

# Negative mass as a cause of dark matter and dark energy

*Austin J. Fearnley  
Manchester  
England  
14 July 2018*

## Abstract

Simulations using Visual Basic code are carried out on motions, along a one dimensional line, of a population of 100 positive and negative masses under their own gravitational forces. The negative masses were seen to repel one another along the line and uniformly occupy the line. The positive masses were initially located as two static clumps and they remained as two clumps without greatly expanding the sizes of the clumps. The clumps separated further apart, in time, accelerating away from each other. Some of the negative masses clustered near the two positive mass clumps, in a similar way to how dark matter clusters near galaxies. The negative masses act to constrain a positive mass clump by surrounding it and repelling the masses back towards the centre of the clump. So negative mass operates by repulsion to obtain a similar effect that dark matter is supposed to obtain by attraction. Replenishment of negative masses as they fill the space along the line is unnecessary as the results show continuous acceleration.

## Introduction

Negative mass is an exotic concept. A vixra paper by Hyoyoung Choi (2009) explores the possibility that negative mass causes both dark matter and dark energy.

The equations governing the motions of positive and negative masses under gravitational forces are described in detail in Choi (2009) and the oddity of runaway motion is highlighted in Wikipedia (2018):

“Positive mass attracts both other positive masses and negative masses.

Negative mass repels both other negative masses and positive masses.

For two positive masses, nothing changes and there is a gravitational pull on each other causing an attraction. Two negative masses would repel because of their negative inertial masses. For different signs however, there is a push that repels the positive mass from the negative mass, and a pull that attracts the negative mass towards the positive one at the same time.

Hence Bondi pointed out that two objects of equal and opposite mass would produce a constant acceleration of the system towards the positive-mass object, an effect called "runaway motion" ...”.

Runaway motion as an effect, however, seems to be desirable rather than anathema for dark energy effects. Detailed illustrations, formulae and calculations for the motions of positive and negative masses under gravitational attraction are shown in Appendix I, paying particular attention to the directions of the accelerations. The formulae for calculations in Appendix I underpin the software code in Appendix B.

This paper simulates the effects of negative mass, whether that could possibly and speculatively be neutrino mass or some unknown source of negative mass, in order to try to obtain results consistent with the effects of dark matter and dark energy. The simulation appears to have been successful and was carried out using Visual Basic code in Microsoft Excel: the code is given in Appendix II.

## Method

Some 100 test particles are used in the simulations. Of these, 50 have positive mass (+1 unit of mass) and 50 have negative mass (-1 unit of mass). Time was counted by the number of cycles that the code used: each cycle of software code counted as one tick of the clock. The starting velocities of all 100 masses were set to zero. The negative masses were initially positioned at random using a uniform distribution between positions -0.5 and +0.5 on a 1D line. The positive masses were divided into two clumps of 25 masses each. One clump was formed of 25 masses initially placed at random using a uniform distribution between positions 1 and 2 on the 1D line. The second clump of positive masses was formed of 25 masses initially placed at random using a uniform distribution between positions -1 and -2 on the 1D line.

During one tick of the clock, that is one cycle through the software code, one mass is picked out in turn to have its acceleration calculated: this is mass  $i$ . For this individual mass, the gravitational effects on it of the other 99 masses (that is the mass  $j$ ) are calculated taking great care to obtain correct directions of the 99 different accelerations acting on mass  $j$  (see Appendix I). In unit time, that is in one cycle of software code, the formula 'newspeed = oldspeed + acceleration \* time' becomes:

newspeed = oldspeed + acceleration, where time = unity.

To get the new position of mass  $i$ , the average speed of mass  $i$  during the unit time interval is:

average speed =  $0.5 * (\text{newspeed} - \text{oldspeed}) = 0.5 * \text{acceleration}$ .

Therefore, the newposition of mass  $i$  = oldposition of mass  $i$  +  $0.5 * \text{acceleration}$ , after a unit time interval.

In the software code, a scaling factor of 0.01 is used in the calculation of the acceleration merely to obtain a reasonable path through the space of the 1D line in a scatterplot. This scaling factor is usable because the units of the line metric are arbitrary.

These calculations were executed for each mass  $i$  in turn and the new positions of all 100 masses were found. This completes one cycle or tick of the clock and was repeated for various numbers of ticks or cycles and results were plotted on the Excel spreadsheet as: index  $i$  of mass (from 1 to 100); value of mass  $i$ , position of mass  $i$ , time elapsed (or number of cycles of the software code).

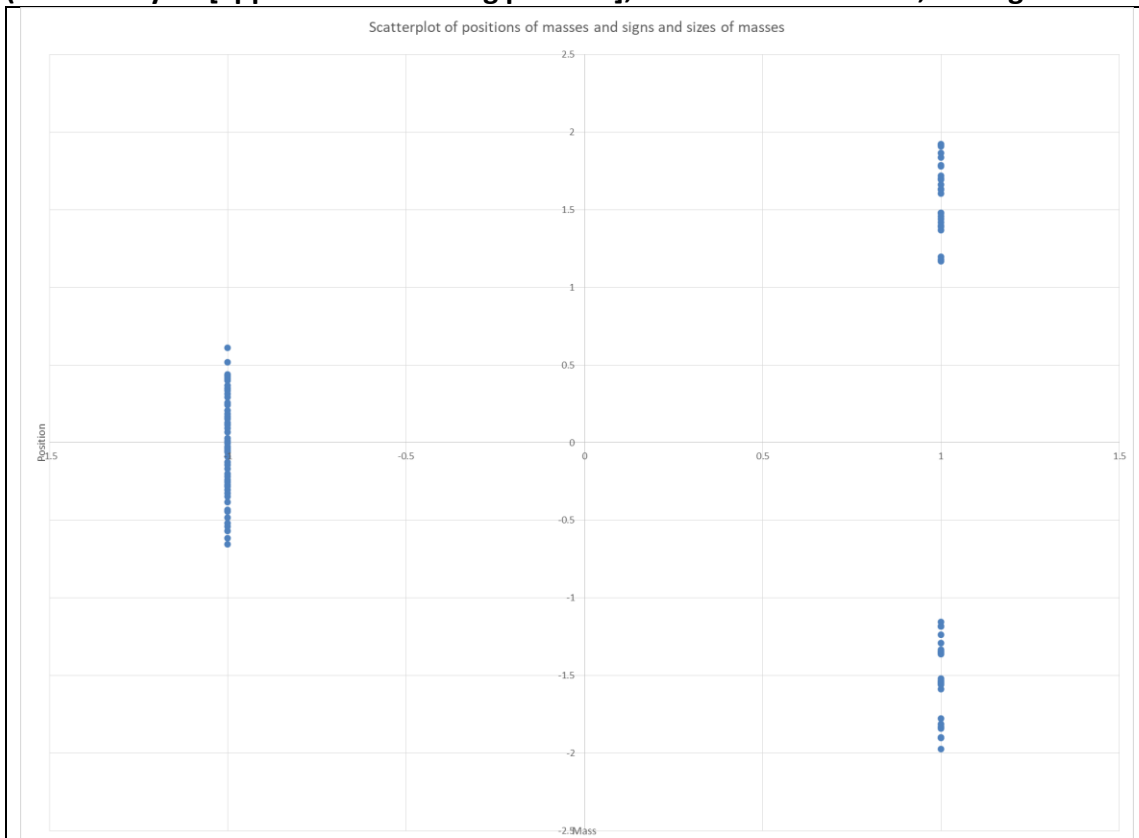
A scatterplot of position against mass for the 100 masses shows the distributions and clumps for positive and negative masses separately and can show negative masses clustering around positive mass clumps and the self-repulsion of the negative masses throughout the space of the 1D line.

## Results

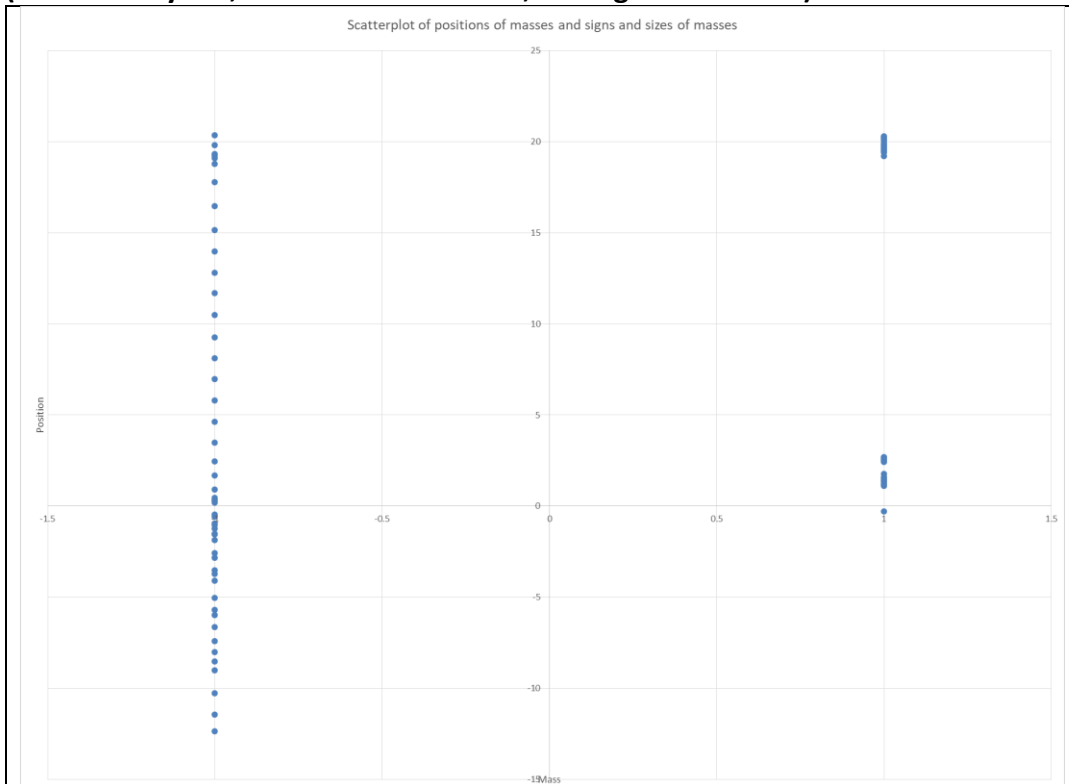
The method given in the previous section was executed and the resulting scatterplots of mass against position were plotted for times elapsed as 1, 20, 40 and 60 units of time in Figures A to D, respectively. Figure A shows positions of masses very close to the starting positions, after only one cycle.

The data in any figure are independent of the data in the other figures, as the software code was not written to pause and display results in a freeze-frame progression. Nevertheless, similar results were obtained in replications of runs. It is clear that the negative masses cluster about the two clumps of positive mass (especially Figures B and C). This shows the negative masses acting like dark matter. Dark matter is regarded as positive mass hidden amongst and extending beyond our (say) galaxy. The negative masses in Figures B and C cannot act exactly in that way. Instead, the negative mass surrounds a clump of positive mass and repels it from expanding outwards. The scale of the position axis changes from figure to figure so the shrinking of clump size is apparent but not real, and clumps retain the same approximate order of magnitude of one unit of length. So, the clumps are not expanding nor shrinking in size.

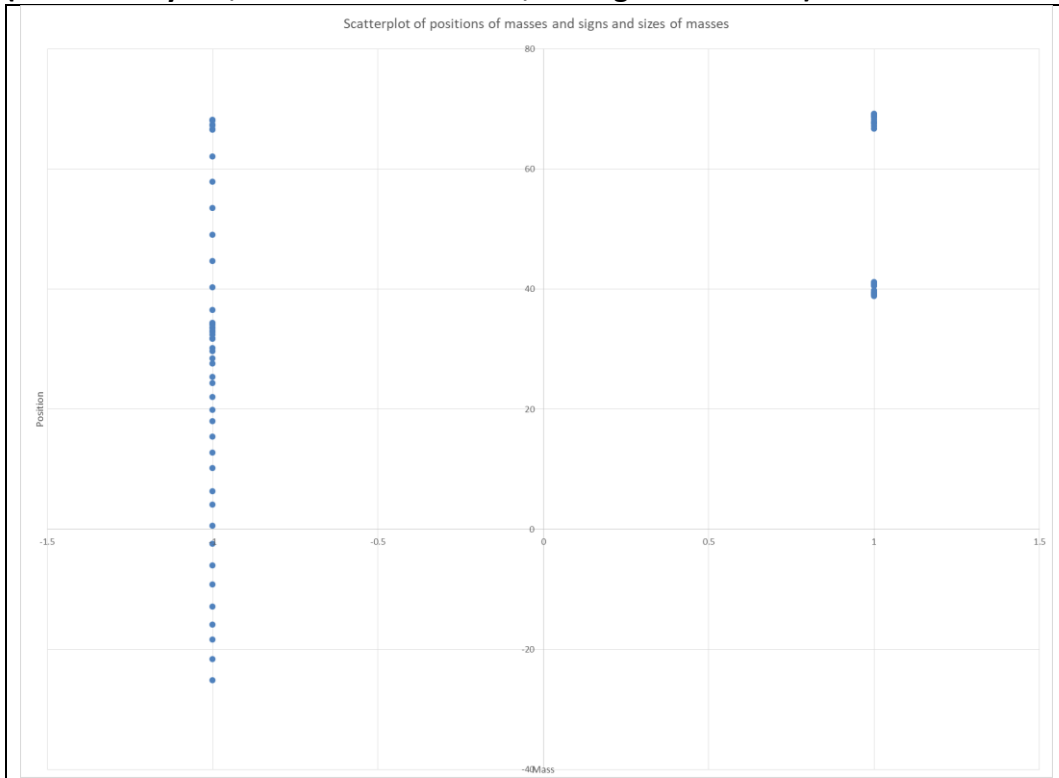
**Figure A: Scatterplot of 100 masses against position**  
(Time = 1 cycle [approximate starting position]; masses are -1 and +1; scaling factor = 0.01)



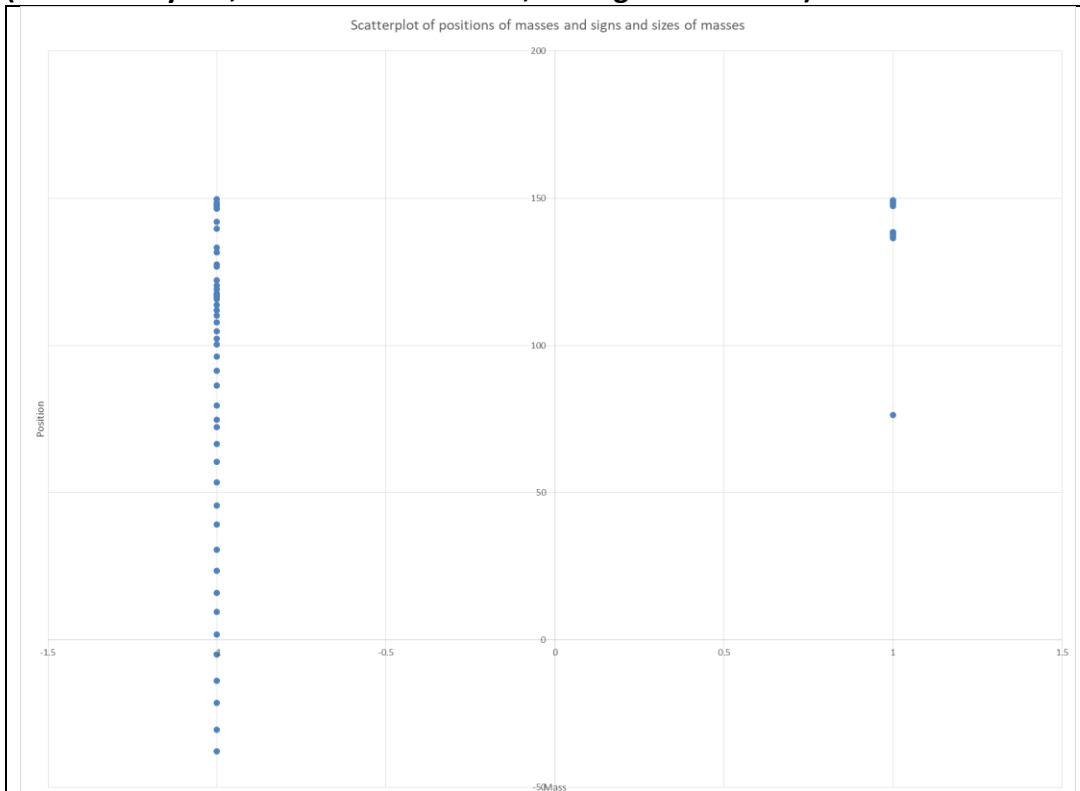
**Figure B: Scatterplot of 100 masses against position**  
(Time = 20 cycles; masses are -1 and +1; scaling factor = 0.01)



**Figure C: Scatterplot of 100 masses against position**  
(Time = 40 cycles; masses are -1 and +1; scaling factor = 0.01)



**Figure D: Scatterplot of 100 masses against position**  
(Time = 60 cycles; masses are -1 and +1; scaling factor = 0.01)



The clumps are however being separated in space from Figures A to C, as is also summarised in Table 1. From time 1 to 20, and again from time 20 to 40, the two clumps of positive matter increase in separation, which is analogous to a dark energy effect. The effect is lost for Figure D but it should be noted that in Figure D the positive matter has run away from most of the negative matter. If the simulations had been more advanced than a simple 1D toy model, then Figure D could indicate that dark energy could eventually cease and the universe could enter a later phase without dark energy where the positive and negative masses have separated out.

**Table 1: Summary of results taken from Figures A to D**

Time, measured in cycles of code	Position of Clump 1 of positive masses	Position of Clump 2 of positive masses	Separation of Clumps 1 and 2	Increase in separation of Clumps 1 and 2
1	1 to 2	-1 to -2	3	n.a.
20	19 to 21	2 to 3	18	15
40	69 to 72	29 to 42	40	22
60	148 to 150	138 to 140	10	negative

n.a. = not applicable

The toy model, however, has some drawbacks. The effects of gravity are assumed to take effect instantaneously; also the calculations are non-relativistic which is unrealistic if the masses are of elementary particles and also unrealistic if the positive masses are far-flung galaxies. Also, non-relativistic calculations are unrealistic if the negative masses are say neutrinos which travel approximately at speed  $c$ . If the negative masses are (say) neutrinos travelling at about speed  $c$  then they should not be outrun by positive masses at the top of Figure D. So Figure D may be unrealistic in the ease at which positive and negative masses can be separated and dark energy effects switched off. Other effects are not modelled here, such as rotations. Other forces are ignored such as electromagnetic forces keeping small clumps of positive masses together.

Merely because of the supposed need for constant replenishment of dark energy material, it is suggested here that dark energy effects may vary over time according to the generation of elementary particle acting as the main player with negative mass. For example say the baton is passed on from tau neutrino to muon neutrino to electron neutrino in successive epochs. If Figure C were to be re-run with masses of +4 and -4, instead of +1 and -1, corresponding to higher masses for higher generations of particles, then {a test run showed that} the two positive mass clumps would have an average position at 250 units after 40 cycles compared to the average position of 50 units from the origin after 40 cycles in Figure C. So higher generations give more acceleration. However, even lower generations give acceleration in Figures A to D until the two types of mass have separated out. My preon model (Fearnley, 2015) shows that higher generations of particles contain more preons than lower generations. That means, because preon numbers are conserved in interactions, and particles in my model

are made from preons rather than from insubstantial energy, that greater numbers of lower-generation particles can be created out of the same preons constituting a smaller number of higher-generation particles, so that a constant density can be maintained by generational decay creating more particles. Figures A to D show, however, that the generation of new dark energy material is not necessarily required to maintain acceleration of separation of positive mass clumps in the circumstances when negative masses are modelled as the cause of dark energy effects.

## References

Fearnley, A.J. (November 2015) "Hexark and Preon Model #7: for Standard Model Elementary Particles, Higgs, Gravitons, Dark Energy and Dark Matter".

<http://vixra.org/abs/1511.0115>

Hyoyoung Choi (2009) "Hypothesis of Dark Matter and Dark Energy with Negative Mass".

<http://vixra.org/abs/0907.0015>

Wikipedia (2018) Negative mass. [https://en.wikipedia.org/wiki/Negative\\_mass](https://en.wikipedia.org/wiki/Negative_mass).

14 July 2018  
Manchester  
England

## APPENDIX I

### Directions of accelerations of positive and negative masses under gravitational forces

The masses lie on a 1D line which has a positive direction of position given by a Cartesian coordinate (either the x or y coordinate: x in this appendix, for convenience of illustration, but y in Figures A to D in the Results section). There are 100 masses in the simulation and the acceleration effects are calculated on each mass in turn based on the accumulated effects on it of the other 99 masses.

The mass under measurement is mass i while the other 99 masses are masses j. It is assumed here that mass i is on the positive x axis, that is, mass i is positioned to the right of the origin. Mass j can be placed either to the right or to the left of mass i. Both masses can be either of positive or negative mass so that there are  $2*2*2 = 8$  orientations of mass i and mass j and these are examined in the following table.

Origin	Mass j	Mass i	Mass j	Direction of acceleration of mass i ( → is + and ← is -)	Sign of position of mass j minus position of mass i = r	Formula for direction of acceleration of mass i = r * sign of mass j
Origin		+1	+1	→	+	+ * + = +
Origin	+1	+1		←	-	- * + = -
Origin		+1	-1	←	+	+ * - = -
Origin	-1	+1		→	-	- * - = +
Origin		-1	+1	→	+	+ * + = +
Origin	+1	-1		←	-	- * + = -
Origin		-1	-1	←	+	+ * - = -
Origin	-1	-1		→	-	- * - = +

Magnitude of acceleration of mass i = force on mass i / mass i = mass j / (separation of masses)<sup>2</sup>

The direction of the acceleration is given by r \* sign of mass j from the columns of the above table. That sign is given by the sign of the term:

(position of mass j – position of mass i) \* mass j.



## APPENDIX II

### Excel Visual Basic software code used in the simulations

```

Sub Button1_Click()
'
' Excel VB program to simulate dark matter clustering around galaxies and dark energy accelerating galaxies
'
Dim signofmass(100) As Integer ' mass values as +1 or -1
Dim oldspeed(100) As Double ' initial speed values
Dim newspeed(100) As Double ' final speed values
Dim oldposition(100) As Double ' initial position values
Dim newposition(100) As Double ' final position values
Dim cycles, i, j, k, time, ii As Integer
Dim sumofspeeds, r, r2, r0, accel As Double ' r is separation between two masses
' sumofspeeds is used in calculation of new positons

Range("c3").Select
Randomize ' Initialize random-number generator

For i = 1 To 25 ' use the random number generator to pick
r0 = Rnd ' a number between 0 and 1
oldposition(i) = r0 + 1 ' a random number between 1 and 2
Next i

For i = 26 To 50 ' use the random number generator to pick
r0 = -Rnd ' a number between 0 and -1
oldposition(i) = r0 - 1 ' a random number between -1 and -2
Next i

For i = 51 To 100 ' use the random number generator to pick
r0 = Rnd ' a number between 0 and 1
oldposition(i) = r0 - 0.5 ' a random number between -0.5 and 0.5
Next i

For i = 1 To 100 ' initialise all starting speeds of masses as zero
oldspeed(i) = 0#
Next i

For i = 1 To 50 ' initialise 50 masses as +1 mass
signofmass(i) = 4 ' *****
Next i

For i = 51 To 100 ' initialise 50 masses as -1 mass
signofmass(i) = -4 ' *****
Next i

cycles = 40 ' number of ticks of the clock *****
time = 0 ' time is the tick of a clock with new positions after every tick

' start of main routine
For k = 1 To cycles ' index for clock ticks
For i = 1 To 100 ' i denotes the mass under calculation
sumofspeeds = 0#
For j = 1 To 100 ' j is all the other 99 masses which can affect the motion of the mass under calculation
If i = j Then GoTo 5000 ' need to exclude the gravitation effect of the mass upon itself

runitvector = (oldposition(j) - oldposition(i)) / (Abs(oldposition(j) - oldposition(i)))
r = (oldposition(j) - oldposition(i))
If r < 0.1 Then r = 0.1 ' cannot let two masses be too close so as to avoid singularity effects
r2 = r * r
accel = 0.01 * runitvector * signofmass(j) / r2 ' acceleration = force / mass *****
' 0.01 is an arbitrary scaling factor

newspeed(i) = oldspeed(i) + accel
sumofspeeds = sumofspeeds + newspeed(i) ' this increments speed added to mass i by effect of mass j

5000 Next j ' go to next other mass
newspeed(i) = sumofspeeds / 99# ' using an average rather than a total will slow down size of effect
per clock tick
Next i ' go to next mass under calculation
time = time + 1 ' keeps track of how many clock ticks there have been

For ii = 1 To 100
newposition(ii) = oldposition(ii) + 0.5* newspeed(ii) ' assumes newspeed values are small
' i.e. assumes acceleration is uniform

'write results to workpage

ActiveCell.Offset(ii, 1).Formula = ii
ActiveCell.Offset(ii, 2).Formula = signofmass(ii)
ActiveCell.Offset(ii, 3).Formula = newposition(ii)
ActiveCell.Offset(ii, 4).Formula = time

```

```
oldspeed(ii) = newspeed(ii)
oldposition(ii) = newposition(ii)
Next ii

Next k      ' end of cycle
End
End Sub
```

The user needs to create a simple scatterplot graph to graph results which have been outputted to the spreadsheet.

The code was written to be operated by clicking on a control button on the spreadsheet, but that is optional.