<u>**Physics of an axial coil gun**</u>

<u>**Part I - Fundamentals**</u>

<u>**The basic configuration of a single stage coil gun**</u>

The following figure shows the key dimensions of an axial coil gun, also known as a Gaussian coil gun. The cylindrical coil of wire is a solenoid. We will describe the solenoid as having a height (or length) equal to H_{coil} on which are wound N_{turns} of copper wire. The solenoid has an inner diameter, or core diameter, equal to twice the core radius R_{core}. The outer diameter of the solenoid depends on the diameter of the copper wire D_{wire} and the number of layers N_{layers} in the winding. We will use the symbol N_{turns} for the number of turns in a single layer. If the winding has more than one layer, then the total number of turns in the winding is equal to $N_{layers} \times N_{turns}$. The solenoid is filled with air.

To describe relative positions, we will use a co-ordinate frame of reference with its origin O at the geometric center of the solenoid. The $\hat{z}$-axis will always point along the axis of the solenoid, passing through one "face" of the solenoid, at the point F. The $\hat{x}$-axis is perpendicular to the $\hat{z}$-axis. Since the whole configuration is symmetric around the $\hat{z}$-axis, we can take the $\hat{x}$-axis to be representative of any radially-directed axis.

Located somewhere along the $\hat{z}$-axis, and also symmetrical around the $\hat{z}$-axis, is a solid cylinder made of steel or some other ferromagnetic material. This is the slug we want the coil gun to fire. We will generally refer to the height (or length) of the slug as H_{slug} and to its radius as R_{slug}, but these two dimensions are not shown in the figure.
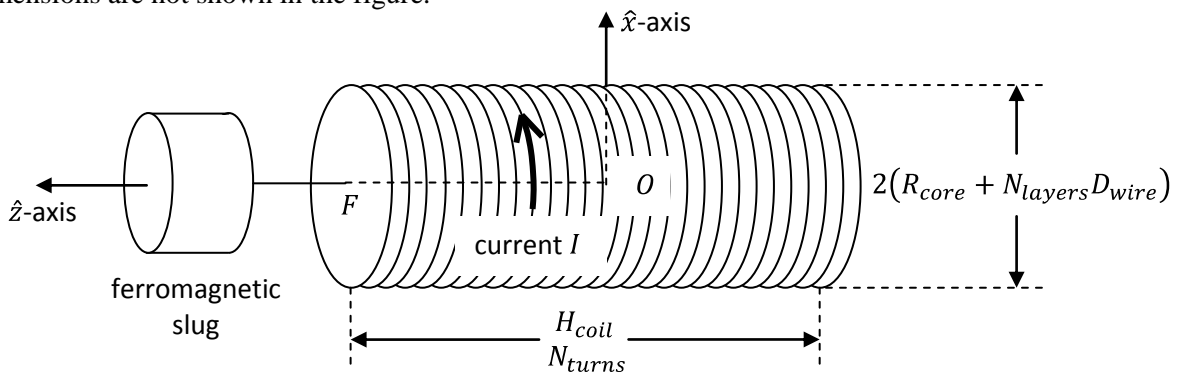
The interesting things happen when current I flows through the coil. The magnetic field generated by the flow of current gives rise to a net force acting on the slug, attracting it towards the center of the coil. It does not matter which way the current flows through the coil; the force exerted on the slug will be directed towards the center of the coil. Nor does it matter whether the slug is located somewhere along the positive $\hat{z}$-axis (as shown) or somewhere along the negative $\hat{z}$-axis. The slug will always be attracted towards the center of the coil.

If the slug is held in a fixed position, then the force will not cause any movement. Instead, the force which attracts the slug towards the coil and the equal and opposite force which attracts the coil towards the slug will be transmitted to whatever structure holds the two pieces apart. The internal stresses set up in the structure will give rise to forces which counter the attractive force between the slug and the coil.

In the usual case, the coil is held firmly in place and the slug is left free to move, or slide, along the $\hat{z}$-axis. In the co-ordinate frame of reference shown, the slug will slide in the direction of the negative $\hat{z}$-

axis. Not only will the slug slide towards the center of the coil, it will accelerate towards the center of the coil. As long as the attractive force continues, the slug will slide faster and faster. If the coil is long enough, the slug will keep accelerating until such time as the friction between the slug and its supporting rail or rails becomes so great that the frictional force just equals the attractive magnetic force, after which time the slug will slide along at a constant speed.

It is our intention to turn the current off at or before the instant when the geometric center of the slug just reaches the origin. If we let the current continue to flow after that point, then the attractive force will be in the opposite direction to the slug's direction of travel, and will start to slow it down. For a given current I, there is a maximum speed which this configuration can impart to the slug. As soon as the slug passes through the origin O, the coil cannot give it more energy. In a sense, the right-hand side of the coil in the figure above is useless. Cutting the coil in half does not help – without the right-hand side of the coil, there could be no left-hand side, just two shorter coils.

The usual objective of this configuration is to maximize the final speed of the slug. To do this, we could try to increase the current I. But, there is a limit to the current which any given power supply can deliver. In order to force larger and larger currents to flow through the coil, we could try to use bigger and bigger power supplies. But, there comes a point beyond which larger power supplies are not feasible. To get larger currents, it is necessary to stop thinking about the power supply as a continuous source of x amperes at y volts, and to start thinking about it as a source of energy which can be released in as short a period of time as possible. One such source of energy, and the one we will examine in this paper, is a charged capacitor.

Using a charged capacitor instead of a constant current supply raises a host of complicating factors to the analysis. One of those factors is the initial placement of the slug. To get the most energy from the coil, the slug needs to be at the right place when the current flow is at its maximum

In the analysis, we will assume that the slug starts from rest. Under the influence of just one coil, the slug will be given a single burst of acceleration. A device with a single coil is called a "single-stage" coil gun. It is possible, of course, to line up several coils in series, so that the slug passes through one coil after another. In such a "multiple-stage" coil gun, the slug is given an additional burst of acceleration by the left-hand side of each coil is passes through.

An introductory example

We will get our feet wet by looking at the dynamics of the slug – "dynamics" being its movement in response to a force – in three simplified cases. In this first example, we will assume that the force exerted on the slug is a constant, irrespective of its position. This is most definitely not the real case. Even if the current flowing through the coil happened to be constant, the force the magnetic field exerts on the slug varies with the slug's position. Nevertheless, this simple case will help us get some of the fundamentals right. For this example, we will assume that the total force acting on the slug has some constant magnitude F_{max} on the slug if it is located anywhere in a some horizontal range D on the active side of the coil, starting from the origin. The configuration is as shown in the following figure.

~ 2 ~
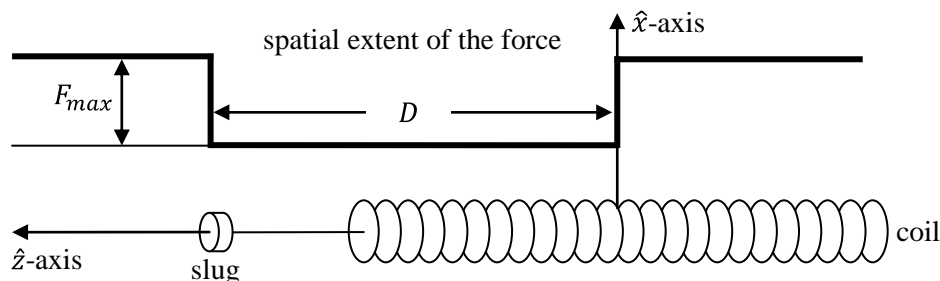
At the bottom of the figure is the physical arrangement of the slug and the coil. The assumed spatial distribution of the force is shown above that. The force "field" extends out past the end of the coil on the left-hand side. I have terminated the force field at the origin on the right-hand side, in expectation that we will turn the current off as soon as the slug reaches the center of the coil. I have parked the slug at abscissa $z = D$, where we would probably place it before starting the "run", exactly where the force field begins. Placing the slug further to the left would prevent the force from acting on it at all. Placing it further to the right would waste some of the force field – by not passing through part of the force field, the slug would not obtain as much acceleration as it could.

The force is zero outside of the range D. Readers will note that the force shown has the value $-F_{max}$ everywhere inside the range D. The minus sign is important. It states the direction in which the force is acting. The principal axis in the following analysis is the $\hat{z}$-axis, along which the slug moves. The positive $\hat{z}$-axis points towards the left in the figure. But, the attractive force exerted by the coil on the slug points towards the right, in the direction of the negative $\hat{z}$-axis. During all times of interest, the slug will be located somewhere along the positive $\hat{z}$-axis, for which the values of z are algebraically positive. But, its speed will be algebraically negative, as the slug travels and accelerates towards the right.

We will "establish" the force field at time $t = 0$, before which time the force is zero. We will assume that the force field is built up in an instant. If we ignore all forces acting on the slug other than that of the force field, or magnetic field, then the slug will accelerate towards the right. The forces we are ignoring are the friction between the slug and its rails, the retarding drag of the air on the slug once it starts moving, and so on. The acceleration of a rigid body is conveniently given by Newton's Third Law, as follows:

$$F = ma \quad (1)$$

where F is the force acting on the body, m is its mass and a is its acceleration. In general, the force and the acceleration are "vector" quantities, meaning that they are characterized not just by their magnitudes, but by their directions as well. In our case, all of the activity takes place along one line, the $\hat{z}$-axis. If the coil or the slug were not symmetric, then the slug might be subjected to some force tending to pull it away from the $\hat{z}$-axis, but we will ignore such imperfections. Since we are dealing with a single axis, the algebraic sign of a quantity like force or acceleration is enough to specify its direction: positive towards the left and negative towards the right.

Now, the acceleration of the slug is the first time derivative of its speed. In turn, the slug's speed is the first time derivative of its position. If the slug's position (let us choose to use the abscissa of slug's geometric center as the measure of its position) is given by the variable z, then its speed and acceleration are written mathematically as:

Parameter	Magnitude	As a vector
Slug's position	z	$z\hat{z}$
Slug's velocity	v or $\dfrac{dz}{dt}$	$\dfrac{dz}{dt}\hat{z}$
Slug's acceleration	a or $\dfrac{dv}{dt}$ or $\dfrac{d^2z}{dt^2}$	$\dfrac{d^2z}{dt^2}\hat{z}$ or $\dfrac{dv}{dt}\hat{z}$

We will use the symbol m for the slug's mass. And, to avoid any confusion about the matter, I will write the value of the force as $F = -|F_{max}|$. For our configuration, Newton's Law can be written in terms of the slug's speed v as follows:

$$-|F_{max}| = m\frac{dv}{dt}$$

$$\rightarrow \quad \frac{dv}{dt} = -\frac{|F_{max}|}{m} \quad (2)$$

Since the magnitude of the force F_{max} and the mass of the slug m are both constant, then the right-hand side of Equation (2) is a constant. That makes it easy to integrate the expression, which will give the slug's speed.

$$\int_{\tau=0}^{t} \frac{dv}{d\tau} d\tau = \int_{\tau=0}^{t} \left(\frac{d^2z}{d\tau^2}\right) d\tau$$

$$\rightarrow \quad \int_{\tau=0}^{t} dv = \int_{\tau=0}^{t} \left(-\frac{|F_{max}|}{m}\right) d\tau$$

$$\rightarrow \quad v(t) - v(0) = -\frac{|F_{max}|}{m} \int_{\tau=0}^{t} d\tau$$

$$\rightarrow \quad v(t) = -\frac{|F_{max}|}{m} t \quad (3)$$

We have integrated through time from the start at $t = 0$ up until some unspecified time t. When evaluating the integral, we needed to add the initial speed of the slug $v(0)$. In our case, since we began with the slug at rest, $v(0) = 0$. We can interpret Equation (3) as follows. The slug's speed v at any time is proportional to the time t. During every unit of time (a second, for example), the slug's speed increases by a further quantum of $-|F_{max}|/m$. The minus sign states that the speed increases in the negative direction, and we know that the negative direction points towards the right in the figure above.

It is important to realize that Equation (3) remains valid only so long as the slug remains within the force field. Once the slug passes through the force field, the force is no longer given by $-|F_{max}|$. Instead, it is zero. The slug no longer accelerates once it passes through the force field. We can determine when that happens by integrating Equation (3) to get the position z of the slug at time t.

$$\frac{dz}{dt} = v$$

$$\rightarrow \quad \int_{\tau=0}^{t} \frac{dz}{d\tau} d\tau = \int_{\tau=0}^{t} v d\tau$$

$$\rightarrow \quad \int_{\tau=0}^{t} dz = \int_{\tau=0}^{t} \left(-\frac{|F_{max}|}{m}\tau\right) d\tau$$

$$\rightarrow \quad z(t) - z(0) = -\frac{|F_{max}|}{m} \int_{\tau=0}^{t} \tau d\tau$$

$$\rightarrow \quad z(t) - z(0) = -\frac{|F_{max}|}{2m} \int_{\tau=0}^{t} d\tau^2$$

$$\rightarrow \quad z(t) = z(0) - \frac{|F_{max}|}{2m} t^2$$

$$\rightarrow \quad z(t) = D - \frac{|F_{max}|}{2m} t^2 \quad (4)$$

Once again, the fact that F_{max} and m are constant makes the integration easy. This time, evaluating the integral requires that we add the initial position of the slug, $z(0)$. This is not zero – remember that we parked the slug at position $z = D$ before we began the run.

We can interpret Equation (4) as follows. At time $t = 0$, the term in t^2 vanishes and the slug's position is equal to D, just as it should be. Thereafter, the slug's position decreases (the minus sign subtracts an increasing amount from D) as the slug moves towards the right. The decreases happen at an increasing rate. Every unit of time (a second, for example), the change in distance given by the term in t^2 is bigger than it was for the previous unit of time. The slug is accelerating.

Equation (4), like Equation (3), is only valid so long as the slug remains within the force field. We can now figure out when the slug leaves the force field. It leaves when it reaches the origin or, said differently, when its position z reaches zero. Let us define the symbol T_{end} as the instant in time when the slug reaches the position $z = 0$. At that instant, Equation (4) can be evaluated as:

$$0 = -\frac{|F_{max}|}{2m}T_{end}^2 + D$$

$$\rightarrow \quad T_{end}^2 = \frac{2mD}{|F_{max}|}$$

$$\rightarrow \quad T_{end} = \sqrt{\frac{2mD}{|F_{max}|}} \qquad (5)$$

Let us look at some of the dependencies in Equation (5). Since the mass m is in the numerator of the fraction, a bigger mass will result in a bigger T_{end}. That makes sense. It is harder to pull on a heavier mass, so it should take longer to pull it through the given distance D. A bigger distance D means we have to drag the slug further, which will also take more time for a given magnitude of force $|F_{max}|$. And, since the force is in the denominator of the fraction, a bigger force will result in a smaller T_{end}. That makes sense, too. A bigger force gets the job done sooner.

One might ask: what is the speed of the slug at the end of the run? We can calculate that by substituting the ending time $t = T_{end}$ into Equation (3), which is the expression for the slug's speed. If we use the symbol V_{end} for the slug's final speed, we get:

$$V_{end} = -\frac{|F_{max}|}{m}T_{end}$$

$$= -\frac{|F_{max}|}{m}\sqrt{\frac{2mD}{|F_{max}|}}$$

$$= -\sqrt{\frac{2D|F_{max}|}{m}} \qquad (6)$$

The minus sign simply means that the slug is traveling towards the right, in the direction of the negative $\hat{z}$-axis. The dependence of the speed on $|F_{max}|$ and m has been reversed. A bigger mass, for example, results in a lower ending speed, and so forth. The square root tells its own story. It tells us that some things get harder to do. Suppose we want to increase the terminal speed of the slug by increasing the force we exert on the slug. Doubling the force $|F_{max}|$ does not double the slug's final speed. Instead, it increases the final speed by a factor of only $\sqrt{2} \cong 1.4$. In order to double the slug's final speed, we would

need to increase the force by a factor of four. Similarly, if building a longer coil caused a proportional increase in the spatial extent D of the force field, we would need to build a coil four times as long to double the final speed of the slug. On the other hand, decreasing the mass of the slug would be a quick way to increase its final speed.

However, we are not interested only in increasing the slug's speed. What we are really interested in the kinetic energy the slug has. The kinetic energy is a better measure of how much work the slug can do. A heavy slug moving slowly can do the same amount of work as a lighter slug moving more quickly. The kinetic energy of a body depends on both its mass and its speed. If we use the symbol K_{end} for the kinetic energy of the slug at the end of the run, then we can write:

$$
\begin{aligned}
K_{end} &= \tfrac{1}{2}mV_{end}^2 \\
&= \tfrac{1}{2}m\left(-\sqrt{\frac{2D|F_{max}|}{m}}\right)^2 \\
&= D|F_{max}| \qquad\qquad (7)
\end{aligned}
$$

This expression changes things quite a bit. It seems that the ending kinetic energy of the slug does not depend on its mass at all. The ending kinetic energy is, in fact, a constant: the magnitude of the force multiplied by the distance through which the slug was accelerated. (This is one manifestation of a general principle of mechanics – that the work done by a force is equal to the product of the force and the distance through which it acts on a body.) It is worth noting that both D and $|F_{max}|$ are properties of the coil, not of the slug.

We are not quite finished with this example. One question which will arise again is the efficiency with which energy was imparted to the slug. We now know the kinetic energy which the slug has at the end of its run, but that energy had to come from somewhere. In this example, we did not need to know anything about the power supply. However, let us suppose that the power supply which was powering the coil was producing energy at a rate of P_{ps}. Electrically, the power supply was pushing current through the coil, overcoming its resistance as well as doing work on the slug. If the power supply produced constant power throughout the run (in reality, it will not), then the total amount of energy it did during the run is equal to the power P_{ps} multiplied by the time during which it was delivered. That length of time, of course, is equal to the duration of the run, T_{end}. If we use the symbol E_{ps} for the total energy supplied by the power supply during the run, then:

$$
\begin{aligned}
E_{ps} &= P_{ps} \times T_{end} \\
&= P_{ps}\sqrt{\frac{2mD}{|F_{max}|}} \qquad (8)
\end{aligned}
$$

We can measure the efficiency as the fraction of the energy provided by the power supply which was converted in kinetic energy of the slug. If we use the symbol η for the efficiency, then;

$$
\eta = \frac{K_{end}}{E_{ps}} = \frac{D|F_{max}|}{P_{ps}}\sqrt{\frac{|F_{max}|}{2mD}} = \frac{1}{P_{ps}}\sqrt{\frac{D|F_{max}|^3}{2m}} \qquad (9)
$$

If the power supply was running continuously, then the efficiency would not be very important. The run will be over in the blink of an eye, and its contribution to global warming is not an issue. But, the

efficiency is important if the power supply is a charged capacitor. We will start the run with a fixed charge on the capacitor, which represents a fixed stock of energy. That is all the energy which will be available for the run, and we will want to use it as efficiently as possible.

The second introductory example

Let us repeat the analysis using a more realistic spatial distribution for the force. The new spatial distribution is shown in the following figure. Although the force on the slug now varies with its distance from the center of the coil, we will, as before, assume that the force field is constant with respect to time.
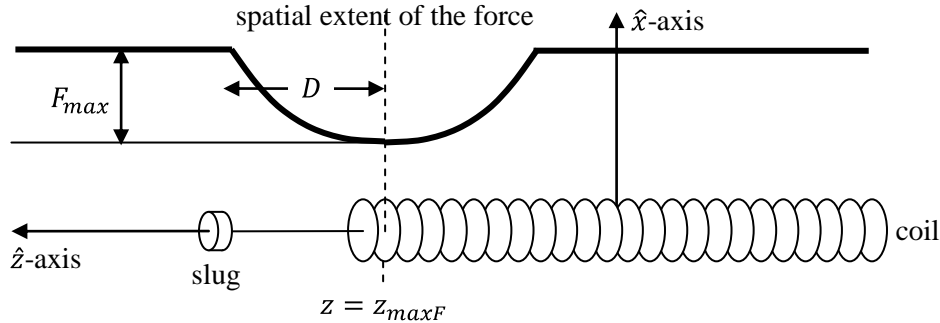
The maximum value of the attractive force occurs near the face of the coil, but not necessarily exactly at the face. We will now use the symbol $F(z)$ as the value of the force at some distance z from the center of the coil. The maximum magnitude of the force will be denoted by F_{max}. We will assume that the force field extends a distance D on either side of the point where it reaches its maximum, which point has the position z_{maxF}. The force, being attractive towards the center of the coil, is algebraically negative. As before, we will park the slug at the limit of the force field before the run starts, so that it gets as much energy from the force field as it can.

In order to get quantitative results, we must have some expression which shows how the force depends on distance. We want something that looks like the "hump" shown but is not too difficult to manipulate mathematically. Suppose we use the following expression for the force field.

$$F(z) = -|F_{max}|\left[1 - \left(\frac{z - z_{maxF}}{D}\right)^2\right] \text{ for the range } |z - z_{maxF}| \leq D \quad (10)$$

This curve happens to be a parabola. At the force maximum, where $z = z_{maxF}$, the expression evaluates out to $-|F_{max}|$. Since $(z - z_{maxF})^2$ is symmetric around $z = z_{maxF}$, the function will look the same on both sides of z_{maxF}. The function will have lower magnitudes on both sides of z_{maxF}, and will reach zero when z is a distance D away from z_{maxF} on either side.

Although the parabola continues outside of the range, we will simply set the value of the force equal to zero there. As in the first example, we will somehow "turn the power off" once the slug gets to the right-hand end of the force field, at $z = z_{maxF} - D$.

Newton's Law and the equations of motion which we integrate to get the speed and then the position, are the same as they were in the first example. Because we have a different force field, though, the results of the integration will be different. Furthermore, since the force depends on the slug's position, which in turn depends on time, we will have to deal with two variables and not with just the time alone. Let's work through it.

Substituting the equation for the force field into Newton's Law, written in terms of the slug's position z, we get:

$$\frac{d^2 z}{dt^2} = \frac{F(z)}{m}$$

$$= -\frac{|F_{max}|}{m}\left[1 - \left(\frac{z - z_{maxF}}{D}\right)^2\right]$$

$$= -\frac{|F_{max}|}{mD^2}[D^2 - (z - z_{maxF})^2]$$

$$\rightarrow \quad \frac{d^2 z}{(z - z_{maxF})^2 - D^2} = \frac{|F_{max}|}{mD^2}dt^2 \qquad (11)$$

Notwithstanding the simplicity of its form, this is quite a difficult integral, difficult enough that we will not even try to solve it in closed form. Instead, we will start over again, using Newton's Law expressed in a different form. Recall that the change in kinetic energy of the slug during any small interval of time is equal to the force acting on the slug multiplied by the (correspondingly small) distance through which the force drags the slug. We can develop this thought more rigorously by taking the derivative of the definition of the slug's kinetic energy as follows:

$$K = \frac{1}{2}mv^2$$

$$\rightarrow \quad \frac{dK}{dt} = mv\frac{dv}{dt}$$

$$\rightarrow \quad \frac{dK}{dt} = m\left(\frac{dz}{dt}\right)\left(\frac{dv}{dt}\right)$$

$$\rightarrow \quad \left(\frac{dK}{dt}\right)\left(\frac{dt}{dz}\right) = m\left(\frac{dv}{dt}\right)$$

$$\rightarrow \quad \left(\frac{dK}{dt}\right)\left(\frac{dt}{dz}\right) = m\left(\frac{d^2 z}{dt^2}\right)$$

$$\rightarrow \quad \frac{dK}{dz} = m\frac{d^2 z}{dt^2}$$

$$\rightarrow \quad \frac{dK}{dz} = F \qquad (12)$$

In other words, the derivative of the slug's kinetic energy with respect to <u>distance</u> is equal to the force at that position. This is very helpful for us. Our expression for the force field is expressed in terms of distance. With time removed from the equation, we can integrate easily. Substituting the expression for the force field gives:

$$\frac{dK}{dz} = -|F_{max}|\left[1 - \left(\frac{z - z_{maxF}}{D}\right)^2\right]$$

$$\rightarrow \quad \int_{K=K_0}^{K} dK = \frac{|F_{max}|}{D^2}\int_{z=z_{maxF}+D}^{z}[(z - z_{maxF})^2 - D^2]dz$$

$$\rightarrow \quad K - K_0 = \frac{|F_{max}|}{D^2}\left[\frac{1}{3}(z - z_{maxF})^3 - D^2 z\right]\Big|_{z=z_{maxF}+D}^{z} \qquad (13)$$

Let me pause to explain. We integrate both sides from the starting configuration. Since the slug is at rest at the start of a run, its starting kinetic energy K_0 will be equal to zero. The force expression on the right-hand side is integrated over distance, not time, from the starting configuration. Since the slug begins the run at the left-most extent of the force field, its starting abscissa is equal to $z_{maxF} + D$. And, of course, the integral is valid only so long as the slug remains within the force field. To continue, we evaluate the right-hand side at its limits to obtain:

$$
\begin{aligned}
K \quad &= \frac{|F_{max}|}{D^2}\left\{\left[\frac{1}{3}(z - z_{maxF})^3 - D^2 z\right] - \left[\frac{1}{3}(z_{maxF} + D - z_{maxF})^3 - D^2(z_{maxF} + D)\right]\right\} \\
&= \frac{|F_{max}|}{D^2}\left[\frac{1}{3}(z - z_{maxF})^3 - D^2 z - \frac{1}{3}D^3 + D^2(z_{maxF} + D)\right] \\
&= \frac{|F_{max}|}{D^2}\left[\frac{1}{3}(z - z_{maxF})^3 - D^2(z - z_{maxF}) + \frac{2}{3}D^3\right] \quad (14)
\end{aligned}
$$

This expression is the kinetic energy of the slug at various distances as it progresses through the force field. We can calculate the kinetic energy of the slug at the end of the run (K_{end}) by evaluating Equation (14) at the abscissa where the slug leaves the force field, which will now occur when $z = z_{maxF} - D$. This gives:

$$
\begin{aligned}
K_{end} \quad &= \frac{|F_{max}|}{D^2}\left[\frac{1}{3}(z_{maxF} - D - z_{maxF})^3 - D^2(z_{maxF} - D - z_{maxF}) + \frac{2}{3}D^3\right] \\
&= \frac{|F_{max}|}{D^2}\left(-\frac{1}{3}D^3 + D^3 + \frac{2}{3}D^3\right) \\
&= \frac{4|F_{max}|}{3D^2}D^3 \\
&= \frac{4}{3}|F_{max}|D \quad (15)
\end{aligned}
$$

Once again, the terminal kinetic energy of the slug does not depend on the slug's mass. It depends only on the parameters F_{maxF} and D of the force field. (Nothing is to be gained by comparing the magnitude of the terminal kinetic energy here with that of the previous example, since the Fs and Ds involved have different meanings in the two examples.)

We can calculate the ending speed of the slug directly from its ending kinetic energy, as follows:

$$
\frac{1}{2}mV_{end}^2 = K_{end}
$$

$$
\rightarrow \quad V_{end} = \pm\sqrt{\frac{2K_{end}}{m}}
$$

$$
\rightarrow \quad V_{end} = -\sqrt{\frac{8|F_{max}|D}{3m}} \quad (16)
$$

Note that taking the square root of V_{end}^2 gives two possible results for V_{end}, one positive and one negative. Since the slug is traveling towards the right, we want the negative one. V_{end} for this force field has the same general dependencies on $|F_{max}|$, D and m as Equation (6), which gives V_{end} for the first example. In both cases, the ending speed is proportional to $\sqrt{|F_{max}|}$ and to $\sqrt{D}$ and inversely proportional to $\sqrt{m}$.

So far, we have said nothing at all about time. How long did it take the slug to pass through the force field? To answer this question, let will intercept the above analysis at Equation (14), which gives the slug's kinetic energy as a function of its position. Substituting the definition of kinetic energy gives:

$$\frac{1}{2}mv^2 = \frac{|F_{max}|}{D^2}\left[\frac{1}{3}(z - z_{maxF})^3 - D^2(z - z_{maxF}) + \frac{2}{3}D^3\right]$$

$$\rightarrow \left(\frac{dz}{dt}\right)^2 = \frac{2|F_{max}|}{mD^2}\left[\frac{1}{3}(z - z_{maxF})^3 - D^2(z - z_{maxF}) + \frac{2}{3}D^3\right]$$

$$\rightarrow \frac{dz}{dt} = -\sqrt{\frac{2|F_{max}|}{mD^2}}\sqrt{\frac{1}{3}(z - z_{maxF})^3 - D^2(z - z_{maxF}) + \frac{2}{3}D^3} \quad (17)$$

We then integrate through the complete run as follows:

$$\int_{z=z_{maxF}+D}^{z=z_{maxF}-D} \frac{dz}{\sqrt{\frac{1}{3}(z - z_{maxF})^3 - D^2(z - z_{maxF}) + \frac{2}{3}D^3}} = -\int_{\tau=0}^{t=T_{end}} \sqrt{\frac{2|F_{max}|}{mD^2}}\,d\tau \quad (18)$$

The right-hand side is an easy integral. Unfortunately, the one on the left-hand side is not. Even though we started with a simple expression for spatial distribution of the force field, we are stuck. There is, however, a silver lining. When we tackle the spatial distribution of a real force field, there will not be a closed-form solution either. This being only an example, we will take the opportunity to test the numerical procedure we are going to need to use anyway.

Numerical integration through a force field

The following figure shows the configuration of the general case we will consider.
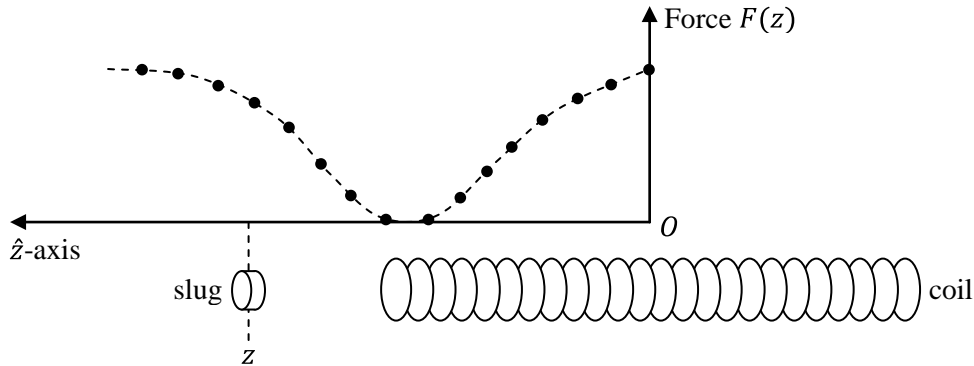
Generally, we will not have a closed-form expression for the force on the slug $F(z)$ as a function of the slug's position z. Instead, we will have calculated a set of points which, if we had enough of them, would constitute the curve. We will probably have something like 100 or 1,000 points, each point described by its distance along the axis from the origin O and the force on the slug when it is at that location. The points may or may not be spaced at equal intervals of distance.

We will integrate numerically by moving the slug in small steps of time towards the origin. Each step will have the same duration. We will keep track of the slug's speed and acceleration as we go along. At each step, we will look up the force exerted on the slug when it is at that location, revise the acceleration in accordance with the force, and then calculate what that does to the speed. Where the slug will be at the start of the next time step is the product of the revised speed and the duration of the present time step.

We want the time steps, and the corresponding steps in the slug's position, to be very small. What constitutes "small" is a matter of judgment. What we will aim for is distance steps that are small enough to let us assume that the force acting on the slug is constant during the step. We know that the force is not constant during a step, no matter how small the step is, but we can minimize the error by using very short time steps.

Suppose that the speed of the slug at the end of the previous step is known. That speed will be the slug's speed at the start of this time step, and we will refer to it as v_{start}. Let us use the symbol ΔT as the length, in seconds, of a time step. Since the time steps are short, the speed of the slug will not change very much during a single time step. We can predict that, at the end of this step, the slug will be a short distance $v_{start}\Delta t$ closer to the origin. If the position of the slug at the start of this time step is given by z_{start}, then its position at the end of this time step is going to be pretty close to $z_{end} = z_{start} + v_{start}\Delta T$. We can look up the magnitude of the force at the start of the step and at the (predicted) end of the step. These two values of the force will be $F(z_{start})$ and $F(z_{start} + v_{start}\Delta T)$, respectively. We will take the average of these two forces, one at the beginning and one at the end, and use the average as, well, the average force acting during this step. So, the average force during this time step will be very close to:

$$F_{avg} = \frac{1}{2}[F(z_{start}) + F(z_{start} + v_{start}\Delta T)] \quad (19)$$

We already know how the slug will accelerate under the influence of a constant force. We looked at this in the first example, above. Since the force is (assumed to be) constant during this time step, the average acceleration during this time step a_{avg} will also be constant. From Newton's Law:

$$a_{avg} = -\frac{|F_{avg}|}{m} \quad (20)$$

As always, we need that provoking minus sign to ensure that the slug moves towards the right. The slug's speed will increase linearly with time, so that, at the end of this time step, it will be:

$$v_{end} = v_{start} + a_{avg}\Delta T \quad (21)$$

The slug's position will change as follows:

$$z_{end} = z_{start} + v_{start}\Delta T + \frac{1}{2}a_{avg}(\Delta T)^2 \quad (22)$$

The slug's position at the end of the time step is the sum of three terms. The first term is its position at the end of the previous step. The second term is the displacement due to the speed it had at the start of the time step. And, the third term is the additional change in position due to the acceleration imparted by the force field during this time step. Once the new position of the slug is known, the cycle can be repeated for the next time step.

We need to re-visit the force field, or rather, the points at which we know the magnitude of the force. The procedure described above assumed that we could look up the value of the force at the beginning of each step and at the predicted end-point of the step. That is not going to be possible. We will have a table for the force at certain distances from the origin, but there is no certainty at all that we will have a value for any particular position where the time steps leave us. We will have to make some kind of estimate for the force for positions between the entries in the table. Making estimates between known points is called interpolation. We will interpolate in a relatively simple way.

We will assume that the force field is linear between adjacent data points. Graphically, that means that the "curve" of the force field is assumed to be a straight line between adjacent data points. The following figure shows the quantities we will use to interpolate.
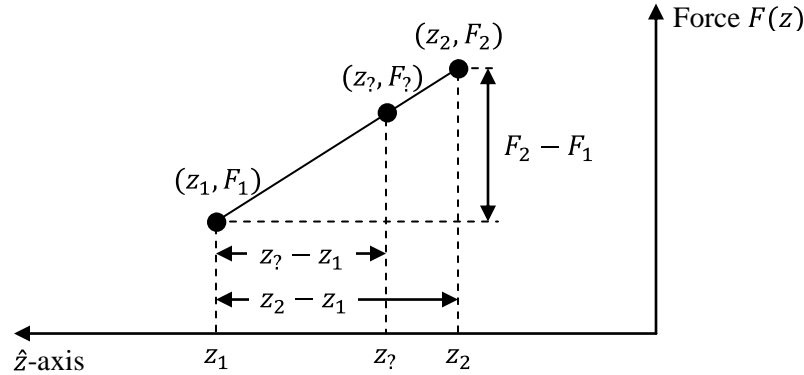
The two points which we have in our data table are labeled (z_1, F_1) and (z_2, F_2), respectively. We want to find the force $F_?$ at some abscissa $z_?$ between the known points z_1 and z_2. The "rise" and "run" between the two known points are $F_2 - F_1$ and $z_2 - z_1$, respectively, so the slope of the line segment is equal to:

$$\text{slope} = \frac{F_2 - F_1}{z_2 - z_1}$$

The line segment passing between the known point on the left (z_1, F_1) and the unknown point $(z_?, F_?)$ will have exactly the same slope, so that:

$$\frac{F_? - F_1}{z_? - z_1} = \frac{F_2 - F_1}{z_2 - z_1}$$

$$\rightarrow \quad F_? - F_1 = \left(\frac{z_? - z_1}{z_2 - z_1}\right)(F_2 - F_1)$$

$$\rightarrow \quad F_? = F_1 + \left(\frac{z_? - z_1}{z_2 - z_1}\right)(F_2 - F_1) \quad (23)$$

Equation (23) gives us a quick means to calculate the magnitude of the force at points between known horizontal locations. Using this estimate will introduce errors into the integration. The error can be reduced, of course, by having more data points in the table. In essence, having more data points in the table is tantamount to spacing the known points more closely together along the $\hat{z}$-axis. The more closely the known points are spaced, the more the curve joining two adjacent points on the curve resembles a straight line. Even so, there will be systematic error. For a real coil, the ideal curve is "curved" – at some distances from the origin, it will be curved upwards and at others it will be curved downwards. In regions where the force field is concave upwards, our estimate will consistently be above the curve. In regions where the force field is concave downwards, our estimate will consistently be below the curve. These consistent errors may or may not balance themselves out during the course of a run.

Appendix "A" attached hereto is a listing of a short Visual Basic program which carries out the integration for the parabolic force field which remains constant with respect to time. The program was developed using Visual Basic 2010 Express, which is available free as a download from Microsoft. The program saves interim results into an Excel file. The program is named *Integration1*.

I carried out the integration using a force field described by Equation (10) with the following physical parameters:

- Maximum force $F_{max} = 100$ Newtons;
- Force peaks at abscissa $z_{maxF} = 19$ centimeters from the center of the coil;
- Force extends a distance $D = 10$ centimeters on either side of the peak and
- Slug's mass $m = 10$ grams.

The force field is calculated at 1001 points and the results saved in an array named Ftable(1001,2). The spatial extent of the force (20 centimeters) was divided into 1000 equal segments, giving rise to 1001 points. The abscissa of point #N is saved as element Ftable(N,1) and the force calculated at that point is saved as element Ftable(N,2).

The integration itself was carried out with a time step ΔT of one microsecond. The following graph shows the slug's speed with respect to time.
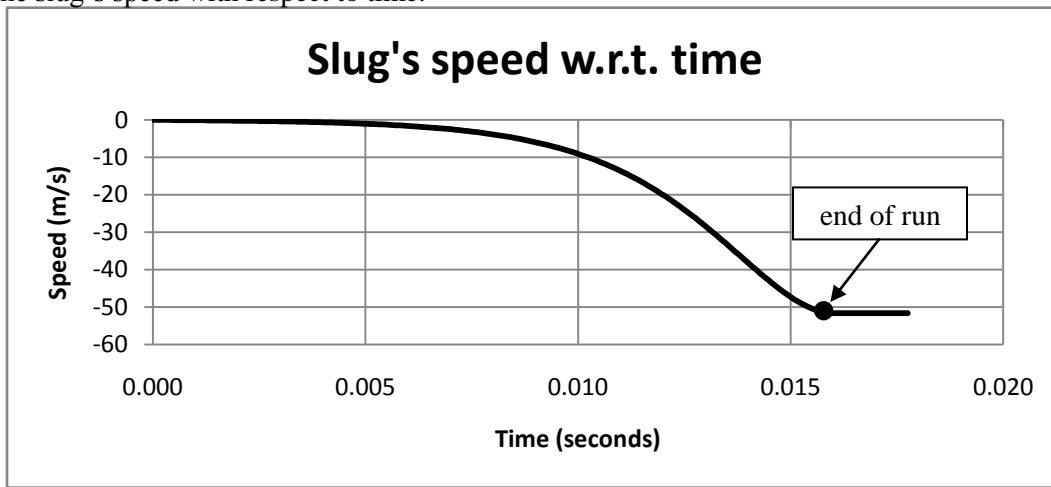
The run takes 15.99 ms, at which time the slug reaches the right-hand extent of the force field. This point is highlighted with a black dot in the graph above. After reaching this point, the slug continues to coast with its ending speed. The terminal speed is -51.64 m/s, with the minus sign reminding us that the slug is traveling in the direction of the negative $\hat{z}$-axis.

The following curve shows the slug's acceleration with respect to time.

~ 13 ~
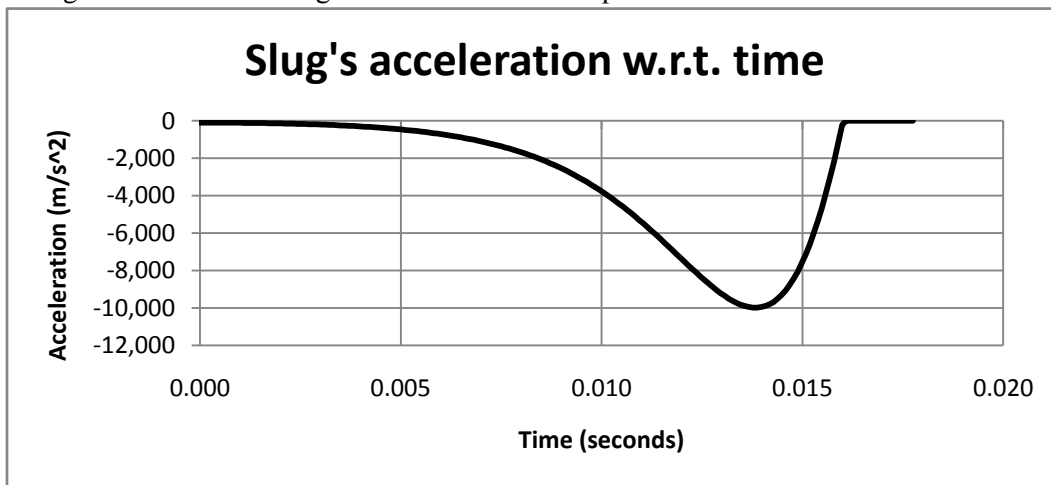
It is apparent that the slug gets off to a very slow start. During the first 5 ms of the run, it hardly accelerates at all. Indeed, I did <u>not</u> place slug at the very left-most extent of the force field at the start of the run. The left-most extent of the force field is the sum of the abscissa of the peak (19 cm) plus the extent of the force (10 cm), or 29 cm from the center of the coil. The force is zero exactly at that spot. Starting the slug exactly at that spot would be useless – it would never be pulled into the force field. To deal with this small difficulty, I "jogged" the slug 0.0005 cm to the right before starting the integration.

This small jog – one-half millimeter – is quite arbitrary but has a huge impact on the length of time taken for the run. Had the jog been smaller, it would have taken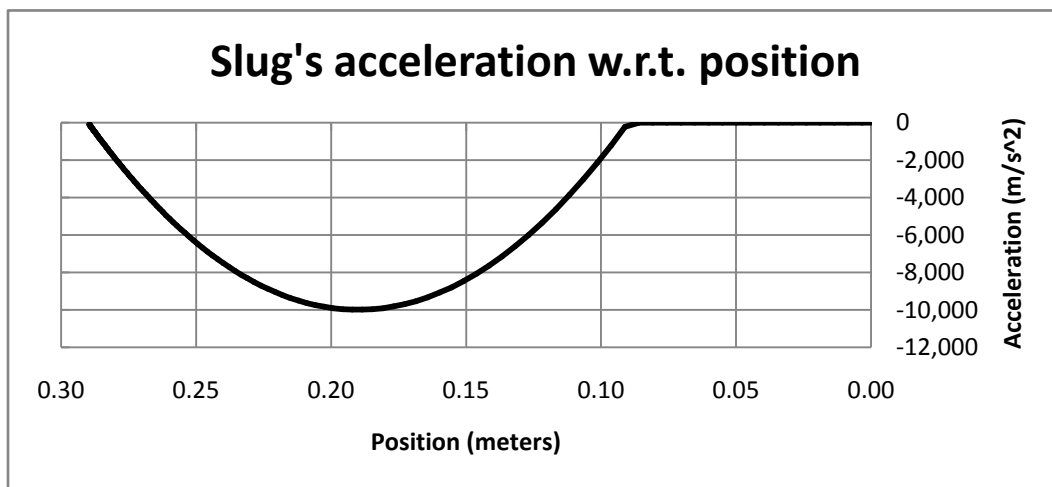 even longer for the slug to creep far enough into the force field for the force to "really take hold". On the other hand, the jog is done at a point where the force is extremely small, so its effect on the slug's final speed and kinetic energy is negligible.

The graph above is the slug's acceleration with respect to <u>time</u>. The following graph shows the acceleration with respect to the slug's <u>position</u>.

Slug's acceleration w.r.t. position

Acceleration (m/s^2): 0, -2,000, -4,000, -6,000, -8,000, -10,000, -12,000

Position (meters): 0.30, 0.25, 0.20, 0.15, 0.10, 0.05, 0.00

This graph shows the symmetry of the force field. In fact, since the slug's mass is a constant, the acceleration is proportional to the force. For example, the maximum force is 100 N. The maximum acceleration will be this maximum force divided by the slug's 10 gram mass, or 10,000 m/s^2. And, the peak occurs where it should – exactly 19 cm from the origin. A plot of the spatial distribution of the force field would look just like this graph.

The slug's kinetic energy is shown in the following graph.

Slug's energy w.r.t. time

Energy (Joules): 0, 2, 4, 6, 8, 10, 12, 14

Time (seconds): 0.000, 0.005, 0.010, 0.015, 0.020
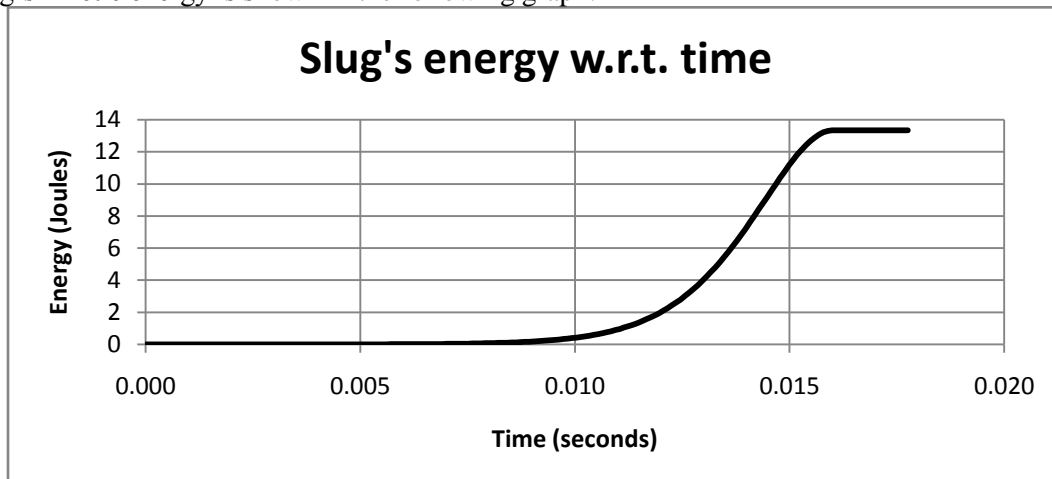
The slug's ending kinetic energy is 13.33 Joules. It should be noted that this kinetic energy and the slug's ending speed of -51.64 m/s are exactly the results of computing Equations (15) and (16), respectively, using the physical parameters assumed.

The third introductory example

The graph of the slug's kinetic energy makes clear, once again, the ineffectiveness at the start of the run. During the first 10 ms or the run, which is about two-thirds of the whole duration of the run, the slug receives hardly any energy at all. Virtually all of the slug's ending energy was imparted to it during the last third of the run.

When the force field is constant with respect to time, as it was in the second introductory example, that waste of time is not a problem. It becomes a serious problem if the stock of energy available to accelerate the slug is fixed. Under those conditions, we would not want to waste energy while the slug is located somewhere on the fringes of the force field. Instead, we would want to start the run off with the slug placed somewhere where the field is strong. The goal of this third introductory example is to explore where we should place the slug to extract as much of the given stock of energy as possible.

I have already said that the stock of energy for the coil gun will be stored in a capacitor. The capacitor will be discharged through the coil. The start of the discharge is the start of the run. If the principal physical characteristic of the coil is the resistance of the wire from which it is wound, then the voltage over the capacitor will decrease to zero exponentially with time. The rate at which the voltage decreases will be determined by the values of the capacitance and the resistance. It is convenient to describe such a resistor-capacitor pair with its time-constant, for which we will use the symbol τ_{RC}. The time-constant has units of time. One time-constant after the discharge begins, the voltage over the capacitor will have decreased from its initial voltage to a fraction $1/e$ of its initial voltage. e is the "natural number", whose value $e = 2.71828 \dots$ After two time-constants, the voltage over the capacitor will have decreased to a fraction $1/e^2$. After three time-constants, the voltage fraction will be $1/e^3$, and so forth. Mathematically, the voltage will never reach exactly zero. In practice, though, the voltage will get small enough to be considered negligible. It is often said that the capacitor is fully discharged after five time-constants. After five time-constants, the capacitor's voltage is $1/e^5 \approx 0.6\%$ of its original value.

That is what happens to the voltage over the capacitor. If the circuit contains only the capacitor and the resistance of the coil's wire, then the current will decrease in the same exponential way as the voltage. In reality, the coil does have other physical properties, such as its inductance. But, setting that aside, it is a reasonable assumption that the current will decrease exponentially with time. Mathematically, we can express this relationship as:

$$I(t) = I_0 e^{-\frac{t}{\tau_{RC}}} \quad (24)$$

where $I(t)$ is the current at some time t, I_0 is the immense current which flows at the start of the run and τ_{RC} is the time-constant of the resistor-capacitor pair.

In this third introductory example, we are not interested in the voltage or the current, but on the force exerted on the slug. We will find out in the next part of this paper that the force is proportional to the square of the current. This means that the force will also decrease exponentially with time, but with one-half the time constant, which is to say:

$$F(z,t) = F(z,0)\left(e^{-\frac{t}{\tau_{RC}}}\right)^2$$
$$= F(z,0)e^{-\frac{2t}{\tau_{RC}}} \qquad (25)$$

Where $F(z,t)$ is the force exerted on the slug if it is a position z at time t and $F(z,0)$ is the spatial distribution, or shape, of the force field at the start of the run, when the current is equal to I_0. Because the spatial distribution of the force does not change with time, it is convenient to represent it as $\mathcal{F}(z)$. For a given coil, it is useful to calculate the spatial distribution when the current I is equal to one Ampere. Then, the dependence of the total force $F(z,t)$ can be separated into two factors, one of which depends on position and the other on current, as follows:

$$F(z,t) = \mathcal{F}(z)I^2$$
$$= \mathcal{F}(z)I_0^2 e^{-\frac{2t}{\tau_{RC}}} \qquad (26)$$

In this example, we will assume that the force field can be expressed mathematically by the following expression:

$$F(z,t) = -|F_{max}|\left[1 - \left(\frac{z - z_0}{D}\right)^2\right]I_0^2 e^{-\frac{2t}{\tau_{RC}}} \quad \text{for the range } |z - z_0| \le D \quad (27)$$

I have kept the same parabolic spatial distribution as in the previous example. The difference is that the force field now depends on time as well, and decreases temporally with the time-constant τ_{RC}. Our new task is to find the best starting position for the slug. We will look for the starting position which gives the slug the most kinetic energy at the end of the run.

Appendix "B" attached hereto is a listing of another short Visual Basic program (named *Integration2*) which searches for this maximum ending energy. In essence, the new program iterates the integration in *Integration1*, with each iteration testing a different starting position. There are three differences and similarities between the two programs.

1.   *Integration2* uses the same table Ftable(1001,2) as *Integration1* to hold data points describing the force field's spatial distribution. The table holds the values of the force at time $t = 0$ calculated with a current $I_0 = 1\ A$. When a value is retrieved from the table for a given abscissa z, the value is then multiplied by the factor $I_0^2 \exp\left(-2t/\tau_{RC}\right)$ evaluated at the current simulation time t.

2.   *Integration2* also uses the same difference equations to describe the mechanics of the slug. In both programs, an average value for the force is calculated for each time step. The slug is assumed to respond to this constant force throughout the time step.

3.   In *Integration1*, this average force value was computed as the simple average of: (i) the force on the slug at its position at the start of a time step, and (ii) the force on the slug at the position it will have at the end of the time step, if it coasts along at the speed it has at the start of the time step. *Integration2* modifies this only slightly: the estimated force at the end of the time step is also reduced by the passage of time during the time step.

To carry out the integration, we also need to specify the time-constant τ_{RC} of the decay. I observe from the graphs in the second introductory example, that the slug did most of its accelerating during the last 6 ms of the run. Based on this, I will arbitrary set the time-constant to 10 ms. We also need to specify a current. For this example, I simply let the current be one Ampere.

I had *Integration2* carry out 99 runs. I started the slug at 99 different positions inside the force field. To do this, I divided the horizontal extent of the force field (the distance $2D$) by 100 (which gives rise to 101 different points), and started the slug at consecutive positions from 1% inside the force field on the left to 1% inside the force field on the right. The kinetic energy which the slug has at the ends of the 99 runs are shown in the following graph.
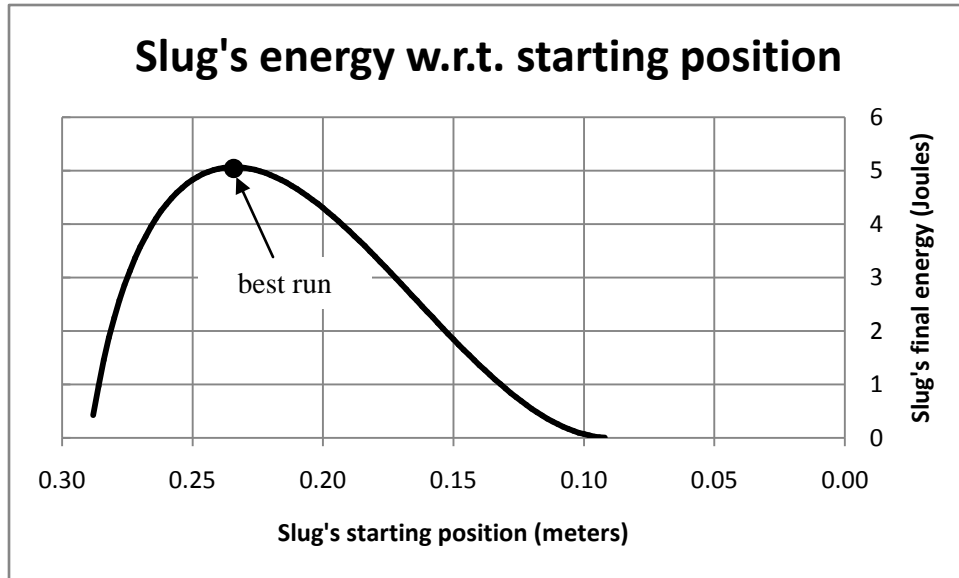
The best run occurred when the slug started off from at a distance 23.4 cm from the origin. This is 4.4 cm to the left of the abscissa where the force field peaks and is 5.6 cm to the right of the abscissa where the force field reaches zero. In other words, the best run occurred when the slug was placed about one-quarter of the way through the force field. And, at the end of this run, the slug had 5.07 Joules of kinetic energy.

Where do we go from here?

Up to this point, we have treated the force field as something that is established by things that are not related to the slug. The force field varied with space and with time, but did not depend on where the slug was or how fast it was moving. That is not realistic, of course. In reality, the slug draws its kinetic energy from the magnetic energy in the force field and, by doing so, changes the force field itself.

We will start looking into this matter – how the slug extracts its energy from the force field – in the next part of this paper.

Jim Hawley
August 2012

An e-mail setting out errors and omissions would be appreciated.

```
Option Strict On
Option Explicit On

' Integration1

Public Class Form1
    Inherits System.Windows.Forms.Form

    '//////////////////////////////////////////////////////////////////////////
    '//////////////////////////////////////////////////////////////////////////
    '//
    '//                              Variables
    '//
    '//////////////////////////////////////////////////////////////////////////
    '//////////////////////////////////////////////////////////////////////////

    ' Timing variables
    Public delT As Double = 0.000001    ' The time step for the integration is 1 us
    Public T As Double                  ' The current time in the integration
    Public Zfinished As Double = 0      ' End the integration when the slug gets here

    ' Variables for the slug
    Public M As Double = 0.01           ' The slug weighs ten grams
    Public Zstart As Double             ' The slug's position at the start of a step
    Public Zend As Double               ' The slug's position at the end of a step
    Public Sstart As Double             ' The slug's speed at the start of a step
    Public Send As Double               ' The slug's speed at the end of a step
    Public Aavg As Double               ' The slug's average acceleration during a step
    Public ESend As Double              ' The slug's kinetic energy at the end of a step
    Public Z0 As Double                 ' The slug's starting position
    Public Zjog As Double = 0.0005      ' A small jog in the slug's starting position

    ' Variables for the mechanics during a single step
    Public ZendEst As Double            ' Estimate of the slug's position at the end
    Public Fstart As Double             ' Force at the slug's starting position
    Public FendEst As Double            ' Force at the slug's estimated ending position
    Public Favg As Double               ' Average force during the step

    ' Variables for the general force field, expressed as points in a table
    Public Ftable(1001, 2) As Double    ' (*,1) is the position; (*,2) is the magnitude
    '                                    ' Force values are stored negative
    Public FnumZ As Int32 = 1001        ' The number of horizontal points in the table

    ' Variables used as an example to generate a force field.
    Public Fmax As Double = -100        ' The maximum force is 100 Newtons
    Public ZmaxF As Double = 0.19       ' The force is a maximum 19 cm from the origin
    Public D As Double = 0.1            ' The force extends 10 cm from the maximum

    ' Variables needed to save the results
    ' (Add COM file "Microsoft Excel 12.0 Object library" to the project's References.)
    Public ExcelFileName As String = "C:\Integration1Results.xlsx"
    Public objExcel As Microsoft.Office.Interop.Excel.Application
    Public objExcelWB As Microsoft.Office.Interop.Excel.Workbook
    Public objExcelWS As Microsoft.Office.Interop.Excel.Worksheet
    Public RowNum As Int32              ' The index of a row in the spreadsheet
    Public delTsave As Double = 0.0001  ' Save the results every 100 us
```

```vb
Public Tlastsave As Double           ' The time of the last save event

Public Sub New()
    InitializeComponent()
    With Me
        Name = ""
        Text = "Axial Coil Gun - Integration #1"
        FormBorderStyle = Windows.Forms.FormBorderStyle.FixedSingle
        Size = New Drawing.Size(800, 800)
        CenterToScreen()
        .Visible = True
        Controls.Add(labelDisplay)
        labelDisplay.BringToFront()
        PerformLayout()
        BringToFront()
    End With
    Initialization()
End Sub

Public labelDisplay As New Windows.Forms.Label With _
    {.Size = New Drawing.Size(250, 100), _
     .Location = New Drawing.Point(175, 190), _
     .Text = "", .TextAlign = ContentAlignment.MiddleLeft}

Private Sub Initialization()
    '/////////////////////////////////////////
    ' Build the table for the sample force field.
    Dim Fleft As Double  ' The left-most abscissa at which to calculate the force
    Dim Fright As Double ' The right-most abscissa at which to calculate the force
    Dim FdelZ As Double  ' The spacing of the abscissas in the table
    Dim ZF As Double     ' An abscissa at which to calculate the force
    Dim F As Double      ' The value of the force at an abscissa
    Fleft = ZmaxF + D
    Fright = ZmaxF - D
    FdelZ = (Fleft - Fright) / (FnumZ - 1)
    For I As Int32 = 1 To FnumZ Step 1
        ' Calculate the next abscissa in the table.
        ZF = Fleft - ((I - 1) * FdelZ)
        ' Calculate the magnitude of the force.
        F = Fmax * (1 - (((ZF - ZmaxF) / D) ^ 2))
        ' Save the abscissa and the magnitude in the table.
        Ftable(I, 1) = ZF
        Ftable(I, 2) = F
    Next I
    '////////////////////////
    ' Open the Excel file.
    Try
        objExcel = CType(CreateObject("Excel.Application"), _
            Microsoft.Office.Interop.Excel.Application)
        objExcel.Visible = False
        objExcelWB = CType(objExcel.Workbooks.Open(ExcelFileName), _
            Microsoft.Office.Interop.Excel.Workbook)
        objExcelWS = CType(objExcelWB.Sheets("Sheet1"), _
            Microsoft.Office.Interop.Excel.Worksheet)
    Catch ex As Exception
        Cursor.Current = Cursors.Default
        MsgBox("Could not open the Excel file to save the output.", vbOKOnly)
        Exit Sub
```

```vb
        End Try
        ' Save a description of this configuration in the Excel file.
        With objExcelWS
            .Cells(1, 1) = "Integration #1"
            .Cells(2, 1) = "The force field is parabola, constant in time."
            .Cells(4, 1) = "Mass = " & Trim(Str(M)) & " kilograms"
            .Cells(5, 1) = "Fmax = " & Trim(Str(Fmax)) & " Newtons"
            .Cells(6, 1) = "Max at Z = " & Trim(Str(ZmaxF)) & " meters"
            .Cells(7, 1) = "D = " & Trim(Str(D)) & " meters"
            .Cells(8, 1) = "Num points = " & Trim(Str(FnumZ))
            .Cells(9, 1) = "Time step = " & Trim(Str(delT)) & " seconds"
        End With
        ' Write the headers for the output.
        With objExcelWS
            .Cells(11, 1) = "Time"
            .Cells(12, 1) = "(s)"
            .Cells(11, 2) = "Position"
            .Cells(12, 2) = "(m)"
            .Cells(11, 3) = "Speed"
            .Cells(12, 3) = "(m/s)"
            .Cells(11, 4) = "Acceleration"
            .Cells(12, 4) = "(m/s^2)"
            .Cells(11, 5) = "Force"
            .Cells(12, 5) = "(N)"
            .Cells(11, 6) = "Energy"
            .Cells(12, 6) = "(J)"
        End With
        ' Set the starting row number for adding entries to the spreadsheet.
        RowNum = 12
        StartIntegrating()
End Sub

Private Sub StartIntegrating()
    '////////////////////////////////////
    ' Set initial conditions for the slug.
    Z0 = ZmaxF + D ' Start the slug at the left-hand side of the force field.
    ' Add a small jog to the right to the slug's starting position.
    Z0 = Z0 - Zjog
    ' Write the starting data to the file.
    RowNum = RowNum + 1
    With objExcelWS
        .Cells(RowNum, 1) = 0
        .Cells(RowNum, 2) = Z0
        .Cells(RowNum, 3) = 0
    End With
    ' Set things up for the first iteration.
    T = 0
    Zstart = Z0
    Sstart = 0
    ' Set a dummy save time.
    Tlastsave = -1000
    '/////////////////////////////
    ' Main loop for the integration.
    Do
        ' Increment the time to the end of the next step.
        T = T + delT
        ' Estimate the ending position of the slug.
        ZendEst = Zstart + (Sstart * delT)
```

```
    ' Find the force at the beginning and end of the step.
    Fstart = InterpolateForceField(Zstart)
    FendEst = InterpolateForceField(ZendEst)
    ' Calculate the average force during the step.
    Favg = 0.5 * (Fstart + FendEst)
    ' Calculate the average acceleration during the step.
    Aavg = Favg / M
    ' Increment the slug's speed.
    Send = Sstart + (Aavg * delT)
    ' Increment the slug's position.
    Zend = Zstart + (Sstart * delT) + ((Aavg * delT * delT) * 0.5)
    ' Calculate the slug's kinetic energy.
    ESend = 0.5 * M * Send * Send
    ' If it is time, then save the data for this step.
    If ((T - Tlastsave) >= delTsave) Then
        RowNum = RowNum + 1
        With objExcelWS
            .Cells(RowNum, 1) = T
            .Cells(RowNum, 2) = Zend
            .Cells(RowNum, 3) = Send
            .Cells(RowNum, 4) = Aavg
            .Cells(RowNum, 5) = Favg
            .Cells(RowNum, 6) = ESend
        End With
        Tlastsave = T
    End If
    ' Check if we should stop integrating.  If so, then save the data
    ' for the last step and then exit.
    If (Zend <= Zfinished) Then
        RowNum = RowNum + 1
        With objExcelWS
            .Cells(RowNum, 1) = T
            .Cells(RowNum, 2) = Zend
            .Cells(RowNum, 3) = Send
            .Cells(RowNum, 4) = Aavg
            .Cells(RowNum, 5) = Favg
            .Cells(RowNum, 6) = ESend
        End With
        Exit Do
    End If
    ' Advise the user about progress.
    labelDisplay.Text = _
        "Current time = " & Str(T) & " s" & vbCrLf & _
        "Current location = " & Str(Zend) & " m" & vbCrLf & _
        "Current speed = " & Str(Send) & " m/s" & vbCrLf & _
        "Current acceleration = " & Str(Aavg) & " m/s^2" & vbCrLf & _
        "Current energy = " & Str(ESend) & " J"
    labelDisplay.Refresh()
    ' Update for the next iteration.
    Zstart = Zend
    Sstart = Send
Loop
' Save and close the output file.
objExcelWB.Save()
objExcelWB.Close()
objExcel.Quit()
' Alert the user to completion.
MsgBox("All finished.")
```

```
        End Sub

        '////////////////////////////////////////////////////////////////////////////////
        '////////////////////////////////////////////////////////////////////////////////
        '//
        '//                      Interpolation in the force field table
        '//
        '////////////////////////////////////////////////////////////////////////////////
        '////////////////////////////////////////////////////////////////////////////////

        Public Function InterpolateForceField(ByVal Z As Double) As Double
            ' This function returns the interpolated magnitude of the force field at
            ' for the abscissa passed as the argument.  The returned value will be
            ' negative unless the given abscissa lies outside the force field, in
            ' which case the function returns zero.
            Dim IndexToLeft As Int32
            Dim Slope As Double
            ' Check if the slug is positioned outside the force field, to the left.
            If (Z > Ftable(1, 1)) Then
                InterpolateForceField = 0
                Exit Function
            End If
            ' Check if the slug is positioned outside the force field, to the right.
            If (Z < Ftable(FnumZ, 1)) Then
                InterpolateForceField = 0
                Exit Function
            End If
            ' Find the index of the abscissa in the force field table which lies just
            ' to the left of the given abscissa.  Note that this routine does not
            ' assume that the points in the table are equally-spaced.  It assumes only
            ' that they are in order of decreasing values, that is, each point is
            ' closer to the origin than the preceding point.
            For I As Int32 = 1 To (FnumZ - 1) Step 1
                If ((Z <= Ftable(I, 1)) And (Z > Ftable(I + 1, 1))) Then
                    IndexToLeft = I
                    Exit For
                End If
            Next I
            ' Let us check for the trivial case, where we are exactly at a data point.
            If (Z = Ftable(IndexToLeft, 1)) Then
                ' By chance, we are exactly at a data point.
                InterpolateForceField = Ftable(IndexToLeft, 2)
                Exit Function
            End If
            ' We deal here with the general case, where we are between two data points.
            Try
                Slope = (Ftable(IndexToLeft + 1, 2) - Ftable(IndexToLeft, 2)) / _
                    (Ftable(IndexToLeft + 1, 1) - Ftable(IndexToLeft, 1))
            Catch ex As Exception
                MsgBox("Error in the force field table.  Two abscissas are the same.")
                InterpolateForceField = 0
                Exit Function
            End Try
            InterpolateForceField = Ftable(IndexToLeft, 2) + _
                (Slope * (Z - Ftable(IndexToLeft, 1)))
        End Function

    End Class
```

```vbnet
Option Strict On
Option Explicit On

' Integration2

Public Class Form1
    Inherits System.Windows.Forms.Form

    '////////////////////////////////////////////////////////////////////////////
    '////////////////////////////////////////////////////////////////////////////
    '//
    '//                              Variables
    '//
    '////////////////////////////////////////////////////////////////////////////
    '////////////////////////////////////////////////////////////////////////////

    ' Time-constant of the force field's decay.
    Public TC As Double = 0.01          ' The force field decays as exp(-T/TC)

    ' Timing variables
    Public delT As Double = 0.000001    ' The time step for the integration is 1 us
    Public T As Double                  ' The current time in the integration
    Public Zfinished As Double = 0      ' End the integration when the slug gets here

    ' Variables for the slug
    Public M As Double = 0.01           ' The slug weighs ten grams
    Public Zstart As Double             ' The slug's position at the start of a step
    Public Zend As Double               ' The slug's position at the end of a step
    Public Sstart As Double             ' The slug's speed at the start of a step
    Public Send As Double               ' The slug's speed at the end of a step
    Public Aavg As Double               ' The slug's average acceleration during a step
    Public ESend As Double              ' The slug's kinetic energy at the end of a step

    ' Variables for the mechanics during a single step
    Public ZendEst As Double            ' Estimate of the slug's position at the end
    Public Fstart As Double             ' Force at the slug's starting position
    Public FendEst As Double            ' Force at the slug's estimated ending position
    Public Favg As Double               ' Average force during the step

    ' Variables for the general force field, expressed as points in a table.
    Public Ftable(1001, 2) As Double    ' (*,1) is the position; (*,2) is the magnitude
    '                                   ' Force values are stored negative
    Public FnumZ As Int32 = 1001        ' The number of horizontal points in the table

    ' Variables used as an example to generate a force field.
    Public Fmax As Double = -100        ' The maximum force is 100 Newtons
    Public ZmaxF As Double = 0.19       ' The force is a maximum 19 cm from the origin
    Public D As Double = 0.1            ' The force extends 10 cm from the maximum

    ' Variables needed to save the results
    ' (Add COM file "Microsoft Excel 12.0 Object library" to the project's References.)
    Public ExcelFileName As String = "C:\Integration2Results.xlsx"
    Public objExcel As Microsoft.Office.Interop.Excel.Application
    Public objExcelWB As Microsoft.Office.Interop.Excel.Workbook
    Public objExcelWS As Microsoft.Office.Interop.Excel.Worksheet
```

```vbnet
Public RowNum As Int32                    ' The index of a row in the spreadsheet

Public Sub New()
    InitializeComponent()
    With Me
        Name = ""
        Text = "Axial Coil Gun - Integration #2"
        FormBorderStyle = Windows.Forms.FormBorderStyle.FixedSingle
        Size = New Drawing.Size(800, 800)
        CenterToScreen()
        .Visible = True
        Controls.Add(labelDisplay)
        labelDisplay.BringToFront()
        PerformLayout()
        BringToFront()
    End With
    Initialization()
End Sub

Public labelDisplay As New Windows.Forms.Label With _
    {.Size = New Drawing.Size(250, 100), _
     .Location = New Drawing.Point(175, 190), _
     .Text = "", .TextAlign = ContentAlignment.MiddleLeft}

Private Sub Initialization()
    '//////////////////////////////////////////
    ' Build the table for the sample force field.
    Dim Fleft As Double  ' The left-most abscissa at which to calculate the force
    Dim Fright As Double ' The right-most abscissa at which to calculate the force
    Dim FdelZ As Double  ' The spacing of the abscissas in the table
    Dim ZF As Double     ' An abscissa at which to calculate the force
    Dim F As Double      ' The value of the force at an abscissa
    Fleft = ZmaxF + D
    Fright = ZmaxF - D
    FdelZ = (Fleft - Fright) / (FnumZ - 1)
    For I As Int32 = 1 To FnumZ Step 1
        ' Calculate the next abscissa in the table.
        ZF = Fleft - ((I - 1) * FdelZ)
        ' Calculate the magnitude of the force.
        F = Fmax * (1 - (((ZF - ZmaxF) / D) ^ 2))
        ' Save the abscissa and the magnitude in the table.
        Ftable(I, 1) = ZF
        Ftable(I, 2) = F
    Next I
    '/////////////////////
    ' Open the Excel file.
    Try
        objExcel = CType(CreateObject("Excel.Application"),  _
            Microsoft.Office.Interop.Excel.Application)
        objExcel.Visible = False
        objExcelWB = CType(objExcel.Workbooks.Open(ExcelFileName),  _
            Microsoft.Office.Interop.Excel.Workbook)
        objExcelWS = CType(objExcelWB.Sheets("Sheet1"),  _
            Microsoft.Office.Interop.Excel.Worksheet)
    Catch ex As Exception
        Cursor.Current = Cursors.Default
        MsgBox("Could not open the Excel file to save the output.", vbOKOnly)
        Exit Sub
```

```vb
        End Try
        ' Save a description of this configuration in the Excel file.
        With objExcelWS
            .Cells(1, 1) = "Integration #2"
            .Cells(2, 1) = "The force field is a parabola with exponential decay"
            .Cells(4, 1) = "Mass = " & Trim(Str(M)) & " kilograms"
            .Cells(5, 1) = "Fmax = " & Trim(Str(Fmax)) & " Newtons"
            .Cells(6, 1) = "Max at Z = " & Trim(Str(ZmaxF)) & " meters"
            .Cells(7, 1) = "D = " & Trim(Str(D)) & " meters"
            .Cells(8, 1) = "Num points = " & Trim(Str(FnumZ))
            .Cells(9, 1) = "Time constant TC = " & Trim(Str(TC)) & " seconds"
            .Cells(10, 1) = "Time step = " & Trim(Str(delT)) & " seconds"
        End With
        ' Write the headers for the output.
        With objExcelWS
            .Cells(12, 1) = "Z0test"
            .Cells(13, 1) = "(m)"
            .Cells(12, 2) = "Energy"
            .Cells(13, 2) = "(J)"
        End With
        ' Set the starting row number for adding entries to the spreadsheet.
        RowNum = 14
        StartIntegrating()
    End Sub

    Private Sub StartIntegrating()
        '/////////////////////////////////////////////////
        ' Main loop for testing different starting positions.
        ' We will try 99 different starting positions, separated by a distance
        ' equal to 1% of the horizontal extent of the force field (2D).
        Dim Z0test As Double ' The slug's position at the start of a run
        Dim EStest As Double ' The slug's kinetic energy at the end of a run
        For I As Int32 = 1 To 99 Step 1
            Z0test = (ZmaxF + D) - (I * 2 * D / 100)
            EStest = IntegrateOneRun(Z0test)
            ' Write the result to the Excel file.
            RowNum = RowNum + 1
            With objExcelWS
                .Cells(RowNum, 1) = Z0test
                .Cells(RowNum, 2) = EStest
            End With
        Next I
        ' Save and close the output file.
        objExcelWB.Save()
        objExcelWB.Close()
        objExcel.Quit()
        ' Alert the user to completion.
        MsgBox("All finished.")
    End Sub

    '///////////////////////////////////////////////////////////////////////////////
    '///////////////////////////////////////////////////////////////////////////////
    '//
    '//                                  IntegrateOneRun
    '//
    '///////////////////////////////////////////////////////////////////////////////
    '///////////////////////////////////////////////////////////////////////////////
```

```vbnet
Public Function IntegrateOneRun(ByVal Z0test As Double) As Double
    ' This function integrates a complete run from the time the slug is at rest
    ' at abscissa Z0test (passed as an argument) until the time that it reaches
    ' Zfinished = 0 (the center of the coil).  The function returns the slug's
    ' kinetic energy at the end of the run.  This function does not write any
    ' data to the Excel file, but it does display its progress to the user on
    ' the screen.
    ' Set things up for the first iteration.
    T = 0
    Zstart = Z0test ' Start the slug where specified by the argument.
    Sstart = 0
    '///////////////////////////////
    ' Main loop for the integration.
    Do
        ' Estimate the ending position of the slug.
        ZendEst = Zstart + (Sstart * delT)
        ' Find the nominal and time-decayed force at the beginning of the step.
        Fstart = InterpolateForceField(Zstart)
        Fstart = Fstart * Math.Exp(-2 * T / TC)
        ' Increment the time to the end of the next step.
        T = T + delT
        ' Find the nominal and time-decayed force at the end of the step.
        FendEst = InterpolateForceField(ZendEst)
        FendEst = FendEst * Math.Exp(-2 * T / TC)
        ' Calculate the average force during the step.
        Favg = 0.5 * (Fstart + FendEst)
        ' Calculate the average acceleration during the step.
        Aavg = Favg / M
        ' Increment the slug's speed.
        Send = Sstart + (Aavg * delT)
        ' Increment the slug's position.
        Zend = Zstart + (Sstart * delT) + ((Aavg * delT * delT) * 0.5)
        ' Calculate the slug's kinetic energy.
        ESend = 0.5 * M * Send * Send
        ' Check if we should stop integrating.
        If (Zend <= Zfinished) Then
            Exit Do
        End If
        ' Advise the user about progress.
        labelDisplay.Text = _
            "Starting position = " & Str(Z0test) & vbCrLf & _
            "Current time = " & Str(T) & " s" & vbCrLf & _
            "Current location = " & Str(Zend) & " m" & vbCrLf & _
            "Current speed = " & Str(Send) & " m/s" & vbCrLf & _
            "Current acceleration = " & Str(Aavg) & " m/s^2" & vbCrLf & _
            "Current energy = " & Str(ESend) & " J"
        labelDisplay.Refresh()
        ' Update for the next iteration.
        Zstart = Zend
        Sstart = Send
    Loop
    ' The return value is the slug's ending energy.
    IntegrateOneRun = ESend
End Function

'////////////////////////////////////////////////////////////////////////////////
'////////////////////////////////////////////////////////////////////////////////
'//
```

```vbnet
'//                         Interpolation in the force field table
'//
'///////////////////////////////////////////////////////////////////////////////
'///////////////////////////////////////////////////////////////////////////////

    Public Function InterpolateForceField(ByVal Z As Double) As Double
        ' This function returns the interpolated magnitude of the force field at
        ' for the abscissa passed as the argument.  The returned value will be
        ' negative unless the given abscissa lies outside the force field, in
        ' which case the function returns zero.
        Dim IndexToLeft As Int32
        Dim Slope As Double
        ' Check if the slug is positioned outside the force field, to the left.
        If (Z > Ftable(1, 1)) Then
            InterpolateForceField = 0
            Exit Function
        End If
        ' Check if the slug is positioned outside the force field, to the right.
        If (Z < Ftable(FnumZ, 1)) Then
            InterpolateForceField = 0
            Exit Function
        End If
        ' Find the index of the abscissa in the force field table which lies just
        ' to the left of the given abscissa.  Note that this routine does not
        ' assume that the points in the table are equally-spaced.  It assumes only
        ' that they are in order of decreasing values, that is, each point is
        ' closer to the origin than the preceding point.
        For I As Int32 = 1 To (FnumZ - 1) Step 1
            If ((Z <= Ftable(I, 1)) And (Z > Ftable(I + 1, 1))) Then
                IndexToLeft = I
                Exit For
            End If
        Next I
        ' Let us check for the trivial case, where we are exactly at a data point.
        If (Z = Ftable(IndexToLeft, 1)) Then
            ' By chance, we are exactly at a data point.
            InterpolateForceField = Ftable(IndexToLeft, 2)
            Exit Function
        End If
        ' We deal here with the general case, where we are between two data points.
        Try
            Slope = (Ftable(IndexToLeft + 1, 2) - Ftable(IndexToLeft, 2)) / _
                (Ftable(IndexToLeft + 1, 1) - Ftable(IndexToLeft, 1))
        Catch ex As Exception
            MsgBox("Error in the force field table.  Two abscissas are the same.")
            InterpolateForceField = 0
            Exit Function
        End Try
        InterpolateForceField = Ftable(IndexToLeft, 2) + _
            (Slope * (Z - Ftable(IndexToLeft, 1)))
    End Function

End Class
```