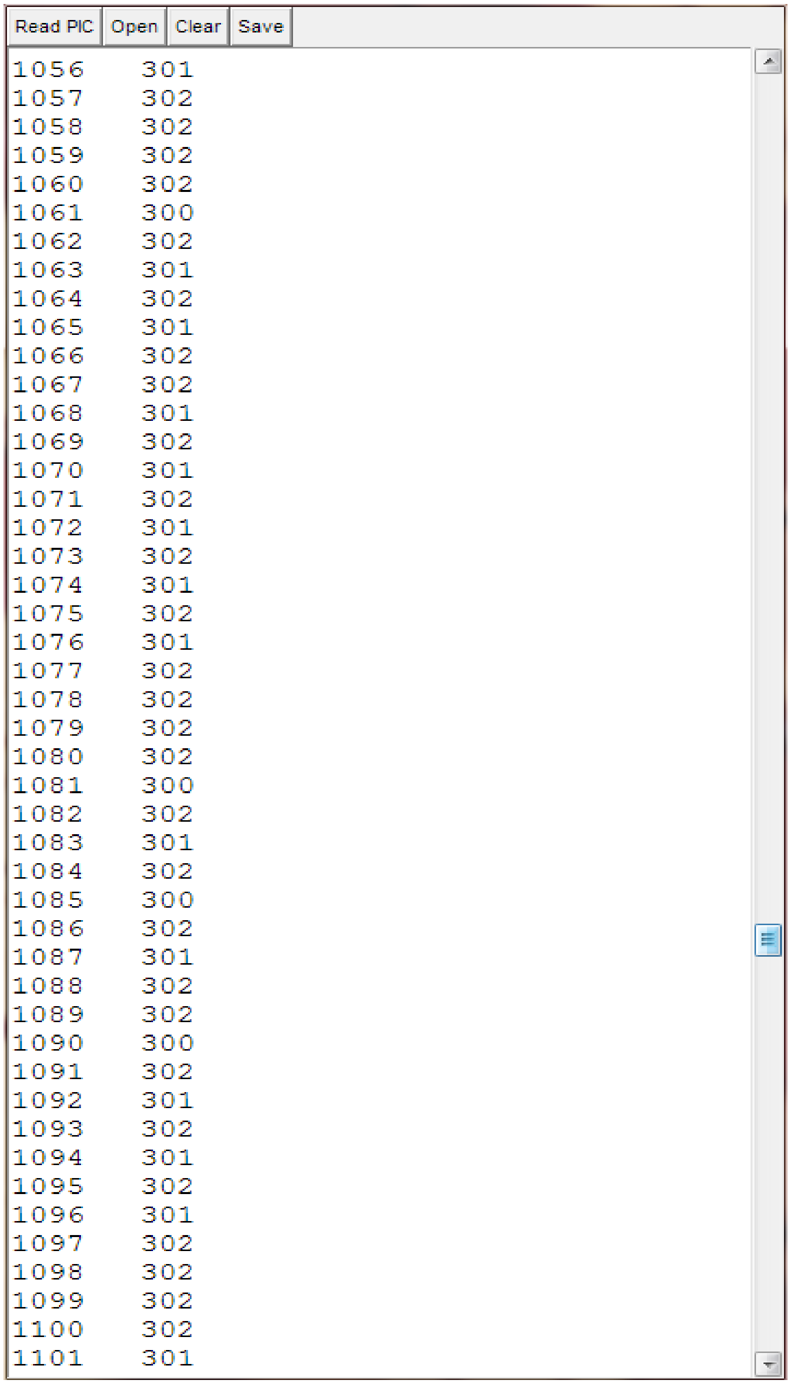
Figure 1.4.1 Performance vs Angle



Taken from Cytron Technologies: *HC-SR04 User's_Manual*

The precision and accuracy of distance measurements is quite high, typically varying ±1mm, as shown in figure 1.4.2. The measurement listed in figure 1.4.2 is of a distance ≅30cm. Due to disciples in the origin i.e., it is difficult to determine the actual location of the sensor, all that may be said of the accuracy of the device is that it is accurate to within 5mm w.r.t. the picboard. When analysing the accuracy of the device, the resolution of the counter should be considered. The test, for the echo output to go high, has an associated resolution of 13 instruction cycles. Also, the counter itself has an associated resolution of 29 instruction cycles. This creates an error of about ± 1.428mm($(13+29)*200*10^{-9}*340/2$). In addition, the device has associated errors with the speed of sound in air, as this is variable and dependant on environment. Additionally, there is a rounding error associated with the counter cycle. The actual period of the cycle should be 5.8823μs ($d=v/2*t \rightarrow t=2*10^{-3}/340$). This implies that the counter has an overestimation of about 1.42%(5.8823/5.8).

Figure 1.4.2: PICLab Output Data at 30cm



1056	301
1057	302
1058	302
1059	302
1060	302
1061	300
1062	302
1063	301
1064	302
1065	301
1066	302
1067	302
1068	301
1069	302
1070	301
1071	302
1072	301
1073	302
1074	301
1075	302
1076	301
1077	302
1078	302
1079	302
1080	302
1081	300
1082	302
1083	301
1084	302
1085	300
1086	302
1087	301
1088	302
1089	302
1090	300
1091	302
1092	301
1093	302
1094	301
1095	302
1096	301
1097	302
1098	302
1099	302
1100	302
1101	301

The time resolution of the device, in the distance measuring mode, is variable and may be approximated by 38.75ms +(5.8μs*distance). The constant term is due to the wait function applied at the beginning of the program. The variable term is the time required to count to a specific distance. This gives a minimum time resolution of about 38.9ms and a maximum of 0.419s.

It should be noted that the maximum distance, time and trial values which may be obtained is 2^{16}-1. However, for the applicational purposes of the device, these values would never be obtained. In practice, the distance values should never surpass 4m and 9999 trials is sufficient for any sized pendulum or oscillator.

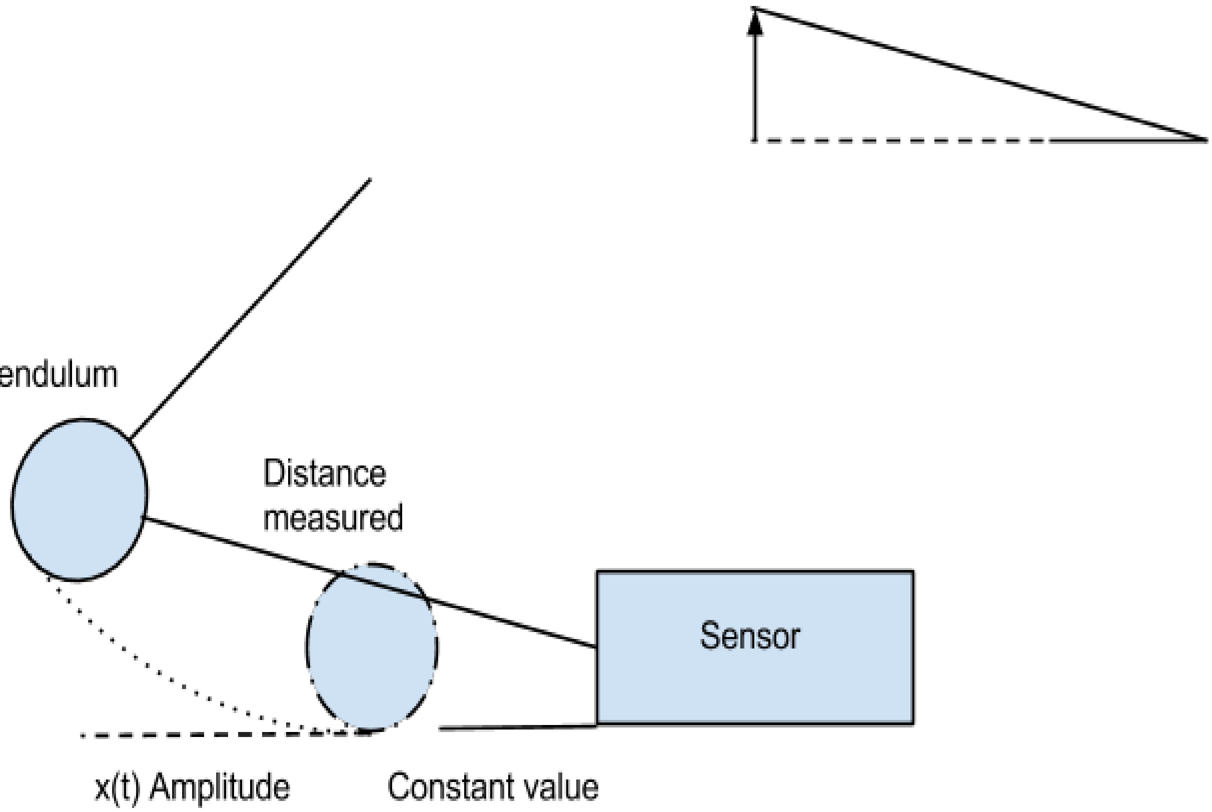
The accuracy and precision of mode 2, the timing mode, is not determinable; due to the dependance of a human input(pressing and releasing the button). Thus, it is not viable to use the measurements obtained for qualitative purposes. The feature was included only to show the independance of period from amplitude for the linear pendulum(i.e., to be able to change the amplitude in the same data set). To calculate the actual period, it is advised that one uses mode 1 to acquire distance data and utilize a graphing program capable of fitting functions to data(e.g., Physica).

1.5: Pendulum Demo:

The equation of a linear pendulum, $\mathbf{x(t)=A*sin(\omega t + \varphi) +C}$, is only applied for linear pendulums. For the purpose of a demonstration of mode one, small angle approximation will be used; so that the data will approximate a linear pendulum well. In addition, if the amplitude of the pendulum is too large, the rangefinder will not be able to detect it, as it will pass the optimal angle threshold.

The period of a linear pendulum can be found with the following equation: $\mathbf{T=2\pi(I/mgL)^{1/2}}$. in the case of a simple pendulum, this reduces to $\mathbf{T=2\pi(L/g)^{0.5}}$. The period of a non-linear, simple pendulum is $\mathbf{T=2\pi(L/g)^{0.5}sin(\theta)}$(Period equation taken from Georgia State University, HyperPhysics). This implies the period should remain constant, within experimental error, until a sufficient amplitude is exceeded. This will be tested by the second mode of operation. Note that the amplitude values obtained in mode two will not reflect actual amplitude values. This is especially true once the amplitude begins to affect the period, as this is when the small angle approximation is no longer valid. This is shown in figure 1.5.1. As the sum of the vectors in not zero, the amplitude measurements would be off if the constant value method was used, hence the data the rangefinder gives is a magnitude only.

Figure 1.5.1 The Effect of Exceeding Small Angles on Amplitude



<u>1.6: References:</u>

Cytron Technologies: *HC-SR04 User's_Manual,* *https://docs.google.com/document/d/1Y-yZnNhMYy7rwhAgyL_pfa39RsB-x2qR4vP8saG73rE/edit?pli=1*

Georgia State University, HyperPhysics, *Mechanics: Periodic Motion;* *http://hyperphysics.phy-astr.gsu.edu/hbase/permot.html#permot*