A Treatise on Electrodynamics

G.R.Dixon

Contents

A Treatise On Electrodynamics

G.R.Dixon

Introduction.

This treatise considers electromagnetic phenomena that are rarely (or at best only briefly) discussed in the literature. Many of the topics were initially treated in individual articles featured on www.maxwellsociety.net.

Several reported results are produced by computer programs. And in selected cases the computer code is included in the appendices. The programming language is Visual Basic … available on most PCs.

The author has chosen to discuss a somewhat eclectic mix of topics, the hope being that readers who are well-grounded in traditional electromagnetic theory will find the treatments entertaining. Above all it is hoped that even seasoned physicists will find something novel and thought provoking.

GRD, Mesa, Arizona, July 2007.

1. Non-Relativistic Electrodynamics.

In this section it is assumed that particle maximum speeds are much less than c, the speed of light. Among other things, particle mechanical and electromagnetic masses are assumed to be constant attributes of a particle. In general, the relativistic term $(1-v^2/c^2)^{1/2}$ does not appear.

1.1. The Fields of a Point Charge.

Reference: Appendix 1.1_1

Several equivalent formulas for the electric (**E**) and magnetic (**B**) fields of a point charge have been derived over the years. A useful form for *computing* purposes is provided by Griffiths:

$$\overline{u} \equiv c\hat{r} - \overline{v},$$

$$\overline{E} = \frac{q}{4\pi\varepsilon_o}\frac{r}{(\overline{r} \bullet \overline{u})^3}[\overline{u}(c^2 - v^2) + \overline{r} \times (\overline{u} \times \overline{a})],$$

$$\overline{B} = \frac{1}{c}\hat{r} \times \overline{E}.$$

In these equations **r** is the vector from the point charge's *retarded* position to the field evaluation point; **v** is the retarded velocity; and **a** is the retarded acceleration. In brief, **r**, **v**, and **a** correspond to the retarded time, t_r.

In order to compute the fields at an arbitrary field evaluation point (which can be any point other than that occupied by the charge itself), the charge's past motion must be known. Provided with this information, a computer can readily approximate the retarded quantities by means of an iterating routine. And having done this, it can compute **E** and **B** directly.

An excellent subject for such an exercise consists of an oscillating point charge whose motion is

$$x = A\sin(\omega t).$$

It is useful to define the wavelength of this oscillation to be

$$\lambda \equiv \frac{2\pi c}{\omega}.$$

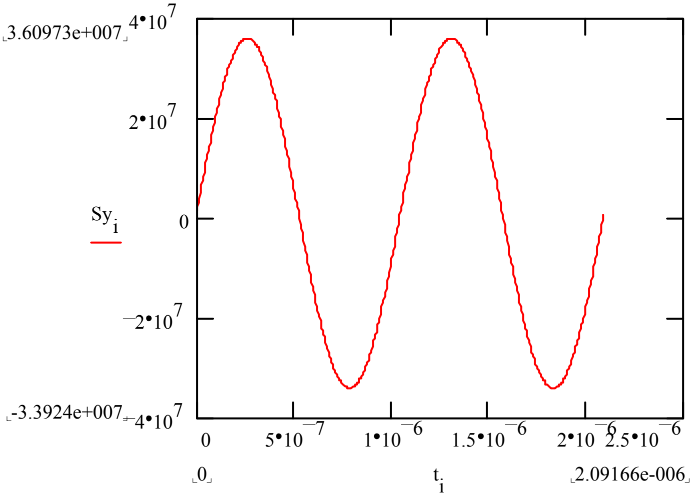
Let us consider 2 points on the y-axis, say one at y=.05λ and a second at y=λ. (The first point is sometimes said to be in the "near fields" region, and the second in the "far fields.")

If we are concerned with the field energy fluxes along the y-axis, we can compute E_x and B_z in each case and then the y-component of the Poynting vector,

$$S_y = \varepsilon_o c^2(-E_x B_z).$$

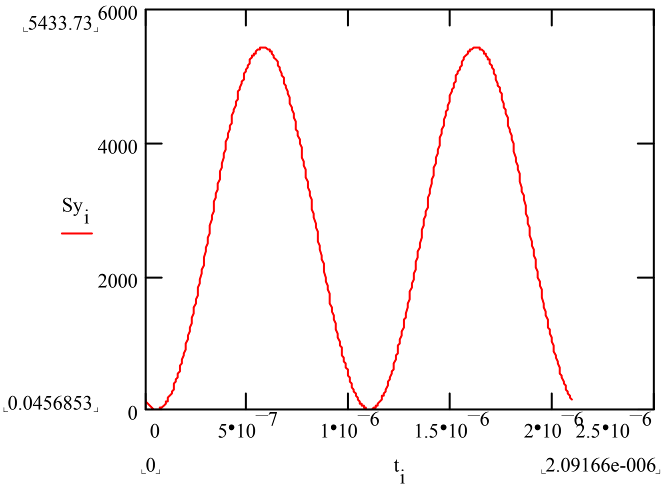
Fig. 1.1_1 plots S_y vs. time at y=.05λ. Fig. 1.1_2 does the same at y=λ.

Figure 1.1_1



S_y, Near Fields Region

Figure 1.1_2



S_y, Far Fields Region

It turns out that, in the near fields region, field energy primarily fluxes away from and back toward the charge. And, like the charge's motion, this energy flux is nearly sinusoidal. Energy considerations suggest that there is a correspondence between this flux and the power expended by the external agent that mechanically drives the charge. For reasons to be made clear shortly, we shall refer to this part of the agent-provided driving force as the "inertial" force. Note that the energy pouring into the fields in one part of a cycle flows back to the driving agent in a succeeding fraction of a cycle.

In the far fields region the energy flux is practically all outward and pulsed. Presumably part of the positive power, expended by the driving agent in one part of a cycle, *cannot be recouped* in a succeeding part of a cycle; it is permanently lost to infinite space. We shall refer to this part of the total agent force as the "radiative" force. In general a driving force may have both inertial and radiative parts:

$$\overline{\mathrm{F}} = \overline{\mathrm{F}}_{\mathrm{inert}} + \overline{\mathrm{F}}_{\mathrm{radiative}}\,.$$

Note the figures' implication that the inertial part is "conservative" in the sense that the net work *per cycle* expended by it is apparently zero. The radiative part is "non-conservative." The net work per cycle expended by it is greater than zero.

1.2. Spherical Shells of Charge.

Although the concept of a point charge is useful, it turns out that the field momentum of any *finite* point charge, moving at any velocity other than zero, is infinite! In order to avoid this singularity, it is useful to consider *spherical shells* of charge. (A finite point charge can then be considered to be the limit of such a distribution as the radius shrinks to zero.)

1.2.1. Modeling the Spherical Shell Using Small Point Charges.

Perhaps ironically, we can *approximate* a spherical shell of charge q as a distribution of smaller, equal point charges, preferably with each charge at the vertex of a symmetric polyhedron (said polyhedron inscribed in the sphere). Each such point charge will experience a small force in the net field of the other constituent charges. And the entire distribution will experience a *net* force (which may or may not be zero) *in its own field*.

Ideally the spherical shell would be modeled as an *infinite* number of *infinitesimal* point charges. For computing purposes we must settle for a *finite* number of small but finite charges.

1.2.2. Implied Agent Forces.

In considering an oscillating charge it was implicitly assumed that some external agent exerted a mechanical (i.e., non-electromagnetic) force on the oscillating charge. In a similar fashion, an external agent must (often implicitly) exert a mechanical force on each small point charge comprising a constant-radius spherical shell … *even when the shell is at rest.* For when the shell is at rest, each constituent point charge experiences a radially outward-pointing electric force in the net electric field of all the other charges. If this force is not mechanically counteracted in every case, the shell's radius (or shape) cannot remain constant. In effect a constituent point charge exerts an outward mechanical force on the constraining agent and the agent exerts an equal and oppositely directed mechanical force on the charge. One might even go further and suggest that it is the charge/field that exerts the force on the agent. The charge might be thought of as the "mechanical hook" whereby the agent interacts with the field.

1.2.3. Net Forces when $\underline{a} = 0$.

When a spherical shell has zero acceleration (and hence constant velocity) the agent forces that hold the shell together *sum to zero*. Thus no work is expended by the agent to maintain the spherical distribution of infinitesimal point charges.

1.2.4. Net Forces when d<u>a</u>/dt=0 (and when <u>a</u> is Nonzero).

When d**a**/dt=0 and **a** is nonzero but constant, then owing to time delays the forces between a spherical shell's diametrically opposed point charges *do not consistently sum to zero*. Computed solutions for the fields of a *finite* number of small but finite point charges indicate that *the charges (and hence the agent, that holds the shell together) collectively experience a net electric force pointing opposite to* ***a.*** In order to maintain the distribution's shape, the driving agent must counteract these acceleration-opposing forces *in addition* to counteracting the omnipresent repulsive forces. And although the repulsive forces sum to zero, the acceleration-opposing forces do *not*! Consequently the driving agent, which counteracts this net electric force, *must exert a net force in the same direction as* ***a***. In effect the spherical shell exhibits an *inertia* that opposes its acceleration. For future reference we shall refer to the acceleration-opposing net force, that the shell experiences *in its own field*, as the *inertial reaction* force. And (as already suggested), the agent's counteraction to this force is dubbed the *inertial* force. The net nonzero force that an external agent must exert when **a** is nonzero of course brings Newton's 2nd law to mind.

1.2.5. Net Forces when da/dt is Nonzero.

Unlike neutral matter, in the case of a spherical shell of *charge* there is yet an additional, reactive force component on each charge increment when d**a**/dt is nonzero! And here again the driving agent must counteract these myriad tiny forces if the distribution's shape is to be maintained. It turns out that this particular component of the total reaction force points in the *same* direction as d**a**/dt. Thus the driving agent's net counteraction must point *opposite* to d**a**/dt. This force, that the charges experience in their own net electric field when d**a**/dt is nonzero, is usually called the *radiation* reaction force. And the agent's counteraction constitutes the *radiative* component of the total agent force.

As will be discussed, a noteworthy difference between the inertial and radiative agent force components is that (a) the magnitude of the inertial component depends on the spherical shell's radius, whereas (b) the magnitude of the radiative component is independent of the radius (and more generally of the distribution's shape).

1.3. Electromagnetic Mass.

1.3.1. Charge/Field Electromagnetic Momentum.

The quantity **S**/c^2 (where **S** is the Poynting vector) has the dimensions of momentum per unit volume. And it is readily shown that the total momentum, in the electromagnetic field of a spherical shell of charge whose velocity is constant, is $q^2\mathbf{v}/6\pi\varepsilon_o Rc^2$ where R is the shell's radius. This suggests that an "electromagnetic mass" be assigned to the charge/field:

$$m_{elecmag} \equiv \frac{q^2}{6\pi\varepsilon_o Rc^2}.$$

Note that $m_{elecmag}$ is inversely proportional to R. Thus a finite *point* charge (with zero R) has *infinite* electromagnetic mass!

1.3.2. Spherical Shell with Constant Velocity.

Since the interactive forces among all the constituent point charges comprising a spherical shell sum to zero when **v** is constant, the agent counteractions sum to zero. This zero force translates to zero impulse over any interval of time, and the momentum in the field accordingly remains a constant $m_{elecmag}\mathbf{v}$.

1.3.3. Spherical Shell with Constant a.

If **a** is constant, then the driving agent must exert a constant *inertial* force component in the same direction as **a**. A computer program indicates that this force does indeed equal $m_{elecmag}\mathbf{a}$. If this force is parallel to **v** (the shell's velocity) then the agent does positive or negative work on the charge/field in any given time interval. In the positive case the momentum in the field increases with time; momentum (and energy) flows from the agent out into the field. In the negative case the field momentum decreases with time; momentum flows in from the field to the agent. In general, when **a** is constant, the law of motion is

$$\overline{F}_{inert} = m_{elecmag}\overline{a}\,.$$

1.3.4. Spherical Shell with Nonzero d<u>a</u>/dt.

As previously pointed out, in the case of a distribution of *charge* there is also a net reactive force component when d**a**/dt is nonzero. And the driving agent must counteract the net d**a**/dt-induced "radiation reaction" component of the total reactive force (as well as any nonzero **a**-induced or inertial reaction components) in order to maintain the distribution's shape. In the case of spherical shells of charge, the d**a**/dt-induced "radiation reaction" force is *independent* of the shell radius:

$$\overline{F}_{RadReact} = \frac{q^2}{6\pi\varepsilon_o c^3}\frac{d\overline{a}}{dt}\,.$$

The part of the total agent force, expended to *counteract* this radiation part of the net reaction force, is

$$\overline{F}_{radiative} = -\frac{q^2}{6\pi\varepsilon_o c^3}\frac{d\overline{a}}{dt}\,.$$

Note that whenever $\mathbf{F}_{\text{radiative}}$ and **v** point in the same direction, then their dot product is positive and the agent is expending power. A computer program (*Reference Appendix 1.3.4_1)* indicates that*, in the case of periodic motion, the power expended per cycle by* $F_{\text{radiative}}$ *equals the electromagnetic energy flux per cycle out through an enclosing surface.*

1.4. "Neutral Matter".

1.4.1. An "Equivalence Principle."

In the case of a spherical shell, any d**a**/dt-induced radiation reaction force is independent of the shell's radius. Thus given some shell of radius R, we may increase or decrease R (keeping the charge constant) and the shell will experience the same radiation reaction force to a given d**a**/dt. Given two *concentric* shells with different radii, we may increase/decrease the radius of either one so that the radii are equal. In this case the combination will experience the same radiation reaction force as a single shell whose charge equals the sum of the two original charges.

1.4.2. Atoms.

In Newtonian mechanics, which deals with "neutral matter," there is no d**a**/dt-related component to the agent force. Newton's 2nd law states simply that

$$\overline{F} = m_{mech}\overline{a}\,.$$

(Here we have written m_{mech} to signify "mechanical mass" … the inertial mass of an *uncharged* particle, to the extent such a concept corresponds to anything in the real world).

Since all "uncharged" atoms consist of equal amounts of positive and negative charge, and since the radiation reaction force is proportional to q^2 (and not to q), it might be wondered why **F**=m**a** works so well in the case of atoms (and more generally in cases of objects composed of atoms). In brief, why did Newton find that **F**=m**a** works in cases of "neutral matter," when neutral matter is in fact composed of electric charges?

We can appreciate why this is so if we model a typical atom as a central, spherical shell of positive charge, surrounded by a larger spherical shell of equal negative charge. At first glance, since $F_{radiative}$ is proportional to q^2, it might seem that (a) when such a distribution is vibrated as a whole, then (b) there should be double the non-conservative work expended per cycle than is required to vibrate either shell alone.

Here, however, we encounter the Equivalence Principle suggested above. Assuming we have equal magnitude charges, two such *superimposed* shells amount to no charge at all! By inference no power is expended to oscillate the concentric pair when their radii are *unequal*. The only requirement for this equivalence is that the two shells be concentric during the oscillation.

1.4.3. Limitations to the "Equivalence Principle."

Note that the "Equivalence Principle" does not apply to the *conservative* (inertial) power expended to counteract the *inertial* reaction force (which is proportional to 1/R). Here two *superimposed*, equal radii shells are *not* equivalent to concentric shells with different radii. Assuming the charge magnitudes are equal, the inner shell has a greater electromagnetic mass.

It is also worth noting that two oppositely signed, oscillated shells with a net charge of zero *will* radiate *when their centers do not coincide*. As will be discussed later, the agent's counteraction of the *interactive* force between these two shells will include a radiative component. It is only when the shells are concentric (as in our simplistic model of an oscillated atom) that they are equivalent to a single shell that has zero charge density.

Of course if we have two concentric spheres of *same-sign* charge then, so far as power expended per cycle to counteract the radiation reaction force goes, the pair is equivalent to a single shell of arbitrary radius and with a charge equal to the sum of the two original charges. Indeed the radius can be arbitrarily small, the implication being that the same *radiated* power occurs when a finite radius shell and an equal-magnitude *point charge* are vibrated. It is only in the case of the *conservative* (or inertial) power expenditure that the 1/R dependence of $m_{elecmag}$ results in a point charge singularity.

1.5. The Laws of Motion.

We may summarize the laws of motion for uncharged and charged particles as follows.

Newton's 1st law essentially provides a recipe for determining whether we are viewing things from an inertial or a non-inertial frame of reference. As such it is universally valid.

Newton's 2^{nd} law, in the case of uncharged particles (neutrons, atoms, etc.) is the familiar

$$\overline{F} = m_{mech}\overline{a} .$$

If a particle is charged, and in particular if such charge is modeled as a spherical shell, then we must add a d**a**/dt term to **F**=m_{mech}**a**:

$$\overline{F} = m_{elecmag}\overline{a} - \frac{q^2}{6\pi\varepsilon_o c^3}\frac{d\overline{a}}{dt} .$$

And, if the particle is a combination of linked mechanical and electromagnetic mass, then we would write

$$\overline{F} = (m_{mech} + m_{elecmag})\overline{a} - \frac{q^2}{6\pi\varepsilon_o c^3}\frac{d\overline{a}}{dt} .$$

Newton's 3^{rd} law is generally the same for neutral (uncharged) and charged particles. In both cases the **a**-induced and any d**a**/dt-induced reaction forces are equal and oppositely directed to **F** when a distribution's shape is held constant. In the case of charged particles, both the inertial and radiation reaction forces can be said to be electric forces that the charge experiences in its own electric field. And to the extent *every uncharged particle is composed of constituent charged particles*, Newton's equal and oppositely directed inertial reaction force is ultimately *always* an electric force. In effect "inertia" … something of a mystery in Newton's day … is seen to be implicit in Maxwell's equations and the Lorentz force law.

But note that while q**E** might explain the reaction forces, *it is upon the driving agent that these forces always act*. To reiterate, the charge can be thought of as the "mechanical hook" whereby the driving agent latches onto and interacts with the electromagnetic field, the two exchanging momentum and energy in the process.

1.6. Periodic Motion.

1.6.1. Reaction Forces.

The inertial reaction force, in the case of an uncharged particle (such as an atom), is simply $-m_{mech}\mathbf{a}$. If the particle's shape is kept constant, then a driving agent must counteract this force, and thus

$$\overline{F} = m_{mech}\overline{a} .$$

If the particle is charged, then there may be a second component to the total reaction force. This second component is called the radiation reaction force:

$$\overline{F}_{RadReact} = \frac{q^2}{6\pi\varepsilon_o c^3}\frac{d\overline{a}}{dt} .$$

And we have called the driving agent’s counteraction the *radiative* part of the total agent force.

The net force, required to counteract both the inertial and radiation reaction forces in cases of charged particles with some mechanical mass, is

$$\overline{F} = (m_{mech} + m_{elecmag})\overline{a} - \frac{q^2}{6\pi\varepsilon_o c^3}\frac{d\overline{a}}{dt} .$$

Let us, for discussion purposes, suppose that a charged particle with zero mechanical mass is being driven sinusoidally:

$$x = A\sin(\omega t).$$

If the charge is distributed in a spherical shell, then we have for the driving agent force:

$$\overline{F} = \frac{q^2}{6\pi\varepsilon_o Rc^2}\overline{a} - \frac{q^2}{6\pi\varepsilon_o c^3}\frac{d\overline{a}}{dt} = \overline{F}_{inert} + \overline{F}_{radiative}.$$

For the specified motion,

$$F = -m_{elecmag}\omega^2 A\sin(\omega t) + \alpha\omega^3 A\cos(\omega t), \alpha \equiv \frac{q^2}{6\pi\varepsilon_o c^3}$$

The rate at which the driving agent does work is

$$P = \overline{F} \bullet \overline{v} = \overline{F}_{inert} \bullet \overline{v} + \overline{F}_{radiative} \bullet \overline{v}.$$

And the net work per cycle done by **F**$_{inert}$ is zero:

$$\int_0^{2\pi/\omega} -m_{elecmag}\omega^2 A\sin(\omega t)\omega A\cos(\omega t)dt = 0$$

Thus we need only concern ourselves with the work per cycle done by **F**radiative. And, since this work is independent of the spherical shell's radius, we can consider a point charge.

1.6.2. Energy Conservation.

The work done per cycle by **F**radiative is always positive:

$$\int_0^{2\pi/\omega} \alpha\omega^3 A\cos(\omega t)\omega A\cos(\omega t) = \alpha\pi\omega^3 A^2 .$$

Energy conservation suggests that this should equal the net energy flux per cycle out through an enclosing surface. And a computer program indicates that this is indeed the case. For example, using a spherical surface that encloses the oscillation, the point charge field solutions can be used to compute **S** at points on the surface and at a sequence of epochs. Numerical integration then demonstrates that

$$\int_0^{2\pi/\omega} dt \oint \overline{S} \bullet d\overline{a} = \alpha\pi\omega^3 A^2 .$$

1.7. Interactive Forces.

Although we have discussed spherical shells of charge, it is generally true that all of the charge increments in *any* distribution of like-sign charge repel the other increments. If the distribution is to maintain its shape then these electric forces must be mechanically counteracted by some agent (usually implied).

Similarly, the distribution will react against an *accelerating* agent with a force proportional to –**a**. As in the case of a spherical shell, the proportionality constant constitutes the distribution's electromagnetic mass. Note that, given some amount of charge, $m_{elecmag}$ will vary from shape to shape. (Typical example: same-charge spherical shells with different radii.)

Finally, *regardless of the shape*, a distribution will react with a force of (αd**a**/dt) when **a** is not constant. And the driving agent must counteract this force too if the distribution is to maintain its shape. Whenever d**a**/dt is nonzero there will be radiation.

What is true for any single distribution is also true for two or more distributions separated by space. Each increment of charge interacts not only with every other increment in its own distribution, but also with every increment in the other distribution(s). It is convenient to refer to the *net* force, that one distribution experiences in the field of other(s), as the *interactive* force. The net force that a distribution experiences in its own field is then dubbed the *self*-force.

An external agent must mechanically counteract both the interactive and the self-forces of an overall "shape" if that shape is to remain static. (Special case: two spherical shells of charge at rest in a given inertial frame of reference.) Interestingly enough, if the agent allows the distance between two distributions to vary, then the interactive force does not invariably vary in a simple way with the separation. As will be demonstrated, if the two distributions oscillate in phase then the interactive force of attraction or repulsion may rise and fall in cyclic fashion as the separation is increased. The "wavelength" plays a prominent role regarding what separation produces the maximum interactive force, and this explains in part why certain length rod antennas are more effective radiators than others are.

The configurations (or distributions) and motions that can be imagined are of course practically infinite. But for purposes of computation each distribution can always be approximated as an array of small but finite point charges. And the point charge field solutions, together with the Lorentz force law, can always be used to compute the self-forces and the interactive forces.

In cases of periodic motion it is always found that the net work per cycle expended by the driving agent, to counteract both the self forces and the interactive forces, equals the field energy flux per cycle out through an enclosing surface. The only caveat here is that the *net* electric and magnetic fields, at points on the enclosing surface, must be computed prior to computing the Poynting vector.

2. Relativistic Electrodynamics.

In this section, cases are considered where maximum particle speeds are close to c.

Two relativistic precepts in the dynamics of *uncharged* particles are that (1) a particle's mechanical mass is a function of its speed, and (2) Newton's 2nd law, as he originally presented it, is relativistically valid:

$$m_{mech}(v) = \gamma m_{mech}(0), \gamma \equiv (1 - v^2 / c^2)^{-1/2}$$

$$\overline{F}_{inert} = \frac{d}{dt}(m_{mech}\overline{v}) = m_{mech}\overline{a} + \overline{v}\frac{d}{dt}(m_{mech}).$$

Electromagnetic mass depends on v in the same manner as mechanical mass does.

In the case of a charged particle, the relativistically correct formula for the radiation reaction force is:

$$\overline{F}_{Rad\,React} = \alpha\gamma^4\left(\frac{d\overline{a}}{dt} + \frac{3}{4}\overline{a}\frac{d\gamma^4}{dt}\right).$$

2.1. The Relativistically Rigorous Radiation Reaction Force.

If one compares the energy flux per cycle time, out through a surface enclosing an oscillating point charge, with the agent work per cycle expended to counteract $\alpha d\mathbf{a}/dt$ (where $\alpha = q^2/6\pi\varepsilon_o c^3$), then as v_{max} (or ωA) approaches c, significant inequalities are encountered. Since the point charge field solutions (used to compute **S** at points on the enclosing surface) are relativistically correct, it must be that the non-relativistic expression for the radiation reaction force requires adjustment. The relativistically correct formula in one dimension is

$$F_{Rad\,React} = \gamma^4\alpha\left(\frac{da}{dt} + \frac{3\gamma^2 va^2}{c^2}\right).$$

When this formula is used, then it is found that, for *all* $\omega A<c$,

$$\int_0^{2\pi/\omega} F_{radiative} v dt = \int_0^{2\pi/\omega} dt \oint \bar{S} \bullet d\bar{a}.$$

In the case of a charge going in a circle at constant speed, the vector equation for $\mathbf{F}_{RadReact}$ simplifies to

$$\bar{F}_{RadReact} = \alpha\gamma^4 \frac{d\bar{a}}{dt}.$$

Here γ is a constant and d**a**/dt points opposite to **v** at all times. Thus a driving agent, in counteracting $\mathbf{F}_{RadReact}$, must exert a constant force in the *same direction* as **v**. In order to keep the orbit from decaying, the particle must be *driven* around the circle by a constant-magnitude, tangential mechanical force. Among other things, the rate at which the driving agent does work is constant in time.

As in the case of oscillating motion, the above results can be compared with the energy flux per cycle time through an enclosing surface. Here again Energy is conserved. Furthermore, it is found that the rate of field energy flux through the enclosing surface is *constant in time*, quite as the driving agent's power expenditure is.

2.2. Newton’s 2nd Law.

The pre-relativistic notion that inertial mass (both mechanical and electromagnetic) is an invariant attribute of a particle also requires adjustment. In relativity theory,

$$m_{mech}(v)=\gamma m_{mech}(0),\gamma\equiv(1-\frac{v^2}{c^2})^{-1/2},$$

$$m_{elecmag}(v)=\gamma m_{elecmag}(0).$$

The full equation of motion for a charged particle with nonzero mechanical mass then becomes

$$\overline{F}=\frac{d}{dt}(m\overline{v})-\alpha(\gamma^4\frac{d\overline{a}}{dt}+\frac{3}{4}\overline{a}\frac{d\gamma^4}{dt}).$$

(Here $m=m_{mech}+m_{elecmag}$.)

When **a** and **v** are parallel, then

$$\frac{d}{dt}(m\overline{v})=\gamma^3 m(0)\overline{a},$$

and when they are perpendicular, then

$$\frac{d}{dt}(m\overline{v}) = \gamma m(0)\overline{a}\,.$$

In the case of periodic motion along the x-axis, the first "conservative" term again results in zero work per cycle:

$$\int_0^{2\pi/\omega} \gamma^3 m(0)avdt = 0\,.$$

Only the work expended by the $F_{radiative}$ part of F is nonzero.

A useful term in many calculations is "P_{rad}," defined to be the instantaneous rate at which $F_{radiative}$ does work:

$$P_{rad} \equiv \overline{F}_{radiative} \bullet \overline{v}\,.$$

That is,

$$P_{rad} = -\frac{q^2}{6\pi\varepsilon_o c^3}(\gamma^4 \frac{d\overline{a}}{dt} + \frac{3}{4}\overline{a}\frac{d\gamma^4}{dt}) \bullet \overline{v}\,.$$

In the case of periodic motion,

$$\int_0^{2\pi/\omega} P_{rad} dt = \int_0^{2\pi/\omega} dt \oint \overline{S} \bullet d\overline{a} \ .$$

However, owing to time delays, one should *not* conclude that, at any moment,

$$P_{rad} = \oint \overline{S} \bullet d\overline{a} \, .$$

In the literature there *is* a formula for the rate at which *radiant* energy fluxes through an enclosing surface. It is known as the Larmor formula and has the form

$$P = \frac{q^2}{6\pi\varepsilon_o c^3} \gamma^6 [a^2 - (\frac{\overline{v}}{c} \times \overline{a})^2] \, .$$

In one-dimensional cases it is customary, on the strength of this formula, to say that P is proportional to a^2. In other words, the assertion is that accelerating charges invariably radiate. There are difficulties with this assertion, however. For example, let us consider an oscillatory motion, x=A sin(ωt). Some agent drives the oscillating particle, expending power in the amount Fv. Now according to Larmor the radiated power is maximum precisely when the acceleration is a maximum. But this happens to be when the velocity is zero!

Another objection to the Larmor formula is discussed in the next section.

2.3. A Non-Radiating, Accelerating Charge.

Reference: Appendix 2.3_1

A rather interesting fact is that a charged particle, subject to a constant force, accelerates but *does not radiate*! In order to prove this, we begin by noting that the driving agent's counteraction to $F_{RadReact}$, in 1 dimension, is

$$F_{radiative} = -\alpha\gamma^4\left(\frac{da}{dt} + \frac{3\gamma^2 va^2}{c^2}\right).$$

Clearly no radiation is emitted when $F_{radiative}=0$, or when

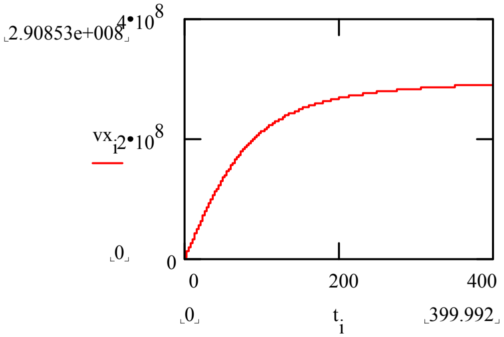
$$\frac{da}{dt} + \frac{3\gamma^2 va^2}{c^2} = 0.$$

One obvious solution is when v is constant (and a=0). But a second solution can be *computed* by imposing initial conditions for v(0) and a(0) and then performing the following algorithm:

```
t=0
Do
    $da/dt = -3\gamma^2 va^2/c^2$
    $v = v + a\ dt$
    $a = a + (da/dt)\ dt$
    $t = t + dt$
Loop until $v \sim c$
```

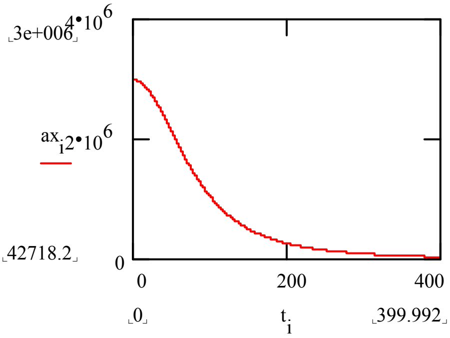
Figs. 2.3_1 and 2.3_2 plot v_x vs. t and a_x vs. t respectively. Note that v approaches c asymptotically and a approaches 0 asymptotically.

Figure 2.3_1



v_x vs. t

Figure 2.3_2



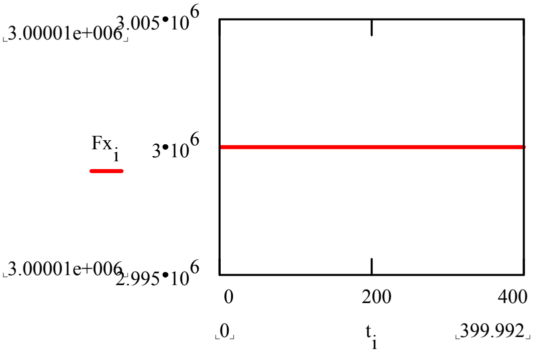
a_x vs. t

Now when $F_{radiative}$ is zero, the equation of motion (in 1 dimension) simplifies to

$$F = \gamma^3 m(0)a.$$

Substituting the computed values of v and a (and using some value for m(0)), Fig. 2.3_3 indicates that a plot of F results in a straight line. F is constant. Whence we conclude that a charged particle accelerates, but does not radiate, when subjected to a constant driving force!

Figure 2.3_3



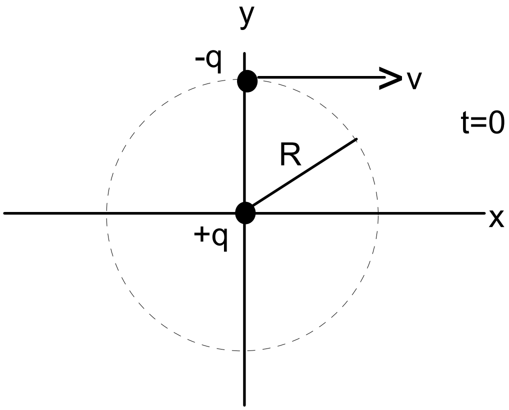
F_x vs. t

2.4. Length Contraction and Time Dilation.

Reference: Appendix 2.4_1

Fig. 2.4_1 depicts a 2-charge system. A positive charge is held fixed in inertial frame K. A negative charge travels at constant speed v around the positive charge.

Figure 2.4_1



A Simple, 2-Charge System

As previously pointed out, the negative charge will emit radiation at a constant rate. And a driving agent must exert a constant-magnitude tangential force in order to maintain the circular orbit. Let us assume that this radiative force is implicitly provided, and focus upon the radially inward *inertial* force,

$$\overline{F}_{inert} = \frac{\gamma m(0)v^2}{R}(-\hat{R}).$$

In the present case we shall assume that **F**$_{inert}$ is the *electric* (Coulombic) force that –q experiences in the electrostatic field of +q:

$$\frac{q^2}{4\pi\varepsilon_o R^2}(-\hat{R}) = \frac{\gamma m(0)v^2}{R}(-\hat{R}).$$

Let us now view matters from frame K’, which moves in the positive x-direction of K at constant speed u=v. At the moment shown, +q moves in the negative x’-direction of K’ at constant speed u, and –q is momentarily at rest on the y’-axis of K’. We would like to compute the path of –q in K’ by using Newton’s 2^{nd} law.

The force experienced by –q in frame K’ is generally specified by the *Lorentz force law,*

$$\overline{\mathbf{F}}' = -q(\overline{E}' + \overline{v}' \times \overline{B}').$$

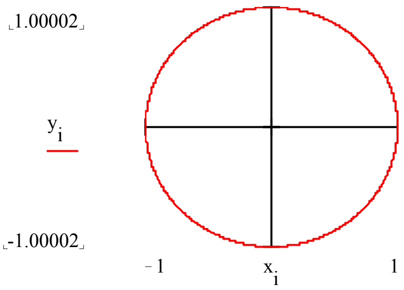
Here **E**’ and **B**’ are the electric and magnetic fields of +q, at the instantaneous position of –q, and **v**’ is the velocity of –q relative to K’. Note that although **B**=0 in K, +q moves in K’ and hence **B**’ is generally nonzero.

The equation of motion for –q in K’ thus becomes

$$-q(\overline{E}' + \overline{v}' \times \overline{B}') = \frac{d}{dt}(m\overline{v}').$$

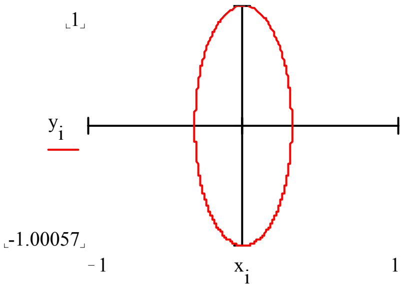
(Note that the radiative force, driving –q around the circle, is not ascribable to the fields of +q.) Starting with initial conditions, the motion of –q is readily computed, producing the following results: (1) the motion of –q is *quasi*-cycloidal, but –q cuts "in front of" and "in back of" +q at distances of only $(1-u^2/c^2)^{1/2}R$; (2) the time for one complete cycle in K' is *longer* than the cycle time of $\tau=2\pi R/v$ in K: $\tau'=(1-u^2/c^2)^{-1/2}\tau$. Figs. 2.4_2 and 2.4_3 depict the computed "shape" in K' for speeds of u=.01c and u=.95c respectively.

Figure 2.4_2



Shape in K', u=.01c

Figure 2.4_3



Shape in K’, u=.95c

These two results, *implicit in the dynamics of Maxwell-Lorentz-Newton*, are generally referred to as length contraction and time dilation. They appear to be generally applicable to all moving systems, including atoms. For example, measuring rods that translate parallel to their lengths are shorter than they are when at rest. And moving clocks run more slowly than their resting counterparts.

2.5. The Lorentz Transformations.

If one synchronizes the clocks in a given inertial frame on the assumption that light propagates with the one speed c in all directions relative to that frame (as experiment indicates is the case), then it is a simple matter to transform the space-time coordinates of events in one inertial frame to the corresponding coordinates in another inertial frame. The resulting, “Lorentz” transformation is slightly more complicated than the pre-relativistic “Galilean” transformation. But it takes into account length contraction and time dilation, and *in general* is consistent with the results predicted by Maxwell-Lorentz-Newton.

Knowing the transformation formulas for x, y, z and t, it is a matter of algebra to derive transformations for the components of velocity, acceleration, and even d**a**/dt. Transformations for the components of $\mathbf{F}_{\text{inertial}}$ and $\mathbf{F}_{\text{radiative}}$ follow, and even a set of general transformations for the components of the electromagnetic field vectors **E** and **B** can be derived.

With regard to the field vectors, it is often desired to have a “snapshot” of the fields in a given frame at some particular instant. It should perhaps be borne in mind, in this regard, that the field transformations transform a snapshot in frame K to the fields at many different instants in K’. However, a computer routine can be employed to build a “snapshot” of the fields in K’, given a knowledge of **E** and **B** in frame K.

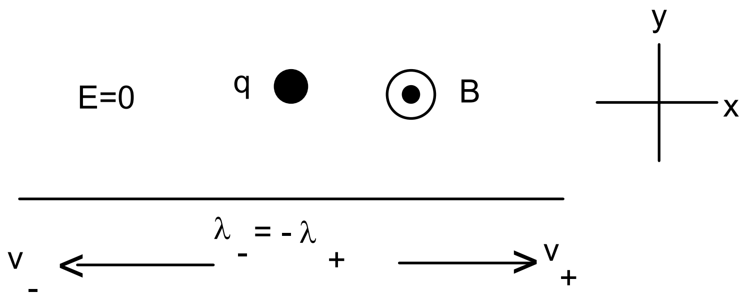
3. Neutral Current Loops.

Microscopic, electrically neutral current loops provide a convenient building block for modeling permanent, uncharged magnets. When the magnet spins around its North/South axis, each of the constituent loops translates in its plane. In the case of a disc-shaped magnet, d**B**/dt may equal zero both when the magnet is at rest and when it is spun. Yet in the spinning case there are theoretically nonzero electric fields.

3.1. Translation-Induced Electric Polarization.

Fig. 3.1_1 depicts a positive test charge, at rest in inertial frame K, above an infinitely long, uncharged line current. The current consists of equal-density, opposite-sign line charges with one sign moving to the left and the other to the right.

Figure 3.1_1



Test Charge and Uncharged Line Current

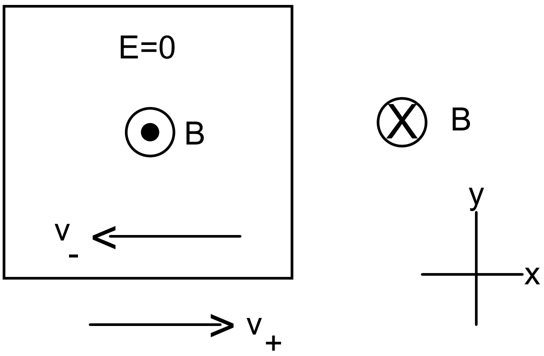
Having a net charge density of zero, the line current engenders no electric field. The test charge accordingly experiences no electric force. The line current does have a magnetic field. But, being at rest, the test charge also experiences zero magnetic force. The total Lorentz force on the test charge is therefore zero.

We would like to view this system from frame K’, moving in the positive x-direction of K at constant speed v. In K’ the positive line charge is at rest and the negative line charge moves even faster toward negative x’. The test charge also moves to the left. **B**’ again points out of the xy-plane, and now the test charge feels a magnetic force upward. Yet, according to the force transformation the *net* force on the test charge must be zero in K’ as it is in K.

The solution to this conundrum lies in the fact that, according to the Lorentz transformation, the net line charge density is not zero in K’. The (now-resting) positive line charge density is *less* than it is in K, and the negative line charge density is *greater* than it is in K. The net line charge density in K’ is therefore negative, and there is an **E**’ field that points toward the current. Thus in addition to the magnetic force acting upward, the test charge experiences an electric force downward. This electric force quite elegantly cancels the upward acting magnetic force; the net Lorentz force in K’ is zero quite as it is in K.

Let us take a segment of the line charge in Fig. 3.1_1 and bend it into a square loop as depicted in Fig. 3.1_2. We produce an uncharged current loop. **E**=0 everywhere and there is now a dipolar **B** field.

Figure 3.1_2



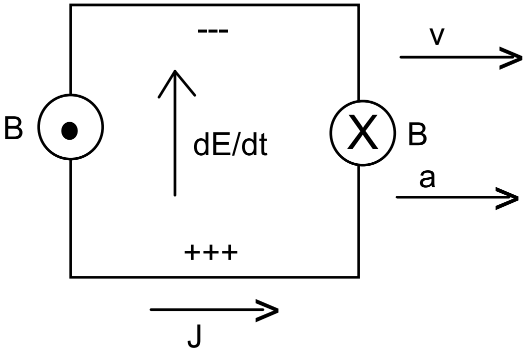
An Uncharged Current Loop

Viewed from frame K', the positive charge density in the bottom leg is less than it is in K, and the negative charge density is greater. The bottom leg thus has a net negative charge density in K'. The charge densities in the side legs are again zero. But the charge density in the top leg is *positive*. In K' the translating current loop is *electrically polarized*. This dipole engenders a dipolar **E**' field in K'. And of course **B**' will not only be nonzero, but at any given point in space it will vary in time.

3.2. Current Loop Electromagnetic Mass (or Inertia).

Fig. 3.2_1 depicts an *accelerating*, uncharged current loop. At any moment the degree of electric polarization is a function of the loop's speed. As v increases in time, so does the electric polarization.

Figure 3.2_1



Accelerating Current Loop

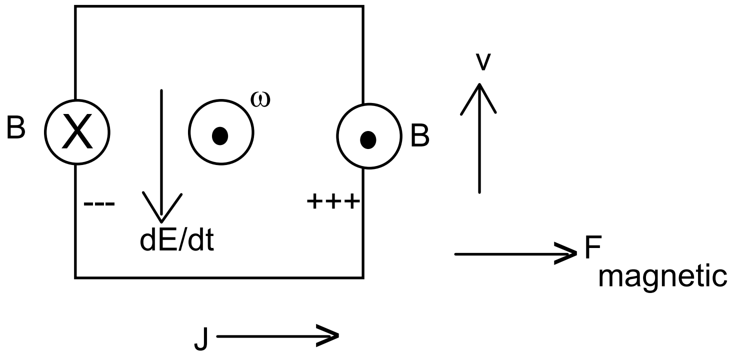
In Fig. 3.2_1 the magnitude of **E** at the loop's center is increasing in time. This induces magnetic field components in the leading and trailing legs as indicated. And the current in each leg experiences a magnetic force *opposite* to the loop's acceleration. The (unshown) agent causing the acceleration must counteract this magnetic force. The exerted force is proportional to **a**, and we may accordingly assign an electromagnetic mass to the loop such that (for v<<c)

$$\overline{F} = m_{elecmag}\overline{a}\,.$$

If the loop is one of the myriad microscopic loops in a disc-shaped magnet, then this inertial resistance to being (tangentially) accelerated will collectively contribute to the magnet's overall moment of inertia. Evidently, given two discs, identical in every way except that one is magnetized and the other is not, the magnetized disc will have a greater moment of inertia. Among other things, a greater torque will be needed in order to attain a given angular acceleration.

Newton's 2nd law of course requires that a current loop also react with an inertial reaction force when it is *radially* accelerated. Fig. 3.2_2 depicts a current loop being forced to travel around a circle (at constant speed). Note that the loop itself rotates CCW as it travels along the circular path.

Figure 3.2_2



Current Loop Going Around a Circular Path

In this case, although the dipolar **E** field has constant magnitude, its *direction* varies as the loop travels along the curved path (rotating in the process). This nonzero d**E**/dt induces **B** field components as indicated in the left and right legs. In each case the resulting magnetic force is radially *outward*; it constitutes the loop's inertial reaction force to radial acceleration. This force must be counteracted by the driving agent if the loop is to continue on the curved path.

4. Disc-Shaped Non-Conducting Magnets

4.1. Modeling a Disc-Shaped, Non-Conducting, Permanent Magnet.

A suggested model for a permanent, disc-shaped magnet is an array of *microscopic*, uncharged current loops. **E**=0 everywhere and the *net* magnetic field (of the overall magnet) is as expected.

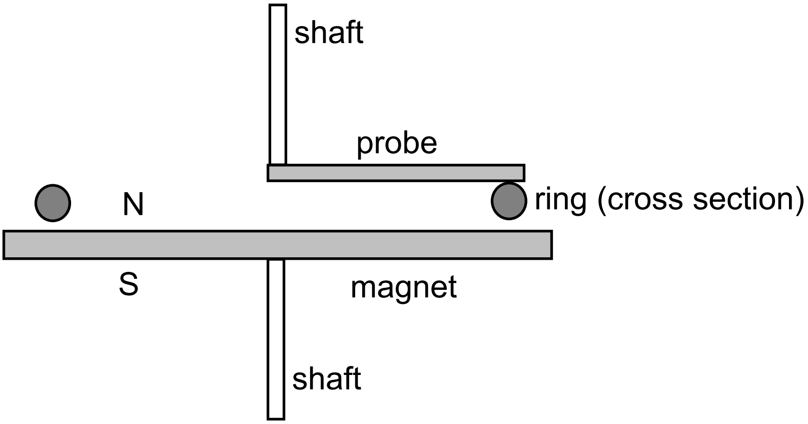
If such a magnet is spun about its axis of symmetry, then each constituent microscopic current loop translates in its plane. Each tiny loop accordingly has an electric dipole moment that points radially toward or away from the spin axis. The *net* electric field of these myriad microscopic current loops is conservative and has a *radial* component above and below the magnet.

It is of historic (pre-relativistic) note that, since d**B**/dt=0 in this case (both when the magnet is at rest and when it is spun), the existence of the electric field in the spinning case was either (a) unsuspected, or (b) if suspected, was of unknown origin. That is, the electric polarization of each microscopic current loop, when the parent magnet is spun, is strictly a relativistic effect.

4.2. Testing for the Spinning Magnet's Electric Field.

Fig. 4.2_1 depicts a conducting probe above a disc-shaped magnet. The probe is attached to a dielectric shaft, and it slides on a conducting ring. The magnet can be independently spun on another shaft.

Figure 4.2_1



Probe/Magnet Assembly

4.2.1. Spinning the Probe.

Let us first spin the probe above the resting magnet. Viewed from above, we shall spin the probe CCW. Conduction electrons in the probe are of course driven toward the probe shaft. An electroscope indicates that the ring has acquired a *positive* charge.

4.2.2. Spinning the Magnet.

We next keep the probe at rest and spin the magnet CCW. Each microscopic current loop in the magnet is electrically polarized, and the net electric field has radially *inward* pointing components. An electroscope should reveal a *negatively* charged ring.

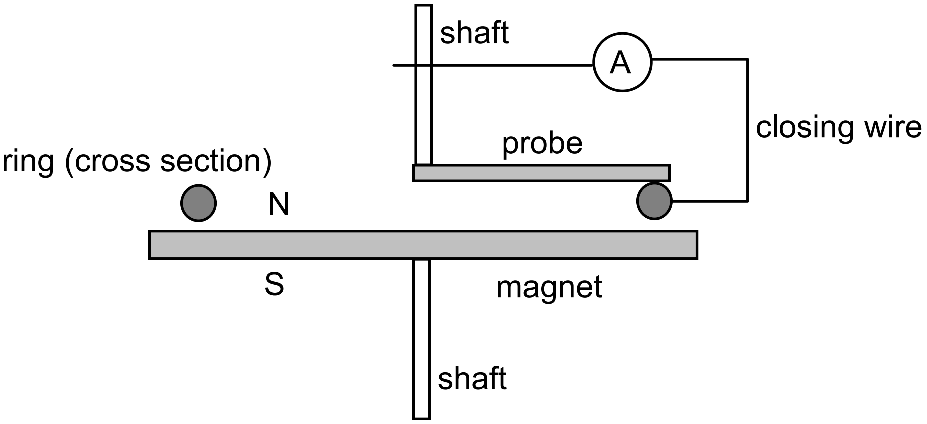
4.2.3. Probe and Magnet Spun Together.

The electric forces on probe conduction electrons oppose the magnetic forces. To the extent they cancel completely, an electroscope should indicate *no charge* on the ring.

4.3. A Homopolar Generator.

Fig. 4.3_1 depicts the assembly of Fig. 4.2_1, but with a closing wire completing the shaft/probe/ring/closing wire circuit. (The shaft is now conducting. The closing wire slips on, but makes electrical contact with, the shaft.)

Figure 4.3_1



A Homopolar Generator

If the magnet is held at rest and the probe is spun, then there will of course be a current.

If the probe is held at rest and the magnet is spun, then no current should flow. This is because the spinning magnet's electric field is *conservative*, and the emf around the circuit is zero.

If the probe and magnet are spun in tandem, then a current can be expected. The magnetic and electric forces on probe conduction electrons hypothetically sum to zero, and the emf in this part of the circuit is zero. This leaves the emf in the closing wire/shaft part of the circuit unopposed and a current should flow CCW around the circuit.

5. Electric Field Energy and Stress Energy.

It is often stated that the electric field energy density is

$$u_E = \frac{\varepsilon_o E^2}{2}.$$

The motivation for this formula is that if we initially have 2 distributions of charge (say 2 spherical shells of radius R) separated by distance D, and if we alter the distance between them, then the gain or loss of field energy precisely equals the positive or negative work we expend to make the change. (Note the implicit assumption that the dimensions of the distributions do not change.)

Quite often it is erroneously supposed that the field energy in a *single* spherical shell equals the work that must be expended to assemble the shell from *infinitely dispersed charge*. But it is readily appreciated that *more* work than the energy in the final field must be expended, in order to assemble the shell.

This can best be appreciated by considering a small change from radius r to radius r-dr. Let us divide the initial surface into N increments. We do a certain amount of work to push the N increments a distance dr toward the center. But we end up with overlaps at the increment fringes! In order to re-establish spherical symmetry, we must *compress* each increment in its plane. Having done that, we find that the gain in field energy equals the work done to push the N increments inward a distance dr. The work done to *compress* the N increments is “hidden” in elastic stresses in the surface of the shell.

It is not difficult to show that the total *field* energy in a shell of charge q and radius R is

$$U_E = \frac{q^2}{8\pi\varepsilon_o R}.$$

How shall we calculate the *stress* energy? The electromagnetic mass is $q^2/6\pi\varepsilon_o Rc^2$, and hence the shell's total energy is

$$m_{elecmag}c^2 = \frac{q^2}{6\pi\varepsilon_o R}.$$

Thus the stress energy is theoretically

$$\frac{q^2}{6\pi\varepsilon_o R} - \frac{q^2}{8\pi\varepsilon_o R} = \frac{q^2}{24\pi\varepsilon_o R}.$$

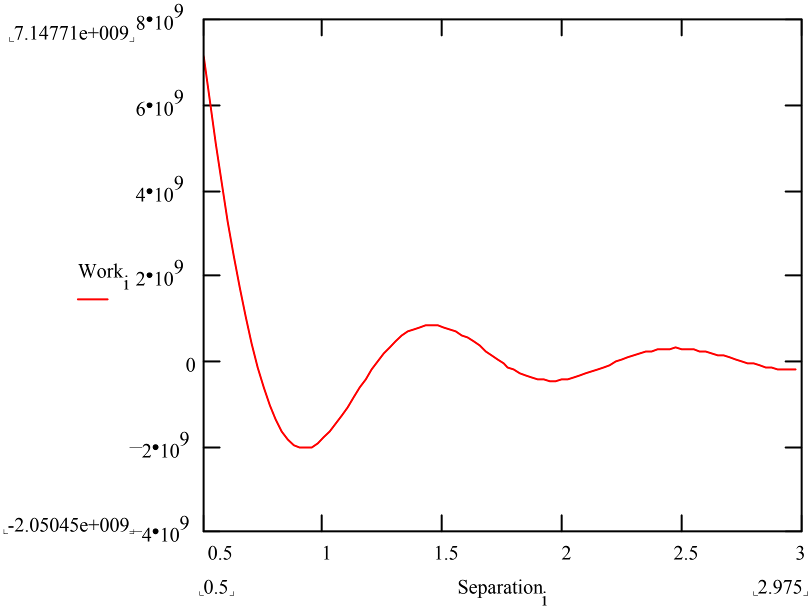
For a time it was believed that the total work needed to assemble a shell from infinitely dispersed charge must just equal the work in the final field, which is to say the work expended to push the charge increments in without compressing them. Poincare was first to recognize that the stresses in *any* distribution of charge also constitute a form of electromagnetic energy.

6. Interacting Charges.

Given two spherical shells of charge, q1 and q2 on the x-axis and at separation L, the charges interact. Each charge increment in q1 experiences a force from every charge increment in q2, and vice versa. Some agent must counteract these interactive forces if the radii are to remain constant. And of course the agent must also counteract the self-forces.

If the charges oscillate in phase, then the constraining agent may expend a net amount of work per cycle to counteract the interactive forces. It turns out that, when the charges oscillate, this work per cycle does not attenuate monotonically with increasing separation. Fig. 6_1 shows the actual, computed work per cycle over the range of separations $.5\lambda \le L \le 3\lambda$. *Reference Appendix 6_1.* Note that at selected separations the work per cycle is actually *negative*. At these separations the interactive forces collectively do a net amount of work per cycle *on* the driving agent! Of course the driving agent *always* does a positive amount of work per cycle in the course of counteracting the radiation reaction forces.

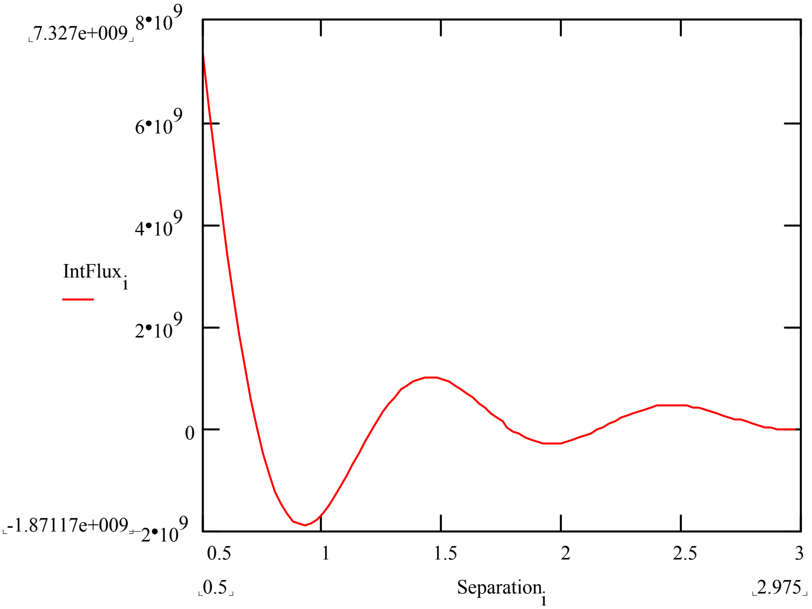
Figure 6_1



Work per Cycle to Counteract Interactive Forces

As might be expected, we can compute the energy flux per cycle time through an enclosing surface, at various values of L. *Reference Appendix 6_2.* If we subtract out the work per cycle expended to counteract the radiation reaction force, then we should get the same result as the work expended to counteract the interactive forces. Fig. 6_2 plots the computed energy flux per cycle time, minus the radiative force's work per cycle, again over the range $.5\lambda \leq L \leq 3\lambda$. Note that the plot is virtually identical to Fig. 6_1. The negative flux at selected values of L indicates that this part of the total energy flux per cycle actually flows *in* through the surface!

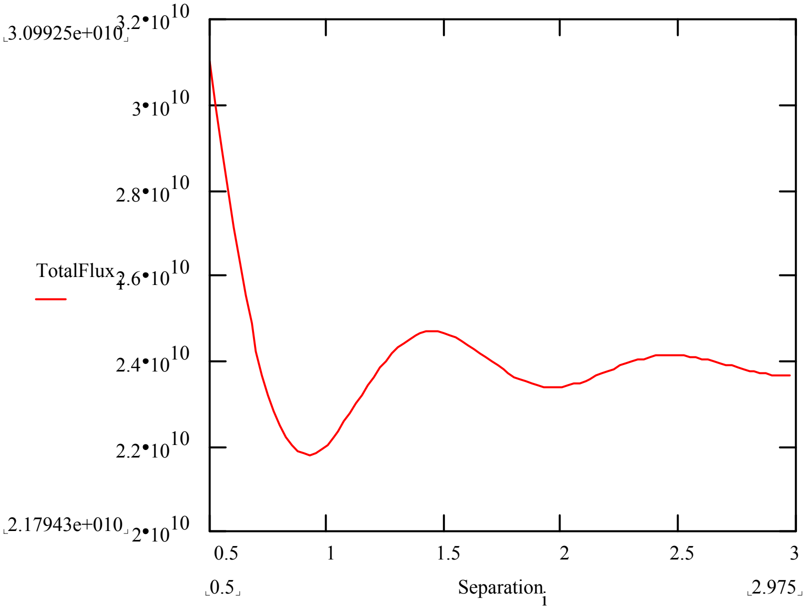
Figure 6_2



Energy Flux per Cycle Time, Interactive Flux Only

Figure 6_3 shows the *total* energy flux per cycle time through the enclosing surface. Note here that it is positive for *all* values of L. (Here the flux contains both the interactive and radiative components.)

Figure 6_3



Total Energy Flux per Cycle Time

In general we may conclude that when multiple charges move such as to radiate, then part of the radiated energy can be attributed to work done while counteracting the interactive forces. This scalar adds to the work per cycle done in the process of counteracting the radiation reaction forces.

Appendix 1.1_1

```
Private Sub cmdComputeS_Click()
'**********
'Compute the y component of the Poynting Vector, at a point
on the y-axis,
'at times throughout an oscillating charge cycle time.
'**********


'Physical and mathematical constants
    Const c As Double = 299792000#     'Speed of light
    Const epsilon0 As Double = 0.00000000000885
'Permittivity constant
    Const pi As Double = 3.14159265358979
    Const Steps As Long = 500     'Steps in an oscillation
    Const A As Double = 1     'Amplitude of oscillation
    Const omega As Double = 0.01 * c / A     'Angular
frequency
    Const tau As Double = 2 * pi / omega     'Period of
oscillation
    Const lambda As Double = c * tau     'Wavelength
'Use following value of Py
    Const Py As Double = 1 * lambda
```

```
    'Const Py As Double = 0.05 * lambda 'y-component of
observation point P

    Const deltat As Double = tau / Steps     'Time between
computed epochs
    Const q As Double = 1     'Charge of one coulomb

'Variables
    Dim i As Long     'Loop index
    Dim t(Steps) As Double     'Current time
    Dim tr As Double     'Retarded time
    Dim dt As Double     't - tr
    Dim dtmin As Double     'Minimum value for dt
    Dim dtmax As Double     'Maximum value for dt
    Dim xr As Double     'Retarded position (on x-axis)
    Dim vr As Double     'Retarded velocity
    Dim ar As Double     'Retarded acceleration
    Dim Drx As Double     'x-component of vector Dr
    Dim Dry As Double     'y-component of vector Dr
    Dim Dr As Double     'magnitude of vector Dr
    Dim ux As Double     'x-component of vector u
    Dim uy As Double     'y-component of vector u
    Dim Ex(Steps) As Double     'x-component of electric field
vector
```

```
    Dim Ey(Steps) As Double     'y-component of electric field
vector
    Dim Bz(Steps) As Double     'z-component of magnetic field
vector
    Dim Sy(Steps) As Double     'y-component of the Poynting
vector

'Turn on the hourglass.
    Screen.MousePointer = vbHourglass

'Set up the epochs.
    For i = 0 To Steps - 1
        t(i) = i * deltat
    Next i

'Compute retarded times, positions, velocities and
accelerations.
    For i = 0 To Steps - 1
        dtmin = 0
        dtmax = Sqr(A ^ 2 + Py ^ 2) / c
        Do
            dt = (dtmax + dtmin) / 2
            tr = t(i) - dt
            xr = A * Sin(omega * tr)
```

```
            If Abs(c * dt - Sqr(xr ^ 2 + Py ^ 2)) < 2 ^ (-30) Then Exit Do
            If c * dt - Sqr(xr ^ 2 + Py ^ 2) > 0 Then
                dtmax = dt
            Else
                dtmin = dt
            End If
        Loop
        vr = omega * A * Cos(omega * tr)
        ar = -(omega ^ 2) * A * Sin(omega * tr)

'Compute the components and magnitude of vector Dr.
        Drx = -xr
        Dry = Py
        Dr = Sqr(Drx ^ 2 + Dry ^ 2)
'Compute the components of vector u.
        ux = c * Drx / Dr - vr
        uy = c * Dry / Dr
'Compute the electric and magnetic field components
        Ex(i) = q / (4 * pi * epsilon0) * ((Dr / (Drx * ux + Dry * uy) ^ 3) * (ux * (c ^ 2 - vr ^ 2) - Dry * uy * ar))
        Ey(i) = q / (4 * pi * epsilon0) * ((Dr / (Drx * ux + Dry * uy) ^ 3) * (uy * (c ^ 2 - vr ^ 2) + Drx * uy * ar))
        Bz(i) = 1 / (Dr * c) * (Drx * Ey(i) - Dry * Ex(i))
'Compute the y-component of the Poynting Vector.
```

```
        Sy(i) = epsilon0 * c ^ 2 * (-Ex(i) * Bz(i))
    Next i
'Write values to a file for plotting.
    Open "c:\winmcad\Physics\POSSy.prn" For Output As 1
    For i = 0 To Steps - 1
        Write #1, t(i), Sy(i)
    Next i
    Close 1
'Turn off the hourglass
    Screen.MousePointer = vbArrow
    MsgBox ("Ready for plotting.")

End Sub
```

Appendix 1.3.4_1

```
Private Sub cmdEnergyConChk_Click()
'************************************
'Given an oscillating spherical shell of charge,
'Compute the total work per cycle done by the driving force, and the
'total energy flux per cycle through a spherical surface enclosing the
'oscillation site.  Display the two results for comparison.
'************************************
   Const c As Double = 299792000#  'speed of light
   Const epsilon0 As Double = 0.00000000000885 'permittivity constant
   Const pi As Double = 3.14159265358979
   Const Steps As Long = 1000
   Const ThetaSteps As Long = 100
   Const Amp As Double = 1 'amplitude
   Const omega As Double = 0.01 * c / Amp  'angular frequency
   Const freq As Double = omega / (2 * pi) 'frequency
   Const tau As Double = 1 / freq  'period
   Const deltat As Double = tau / Steps    'time increment
   Const q As Double = 1   'shell charge
```

```
    Const BigRadius As Double = 10 * Amp    'radius of sphere for energy flux
    Const Radius As Double = 0.005  'radius of spherical shell
    Const dtheta As Double = pi / ThetaSteps    'step in azimuth
    Const m As Double = q ^ 2 / (6 * pi * epsilon0 * Radius * c ^ 2)    'elecmag mass

    Dim theta As Double
    Dim x, v, gamma, a, dadt, Py, Px As Double
    Dim i, j As Long    'indexes
    Dim t As Double
    Dim tr As Double    'retarded time
    Dim dt As Double
    Dim dtmin As Double
    Dim dtmax As Double
    Dim xr As Double
    Dim vr As Double
    Dim ar As Double
    Dim Drx As Double
    Dim Dry As Double
    Dim Dr As Double
    Dim ux As Double
    Dim uy As Double
    Dim Ex As Double
    Dim Ey As Double
```

```
    Dim Bz As Double
    Dim Sx As Double
    Dim Sy As Double
    Dim Snormal As Double
    Dim unitx, unity, dA As Double
    Dim EFlux As Double
    Dim F, P As Double
    Dim WorkPerCycle As Double

    WorkPerCycle = 0
    EFlux = 0
    'For each time epoch...
    For i = 0 To Steps - 1
        '...compute the work done by the agent force during that time increment...
        t = i * deltat
        x = Amp * Sin(omega * t)
        v = omega * Amp * Cos(omega * t)
        a = -omega ^ 2 * Amp * Sin(omega * t)
        dadt = -omega ^ 3 * Amp * Cos(omega * t)
        gamma = 1 / Sqr(1 - v ^ 2 / c ^ 2)
        F = gamma ^ 3 * m * a - gamma ^ 4 * q ^ 2 / (6 * pi * epsilon0 * c ^ 3) * (dadt + 3 * gamma ^ 2 * v * a ^ 2 / c ^ 2)
        P = F * v
        '...and update the running total.
```

```
        WorkPerCycle = WorkPerCycle + P * deltat
    Next i
    'Then for each time epoch...
    For i = 0 To Steps - 1
        Debug.Print i
        t = i * deltat
        '...and for each azimuth angle
        For j = 0 To ThetaSteps - 1
            '...set up for flux dot product with area increment...
            theta = j * dtheta + dtheta / 2
            unitx = Cos(theta)
            unity = Sin(theta)
            Px = BigRadius * Cos(theta) 'point of Poynting
evauation
            Py = BigRadius * Sin(theta)
            dA = 2 * pi * Py * BigRadius * dtheta   'area of strip
            '...and find the fields.
            dtmin = 0
            dtmax = 4 * (Amp + BigRadius) / c
            Do
                dt = (dtmax + dtmin) / 2
                tr = t - dt
                xr = Amp * Sin(omega * tr)
                Drx = Px - xr
                Dry = Py
```

```
                Dr = Sqr(Drx ^ 2 + Dry ^ 2)
                If Abs(c * dt - Dr) < 2 ^ (-30) Then Exit Do
                If c * dt - Dr > 0 Then
                    dtmax = dt
                Else
                    dtmin = dt
                End If
            Loop
            vr = omega * Amp * Cos(omega * tr)
            ar = -(omega ^ 2) * Amp * Sin(omega * tr)
            ux = c * Drx / Dr - vr
            uy = c * Dry / Dr
            Ex = q / (4 * pi * epsilon0) * Dr / (Drx * ux + Dry * uy) ^ 3 * (ux * (c ^ 2 - vr ^ 2) - Dry * uy * ar)
            Ey = q / (4 * pi * epsilon0) * Dr / (Drx * ux + Dry * uy) ^ 3 * (uy * (c ^ 2 - vr ^ 2) + Drx * uy * ar)
            Bz = 1 / (c * Dr) * (Drx * Ey - Dry * Ex)
            'Then compute the components of the Poynting vector...
            Sx = epsilon0 * c ^ 2 * (Ey * Bz)
            Sy = epsilon0 * c ^ 2 * (-Ex * Bz)
            '...and form the dot product of it with the area strip
            Snormal = Sx * unitx + Sy * unity
            'Update the total energy flux through the spherical surface.
```

```
            EFlux = EFlux + Snormal * dA * deltat
        Next j
    Next i
    MsgBox ("W = " & WorkPerCycle & ", Erad =" & EFlux)
    Stop

End Sub
```

Appendix 2.3_1

```
Private Sub NonRadAccelQ_Click()
'**********
'Compute the x-components of velocity and acceleration when
'the negative of the rad react force acting on a charge
'is zero.  Substitute the values in Newton 2 in order to
'determine the acting force.  Output all the values for plotting.
'**********

    'Physical and mathematical constants
    Const c As Double = 299792000#  'Speed of light
    Const steps As Double = 500000  'Number of iterations
    Const dt As Double = 400 / steps    'Time between
computations
    Const vxinit As Double = 0  'Initial velocity
    Const axinit As Double = 3000000#   'Initial acceleration
    Const mem As Double = 1     'Electromagnetic mass

    Dim index As Long   'Loop counter
    Dim t(steps) As Double  'Time
    Dim vx(steps) As Double 'Velocity
    Dim ax(steps) As Double 'Acceleration
    Dim Fx(steps) As Double 'Force
```

```
    Dim daxdt As Double
    Dim gamma As Double
    Dim FxAv As Double  'Average force (to average out small numerical errors)

    'Set initial velocity and acceleration.
    index = 0
    vx(index) = vxinit
    ax(index) = axinit
    For index = 0 To steps - 1
        'Compute dax/dt and use this to compute next v and a.
        t(index) = index * dt
        gamma = 1 / Sqr(1 - vx(index) ^ 2 / c ^ 2)
        daxdt = -3 * gamma ^ 2 * vx(index) * ax(index) ^ 2 / c ^ 2
        vx(index + 1) = vx(index) + ax(index) * dt
        ax(index + 1) = ax(index) + daxdt * dt
    Next index
    'Open output files for writing plot data to.
    Open "c:\Winmcad\Physics\POSsoln23_1vx.prn" For Output As #1
    Open "c:\Winmcad\Physics\POSsoln23_1ax.prn" For Output As #2
    Open "c:\Winmcad\Physics\POSsoln23_1Fx.prn" For Output As #3
    'Write every tenth value of velocity and acceleration.
```

```
    For index = 0 To steps / 10 - 1
        Write #1, t(10 * index), vx(10 * index)
        Write #2, t(10 * index), ax(10 * index)
    Next index
    'Find the average force.
    FxAv = 0
    For index = 0 To steps - 1
        gamma = 1 / Sqr(1 - vx(index) ^ 2 / c ^ 2)
        Fx(index) = gamma ^ 3 * mem * ax(index)
        FxAv = FxAv + Fx(index)
    Next index
    FxAv = FxAv / steps
    'If percent difference between force and average force is less than .00001,
    'then the difference can be written off as a numerical error.
    For index = 0 To steps - 1
        If Abs((Fx(index) - FxAv) / Fx(index)) < 0.00001 Then
            Fx(index) = FxAv
        Else
            MsgBox ("FxAv = " & FxAv & ", Fx(" & index & ") = " & Fx(index))
        End If
    Next index
    'Write force.
    For index = 0 To steps / 10 - 1
```

```
        Write #3, t(10 * index), Fx(10 * index)
    Next index
    Close
    MsgBox ("Ready for plotting.")
    Stop

End Sub
```

Appendix 2.4_1

```
Option Explicit

Private Sub cmdLContractTDilation_Click()
'***************************************
'Given a charged satellite, circling an oppositely charged
'central body at constant speed, use Maxwell, Lorentz and
'Newton to compute the satellite's motion relative to another
'inertial frame. Output data reflecting the system shape
'for plotting purposes.
'***************************************
'Constants.
Const c As Double = 299792000# 'Speed of light
Const epsilon0 As Double = 0.00000000000885 'Permittivity constant
Const pi As Double = 3.14159265358979
Const Steps As Long = 1000000 'Steps in an oscillation

Const vp As Double = 0.01 * c 'Satellite speed in Kprime
‘Const vp as Double = .95 * c

Const vq As Double = vp 'Central body speed in K
Const Rp As Double = 1 'Orbital radius in Kprime
```

```
Const m0 As Double = 1 'Satellite rest mass
'Const taup As Double = 4 * pi * Rp / vp 'Orbital period in
Kprime
Const taup As Double = 2 * pi * Rp / vp
'Variables
Dim tau As Double 'Quasi-cycloid period in K
Dim deltat As Double 'Time between motion updates
Dim gammap As Double '1/sqr(1-v2^2/c^2)
Dim q As Double 'Computed charge for circular motion in
Kprime
Dim x As Double 'Computed satellite position in K, at time t
Dim xRepo(Steps) As Double 'x-vq*t (used to plot shape in K)
Dim y(Steps) As Double 'satellite y-position in K
Dim deltax, deltay As Double 'Displacements from central
body in K
Dim R As Double 'Distance from central body to satellite, in K
Dim theta As Double 'Angle between x-axis and R vector
Dim vx, vy, v, gamma As Double 'Satellite variables
Dim ax, ay As Double 'Satellite variables
Dim t As Double 'Current time in K
Dim Ex As Double 'x-component of electric field vector at
satellite in K
Dim Ey As Double 'y-component of electric field vector at
satellite in K
```

```
Dim Bz As Double 'z-component of magnetic field vector at satellite in K
Dim Fx, Fy As Double 'Lorentz force acting on satellite in K
Dim Denom, Num As Double 'Used in solutions for ax and ay
Dim index As Long
Dim C1 As Double 'Saves compute time
Dim SecondMax As Boolean
Dim ymax As Double 'Second max value of satellite y position

'Try an educated trial value for the quasi-cycloid period in K.
tau = 1 / Sqr(1 - vq ^ 2 / c ^ 2) * taup 'Trial value for period in K
deltat = tau / Steps
gammap = 1 / Sqr(1 - vq ^ 2 / c ^ 2)
'Find what size charge will cause circular motion in Kprime.
q = Sqr(4 * pi * epsilon0 * Rp * gammap * m0 * vp ^ 2)
C1 = q / (4 * pi * epsilon0)
'Initialize satellite variables (for t=0) in K.
x = 0
y(0) = Rp
vx = 0
vy = 0
v = 0
'Then repeatedly compute the repositioned satellite x-position (x-vq*t)
```

```
'and y-position for shape plotting.
For index = 0 To Steps - 1
    '1. Compute the fields.
    t = index * deltat
    'Use previously computed satellite positions.
    xRepo(index) = x - vq * t
    'R is always computed from the central body.
    deltax = xRepo(index)
    deltay = y(index)
    'R might vary in K.
    R = Sqr(deltax ^ 2 + deltay ^ 2)
    'theta is the angle between the x-axis and the current
satellite
    'position, with the central body at the junction of the angle
legs.
    If deltax = 0 Then
        If t = 0 Then
            theta = pi / 2
        Else
            theta = 3 * pi / 2
        End If
    Else
        If deltax < 0 And deltay >= 0 Then theta = pi / 2 + Atn(-
deltax / deltay)
```

```
        If deltax < 0 And deltay < 0 Then theta = pi + Atn(deltay /
deltax)
        If deltax >= 0 And deltay < 0 Then theta = 3 * pi / 2 +
Atn(deltax / -deltay)
        If deltax >= 0 And deltay >= 0 Then theta = Atn(deltay /
deltax)
    End If
    v = Sqr(vx ^ 2 + vy ^ 2)
    gamma = 1 / Sqr(1 - v ^ 2 / c ^ 2)
    Ex = C1 * Cos(theta) / R ^ 2 * (1 - vq ^ 2 / c ^ 2) / (Sqr(1 -
(vq / c * Sin(theta)) ^ 2)) ^ 3
    Ey = C1 * Sin(theta) / R ^ 2 * (1 - vq ^ 2 / c ^ 2) / (Sqr(1 -
(vq / c * Sin(theta)) ^ 2)) ^ 3
    Bz = 1 / c ^ 2 * vq * Ey
    '2. Compute the Lorentz force.
    Fx = -q * (Ex + vy * Bz)
    Fy = -q * (Ey - vx * Bz)
    '3. Compute the acceleration.
    'See the article for these formulas.
    Num = c ^ 2 * (c ^ 2 + gamma ^ 2 * vy ^ 2) * Fx - gamma ^ 2
* c ^ 2 * vx * vy * Fy
    Denom = (c ^ 2 + gamma ^ 2 * vy ^ 2) * (c ^ 2 * gamma *
m0 + gamma ^ 3 * m0 * vx ^ 2) - gamma ^ 5 * m0 * vx ^ 2 * vy
^ 2
    ax = Num / Denom
```

```
    Num = c ^ 2 * (c ^ 2 + gamma ^ 2 * vx ^ 2) * Fy - gamma ^ 2 * c ^ 2 * vx * vy * Fx
    Denom = (c ^ 2 + gamma ^ 2 * vx ^ 2) * (c ^ 2 * gamma * m0 + gamma ^ 3 * m0 * vy ^ 2) - gamma ^ 5 * m0 * vx ^ 2 * vy ^ 2
    ay = Num / Denom
    '4. Update the position and velocity.
    x = x + vx * deltat + 1 / 2 * ax * deltat ^ 2
    If index < Steps - 1 Then y(index + 1) = y(index) + vy * deltat + 1 / 2 * ay * deltat ^ 2
    vx = vx + ax * deltat
    vy = vy + ay * deltat
    '5. Iterate.
Next index
'Show the xRepo value at which the satellite cuts the x-axis.
index = 0
Do
    If Abs(y(index + 1)) < Abs(y(index)) And Abs(y(index + 1)) < Abs(y(index + 2)) Then
        MsgBox ("xRepo value where satellite cuts x-axis = " & Abs(xRepo(index + 1)))
        MsgBox ("xRepo/Rp = " & Abs(xRepo(index)) / Rp)
        MsgBox ("sqr(1-vq^2/c^2) = " & Sqr(1 - vq ^ 2 / c ^ 2))
        Exit Do
    End If
```

```
    index = index + 1
Loop
'Find out when the satellite again has its maximum positive value.
t = 0
index = 0
SecondMax = False
ymax = R
Do
    If y(index + 1) > y(index) Then
        ymax = y(index + 1)
        tau = t + deltat
    End If
    index = index + 1
    t = index * deltat
Loop Until index = Steps - 1
tau = t
MsgBox ("taup = " & taup & ", tau = " & tau & ", tau/taup = " & tau / taup)
MsgBox ("1/sqr(1-vq^2/c^2) = " & 1 / Sqr(1 - vq ^ 2 / c ^ 2))
'Output the xRepo and y satellite positions for shape plotting.
Open "c:\WINMCAD\Physics\XformElecDyn.prn" For Output As #1
For index = 0 To Steps / 1000 - 1
    Debug.Print index
```

```
    Write #1, xRepo(1000 * index), y(1000 * index)
Next index
Close
MsgBox ("Ready for plotting")

End Sub
```

Appendix 6_1

```
Private Sub cmdComputeWInt_Click()
'***********************
'Compute the work per cycle expended to counteract the
interactive
'forces on q1 and q2, at a range of separations.
'***********************


'Physical and mathematical constants
   Const c As Double = 299792000#     'Speed of light
   Const epsilon0 As Double = 0.00000000000885
'Permittivity constant
   Const pi As Double = 3.14159265358979
   Const Steps As Long = 100
   Const q As Double = 1
   Const lambda As Double = 0.1    'wavelength
   Const omega As Double = 2 * pi * c / lambda 'angular
frequency
   Const freq As Double = omega / (2 * pi) 'frequency
   Const tau As Double = 1 / freq  'period
   Const deltat As Double = tau / Steps    'time interval
between epochs
```

```
   Const Amp As Double = 0.1 * c / omega   'amplitude, in
meters
   Const LMin As Double = 0.5 * lambda 'minimum separation
   Const Lmax As Double = 3 * lambda   'maximum separation
   Const deltaL As Double = (Lmax - LMin) / Steps  'interval
between separations

'Variables
   Dim Px1, Px2 As Double   'coordinates of q1 and q2
   Dim Index, LIndex As Long  'Loop index
   Dim t(Steps) As Double     'Current time
   Dim tr As Double     'Retarded time
   Dim dt As Double     't - tr
   Dim dtmin As Double     'Minimum value for dt
   Dim dtmax As Double     'Maximum value for dt
   Dim xr1, xr2 As Double    'Retarded positions (on x-axis)
   Dim vrx1, vrx2 As Double   'Retarded velocities
   Dim arx1, arx2 As Double   'Retarded accelerations
   Dim Drx1, Drx2 As Double    'x-components of vectors Dr
   Dim Dr1, Dr2 As Double    'magnitudes of vectors Dr
   Dim ux1, ux2 As Double    'x-components of vectors u
   Dim Ex1(Steps), Ex2(Steps) As Double   'x-components of
electric field vectors
   Dim Fx1(Steps), Fx2(Steps), Fx(Steps) As Double
'interactive forces
```

```
    Dim WInt(Steps) As Double   'Work per cycle
    Dim L(Steps) As Double  'charge separation
    Dim vx As Double    'present charge velocity

    For LIndex = 0 To Steps - 1
        WInt(LIndex) = 0
    Next LIndex
    'For each separation...
    For LIndex = 0 To Steps - 1
        Debug.Print LIndex
        L(LIndex) = LMin + LIndex * deltaL
        'Compute the work per cycle to counteract the interactive
forces.
        For Index = 0 To Steps - 1
            'Compute the current positions of q1 and q2.
            t(Index) = Index * deltat
            Px1 = -L(LIndex) / 2 + Amp * Sin(omega * t(Index))
            Px2 = L(LIndex) / 2 + Amp * Sin(omega * t(Index))
            'Compute the electric field of q1, at the position of q2,
and
            'the electric field of q2 at the position of q1.

            'Compute the needed retarded quantities for q1.
            dtmin = 0
            dtmax = 2 * Lmax / c
```

```
Do
    dt = (dtmax + dtmin) / 2
    tr = t(Index) - dt
    xr1 = -L(LIndex) / 2 + Amp * Sin(omega * tr)
    Drx1 = Px2 - xr1
    Dr1 = Abs(Drx1)
    If Abs(c * dt - Dr1) < 2 ^ (-30) Then Exit Do
    If c * dt - Dr1 > 0 Then
        dtmax = dt
    Else
        dtmin = dt
    End If
Loop
vrx1 = omega * Amp * Cos(omega * tr)
arx1 = -(omega ^ 2) * Amp * Sin(omega * tr)
'Compute the needed retarded quantities for q2.
dtmin = 0
dtmax = 2 * Lmax / c
Do
    dt = (dtmax + dtmin) / 2
    tr = t(Index) - dt
    xr2 = L(LIndex) / 2 + Amp * Sin(omega * tr)
    Drx2 = Px1 - xr2
    Dr2 = Abs(Drx2)
    If Abs(c * dt - Dr2) < 2 ^ (-30) Then Exit Do
```

```
            If c * dt - Dr2 > 0 Then
                dtmax = dt
            Else
                dtmin = dt
            End If
        Loop
        vrx2 = omega * Amp * Cos(omega * tr)
        arx2 = -(omega ^ 2) * Amp * Sin(omega * tr)
        'Compute the components of vectors u.
        ux1 = c * Drx1 / Dr1 - vrx1
        ux2 = c * Drx2 / Dr2 - vrx2
        'Compute the electric field components
        Ex1(Index) = q / (4 * pi * epsilon0) * Dr1 / ((Drx1 * ux1) ^ 3) * (ux1 * (c ^ 2 - vrx1 ^ 2))
        Ex2(Index) = q / (4 * pi * epsilon0) * Dr2 / ((Drx2 * ux2) ^ 3) * (ux2 * (c ^ 2 - vrx2 ^ 2))
    Next Index
    'Now compute the electric (interactive) forces on q1 and q2.
    For Index = 0 To Steps - 1
        Fx1(Index) = q * Ex2(Index)
        Fx2(Index) = q * Ex1(Index)
        Fx(Index) = Fx1(Index) + Fx2(Index)
    Next Index
```

```
        'Compute the work per cycle expended to counteract the
interactive
        'forces, at the current separation.
        For Index = 0 To Steps - 1
            vx = omega * Amp * Cos(omega * t(Index))
            WInt(LIndex) = WInt(LIndex) - Fx(Index) * vx * deltat
        Next Index
    Next LIndex
    'Output the works per cycle for plotting purposes.
    Open "c:\winmcad\WorkTwoCharges.prn" For Output As 1
    For Index = 0 To Steps - 1
        Write #1, L(Index) / lambda, WInt(Index)
    Next Index
    Close 1
    MsgBox ("Ready for plotting.")
End Sub
```

Appendix 6_2

```
Private Sub cmdEradL_Click()
'**********
'Compute the energy flux per cycle through a spherical
surface surrounding
'an oscillating pair of charges, separated by a distance L that
varies
'from .5 lambda to 3 lambda.
'**********


'Physical and mathematical constants
   Const c As Double = 299792000#     'Speed of light
   Const epsilon0 As Double = 0.00000000000885
'Permittivity constant
   Const pi As Double = 3.14159265358979
   Const Steps As Long = 100
   Const q As Double = 1
   Const lambda As Double = 0.1    'wavelength (meters)
   Const omega As Double = 2 * pi * c / lambda 'angular
frequency
   Const freq As Double = omega / (2 * pi) 'frequency
   Const tau As Double = 1 / freq  'period
```

```
    Const deltat As Double = tau / Steps    'time interval between epochs
    Const Amp As Double = 0.1 * c / omega   'amplitude (meters)
    Const LMin As Double = 0.5 * lambda 'minimum separation
    Const Lmax As Double = 3 * lambda   'maximum separation
    Const deltaL As Double = (Lmax - LMin) / Steps  'interval between separations
    Const dtheta As Double = pi / Steps 'interval between angles over sphere
    Const Radius As Double = 2 * (Lmax / 2 + Amp)   'spherical radius

'Variables
    Dim theta As Double 'angle between x-axis and point on sphere (in xy-plane)
    Dim Py, Px As Double    'coordinates of point on sphere
    Dim TimeIndex, ThetaIndex, LIndex As Long   'Loop index
    Dim t(Steps) As Double     'Current time
    Dim tr As Double     'Retarded time
    Dim dt As Double     't - tr
    Dim dtmin As Double     'Minimum value for dt
    Dim dtmax As Double     'Maximum value for dt
    Dim xr1, xr2 As Double    'Retarded positions (on x-axis)
    Dim vrx1, vrx2 As Double   'Retarded velocities
```

```
   Dim arx1, arx2 As Double   'Retarded accelerations
   Dim Drx1, Drx2 As Double    'x-components of vectors Dr
   Dim Dry1, Dry2 As Double    'y-components of vectors Dr
   Dim Dr1, Dr2 As Double    'magnitudes of vectors Dr
   Dim ux1, ux2 As Double    'x-components of vectors u
   Dim uy1, uy2 As Double    'y-components of vectors u
   Dim Ex1, Ex2, Ex As Double   'x-components of electric
field vectors
   Dim Ey1, Ey2, Ey As Double   'y-components of electric
field vectors
   Dim Bz1, Bz2, Bz As Double   'z-components of magnetic
field vectors
   Dim Sx As Double    'x-component of Poynting vector at
point on sphere
   Dim Sy As Double    'y-component of Poynting vector
   Dim unitx, unity, dA As Double  'unit vectors and spherical
area increment
   Dim EFluxRate(Steps), EFlux(Steps) As Double 'Enery flux
at different angles
   Dim Snormal As Double   'normal component of Poynting
vector
   Dim L(Steps) As Double 'separation of q1 and q2
   Dim EradL(Steps) As Double  'energy flux per cycle at
separation L
```

```
    Dim RadWork As Double   'Work per cycle to counteract
rad react forces
    Dim ERadInt(Steps) As Double  'Total work minus
RadWork

    'Zero out each value of Erad(L)
    For LIndex = 0 To Steps - 1
        EradL(LIndex) = 0
    Next LIndex
    'Compute Erad(L) for each separation from the Minimum to
the Maximum L.
    For LIndex = 0 To Steps - 1
        Debug.Print LIndex
        'Set the separation for this loop.
        L(LIndex) = LMin + LIndex * deltaL
        'Zero out the energy fluxes for each epoch, present
separation.
        For TimeIndex = 0 To Steps - 1
            EFlux(TimeIndex) = 0
        Next TimeIndex
        'Compute the energy flux through the spherical surface,
for each time
        'interval.
        For TimeIndex = 0 To Steps - 1  'time epochs
            t(TimeIndex) = TimeIndex * deltat
```

```
'Zero out the energy flux rates for each theta interval.
For ThetaIndex = 0 To Steps - 1
   EFluxRate(ThetaIndex) = 0
Next ThetaIndex
'Compute the energy flux rate through each theta interval.
For ThetaIndex = 0 To Steps - 1
   theta = ThetaIndex * dtheta + dtheta / 2
   unitx = Cos(theta)
   unity = Sin(theta)
   Px = Radius * Cos(theta)
   Py = Radius * Sin(theta)
   dA = 2 * pi * Py * Radius * dtheta

   'First do q1.
   dtmin = 0
   dtmax = 2 * Radius / c
   Do
      dt = (dtmax + dtmin) / 2
      tr = t(TimeIndex) - dt
      xr1 = -L(LIndex) / 2 + Amp * Sin(omega * tr)
      Drx1 = Px - xr1
      Dry1 = Py
      Dr1 = Sqr(Drx1 ^ 2 + Dry1 ^ 2)
      If Abs(c * dt - Dr1) < 2 ^ (-30) Then Exit Do
```

```
    If c * dt - Dr1 > 0 Then
        dtmax = dt
    Else
        dtmin = dt
    End If
Loop
vrx1 = omega * Amp * Cos(omega * tr)
arx1 = -(omega ^ 2) * Amp * Sin(omega * tr)

'Then do q2.
dtmin = 0
dtmax = 2 * Radius / c
Do
    dt = (dtmax + dtmin) / 2
    tr = t(TimeIndex) - dt
    xr2 = L(LIndex) / 2 + Amp * Sin(omega * tr)
    Drx2 = Px - xr2
    Dry2 = Py
    Dr2 = Sqr(Drx2 ^ 2 + Dry2 ^ 2)
    If Abs(c * dt - Dr2) < 2 ^ (-30) Then Exit Do
    If c * dt - Dr2 > 0 Then
        dtmax = dt
    Else
        dtmin = dt
    End If
```

```
Loop
vrx2 = omega * Amp * Cos(omega * tr)
arx2 = -(omega ^ 2) * Amp * Sin(omega * tr)

'Compute the components of vectors u.
ux1 = c * Drx1 / Dr1 - vrx1
uy1 = c * Dry1 / Dr1
ux2 = c * Drx2 / Dr2 - vrx2
uy2 = c * Dry2 / Dr2
'Compute the electric and magnetic field components
Ex1 = q / (4 * pi * epsilon0) * Dr1 / (Drx1 * ux1 + Dry1 * uy1) ^ 3 * (ux1 * (c ^ 2 - vrx1 ^ 2) + Dry1 * (-uy1 * arx1))
Ey1 = q / (4 * pi * epsilon0) * Dr1 / (Drx1 * ux1 + Dry1 * uy1) ^ 3 * (uy1 * (c ^ 2 - vrx1 ^ 2) - Drx1 * (-uy1 * arx1))
Bz1 = 1 / (c * Dr1) * (Drx1 * Ey1 - Dry1 * Ex1)
Ex2 = q / (4 * pi * epsilon0) * Dr2 / (Drx2 * ux2 + Dry2 * uy2) ^ 3 * (ux2 * (c ^ 2 - vrx2 ^ 2) + Dry2 * (-uy2 * arx2))
Ey2 = q / (4 * pi * epsilon0) * Dr2 / (Drx2 * ux2 + Dry2 * uy2) ^ 3 * (uy2 * (c ^ 2 - vrx2 ^ 2) - Drx2 * (-uy2 * arx2))
Bz2 = 1 / (c * Dr2) * (Drx2 * Ey2 - Dry2 * Ex2)
'Sum the field components for the 2 charges.
Ex = Ex1 + Ex2
Ey = Ey1 + Ey2
Bz = Bz1 + Bz2
```

```
                'Compute the Poynting vector components, using the net E and B
                'field components.
                Sx = epsilon0 * c ^ 2 * (Ey * Bz)
                Sy = epsilon0 * c ^ 2 * (-Ex * Bz)
                'Find the Poynting vector component that is normal to the
                'spherical surface.
                Snormal = Sx * unitx + Sy * unity
                EFluxRate(ThetaIndex) = EFluxRate(ThetaIndex) + Snormal * dA
            'Repeat for the next angle theta.
            Next ThetaIndex
            'Now update the running sum of energy fluxes.
            For ThetaIndex = 0 To Steps - 1
                EFlux(TimeIndex) = EFlux(TimeIndex) + EFluxRate(ThetaIndex) * deltat
            Next ThetaIndex
        Next TimeIndex
        'Once the energy fluxes for each time interval has been computed, add
        'the energy fluxes to Erad for the current separation.
        For TimeIndex = 0 To Steps - 1
            EradL(LIndex) = EradL(LIndex) + EFlux(TimeIndex)
        Next TimeIndex
```

```
    Next LIndex
    'Having computed Erad(L), optionally subtract out the
(constant) work expended
    'to counteract the radiation reaction force.
    RadWork = q ^ 2 * omega ^ 3 * Amp ^ 2 / (3 * epsilon0 * c ^
3)
    For LIndex = 0 To Steps - 1
        ERadInt(LIndex) = EradL(LIndex) - RadWork
    Next LIndex
    'After computing all the energy fluxes over the entire range
of L,
    'output the values for plotting purposes.
    Open "c:\winmcad\ERad.prn" For Output As 1
    For LIndex = 0 To Steps - 1
        Write #1, L(LIndex) / lambda, EradL(LIndex),
ERadInt(LIndex)
    Next LIndex
    Close 1
    MsgBox ("Ready for plotting.")

End Sub
```