`Ultra Sonic HC SR04 LCD Tape Measure

`Pat McMahon 18/2/2022

setfreq m4

symbol trig = **5** ' Define output pin for Trigger pulse

symbol echo = c.0 ' Define input pin for Echo pulse

symbol range = b1 ' 16 bit variable for range (remember w1 = b2 + b3)

main:

 ` pulsout trig,2 ' produces about 20uS pulse (must be minimum of 10uS)

 `pulsin echo,1,b1 ' measures the range in 10uS steps

 ` pause 10 ' HC-SR04 mandatory 10mS recharge period after ranging completes

' now convert range to cm (divide by 5.8) or inches (divide by 14.9)

' as picaxe cannot use fractions, multiply by 10 then divide by full number

` Also divide range time by 2 because of time of trig out then bounce echo back to sensor.

' e.g. range/2 x 10 then / 58 is the same as range / 5.8

 let b1 = b1/ **2** \* **10** / **58**

 sertxd ("range ",#range," Cm ",cr,lf)

 pulsout **5**,**2** ' produces about 20uS pulse (must be minimum of 10uS)

 pulsin c.0,**1**,b1 ' measures the range in 10uS steps

 pause **10** ' HC-SR04 mandatory 10mS recharge period after ranging completes

 let b1 = b1 \* **10** /**58**' multiply by 10 then divide by 58 for 4meg or 28 for 8meg

 debug

 pause **1000**

 serout **1**,N2400,(**254**,**1**) ; Clears OLED display

 pause **30**

 serout **1**,N2400,(**254**,**128**,"Distance out "); Displays the "Distance Out" words on the 1st line of OLED

 serout **1**,N2400,(**254**,**192**,"=",#b1,"cm "); Displays the value of the distance on the 2nd line of OLED

 pause **300**

 goto main ; Repeats

if b1>**31** then goforwards

if b1<=**30** then gobackwards

goto main

goforwards:

high **3**,**4**

low **1**,**2**

pause **500**

low **3**,**4**

pause **100**

goto main

gobackwards:

high **1**,**2**

low **3**,**4**

pause **2000**

high **3** `go right

low **1**,**2**,**4**

pause **2000**

goto main