AC 2007-2177: SIMULATION AND ANIMATION OF ENGINEERING SYSTEMS: NO SPECIALIZED SOFTWARE OR PROGRAMMING REQUIRED

William Ziegler, State University of New York-Binghamton

William Ziegler is an Associate Professor in the Thomas J. Watson School of Engineering and Applied Science, and Faculty Master of Newing College, at the State University of New York - Binghamton University.

Simulation and Animation of Engineering Systems:
No Specialized Software or Programming Required

Abstract:

Many topics in engineering curricula rely on visual components to help convey concepts that are difficult to describe in purely text-based form. Similarly, raw data plotted in a colorful three-dimensional graph brings life to otherwise static numerical information. The extraordinary value of providing a visual component to teaching and learning is well documented. Adding animation to visual components only serves to enhance the learning experience even further. The use of animation allows students to view dynamic processes in various topics throughout all engineering disciplines in a manner that effectively engages the student in the learning process. However, the task of creating simulations and animations is frequently a daunting, time-consuming task where regardless of the usefulness there is simply not enough time in a day to create such a worthwhile experience for students. Therefore, providing instructors with the capability to create, with ease, animations of complex engineering systems and providing students with the opportunity to view these “living” systems is invaluable in the quest to enhance teaching and learning and to retain student interest in the subject matter.

This paper will provide a discussion centered on two intertwined themes. The first theme provides details regarding useful techniques for creating animation of dynamic processes applicable across all engineering disciplines, without the use of programming or specialized software. The second theme is the development of an example of animating the architecture of a computer system. This analytical example evolves from a course in digital systems, and actively demonstrates the physical interrelationships of individual components of a computer system as a program executes in real-time, clock-pulse by clock-pulse. This simulation/animation provides an effective opportunity for students to visualize the movement of data throughout the architecture of the system as a program executes in either a continuous or stop-and-go fashion at a speed determined by the analytical capabilities of the student user of the system. Even though the example is from the topic of digital systems, the explanation of the simulation/animation techniques is presented in a manner that is applicable across all engineering disciplines.

Introduction:

Engineering students are typically hands-on visually oriented learners. The extraordinary value of providing a visual component to teaching and learning is well documented. Many topics in engineering curricula rely on visual components to help convey concepts that are difficult to describe in purely text-based form. Adding animation to visual components only serves to enhance the learning experience even further. Bringing animation to raw data plotted in a colorful three-dimensional graph brings life to otherwise static numerical information.

Simulation and animation can be utilized by an instructor to illustrate concepts in a classroom setting or by students to solve problems in a laboratory setting.
The use of animation provides a method to model and observe dynamic processes in all engineering disciplines in a manner that effectively engages the observer.

Tools for Simulation and Animation:

MATLAB has been utilized extensively and is one of the most common animation and simulation tools. For example, Jacquot et al. used MATLAB to animate electrical transmission lines[1], for clarification of partial differential equation solutions using 2-dimensional (2-D) and 3-dimensional (3-D) graphics[2], and for providing visualization of partial differential equations[3].

In addition to MATLAB, there are many other useful tools. Watkins et al.[4] used computer animation to create a visualization tool for dynamic systems simulations, and Valocchi et al.[5] created a Web-based interactive simulator to study groundwater pollutant fate and transport.

However, the task of creating simulations and animations is frequently a daunting, time-consuming task where regardless of the usefulness there is simply not enough time in a day to create such a worthwhile experience for students. For example, a simple MATLAB animation may consist of lengthy computer programs consisting of several hundred lines of programming to produce even the most basic animations as shown in Figure 1.[1,6]

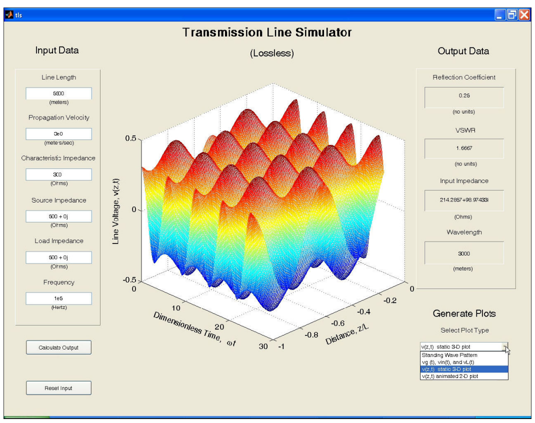


Figure 1. Transmission Line Simulator requiring non-trivial programming skills[1,6]

Therefore, providing instructors with the capability to create, with ease, animations of complex engineering systems and providing students with the opportunity to view these "living" systems is invaluable in the quest to enhance teaching and learning and to retain student interest in the subject matter.

However, this is not suggesting that all simulations or animations can be accomplished with ease, just that there are relatively easy methods to animate and simulate a large variety of applications.

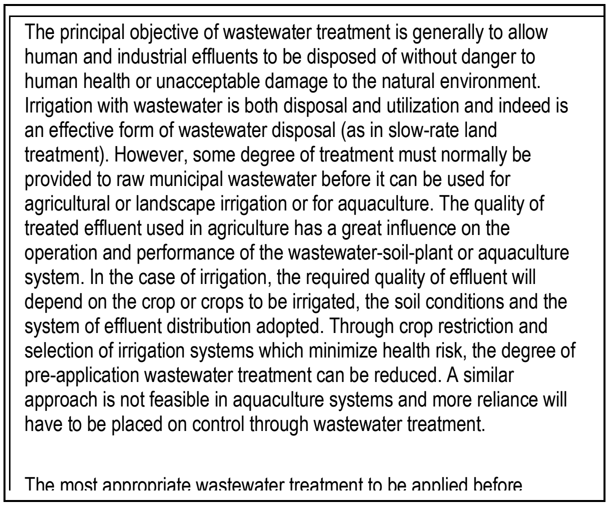
The tool highlighted in this paper is Power Point. With relative ease, and without having to perform any programming, Power Point can be used to bring life to engineering systems.

Options for Conveying Technical Content:

The choices available for providing technical content can be categorized into six areas:

1. verbal only
2. text only
3. still black and white visuals in 2-D
4. still color visuals in 3-D
5. animated graphics
6. live or recorded video.

For the process shown in Figure 4, imagine attempting to understand the engineering processes displayed using only text (as sampled in Figure 2). A text-only description would be lengthy, cumbersome, and confusing. The black and white 2-D still visual as sampled in Figure 3 helps clarify the engineering processes. Figure 4 is a sample of a still 3-D color visual and provides considerable assistance in conveying the information. Animating Figure 4 to allow visualization of the entire engineering process would certainly engage the observer in the processes being performed. An example of extreme animation is shown in Figure 5, where the diagram comes to life showing the movement of the gears, levers, and full mechanics.


The principal objective of wastewater treatment is generally to allow human and industrial effluents to be disposed of without danger to human health or unacceptable damage to the natural environment. Irrigation with wastewater is both disposal and utilization and indeed is an effective form of wastewater disposal (as in slow-rate land treatment). However, some degree of treatment must normally be provided to raw municipal wastewater before it can be used for agricultural or landscape irrigation or for aquaculture. The quality of treated effluent used in agriculture has a great influence on the operation and performance of the wastewater-soil-plant or aquaculture system. In the case of irrigation, the required quality of effluent will depend on the crop or crops to be irrigated, the soil conditions and the system of effluent distribution adopted. Through crop restriction and selection of irrigation systems which minimize health risk, the degree of pre-application wastewater treatment can be reduced. A similar approach is not feasible in aquaculture systems and more reliance will have to be placed on control through wastewater treatment.

The most appropriate wastewater treatment to be applied before

Figure 2: Description in pure text[7]

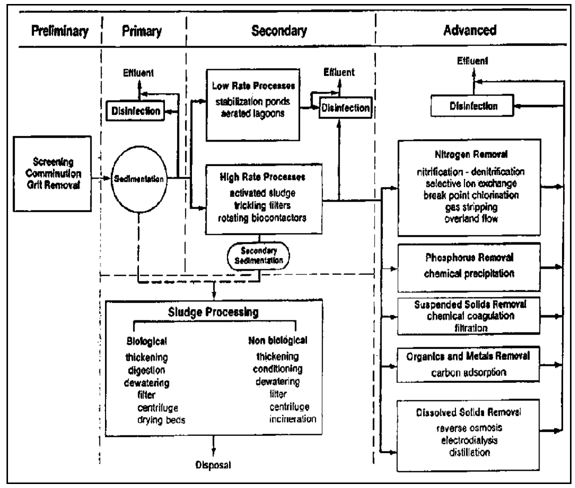


Figure 3: Still visual in 2-D[7]

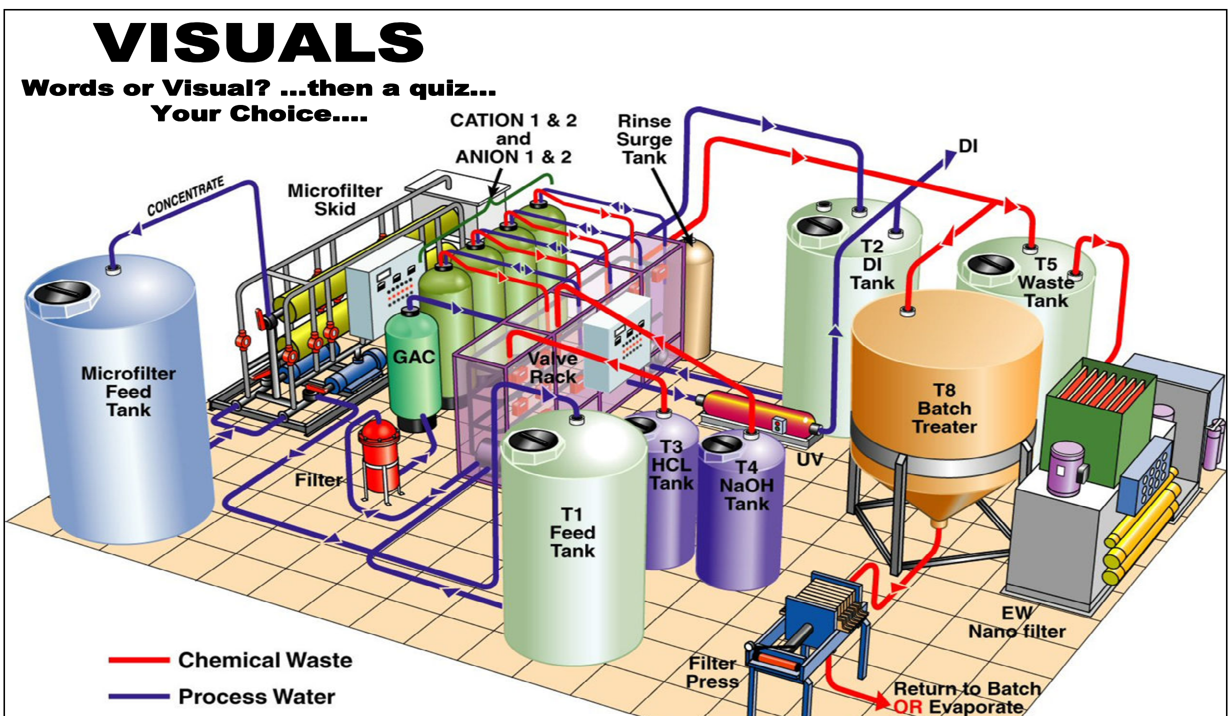


Figure 4: Color enhanced 3-D visual[7]

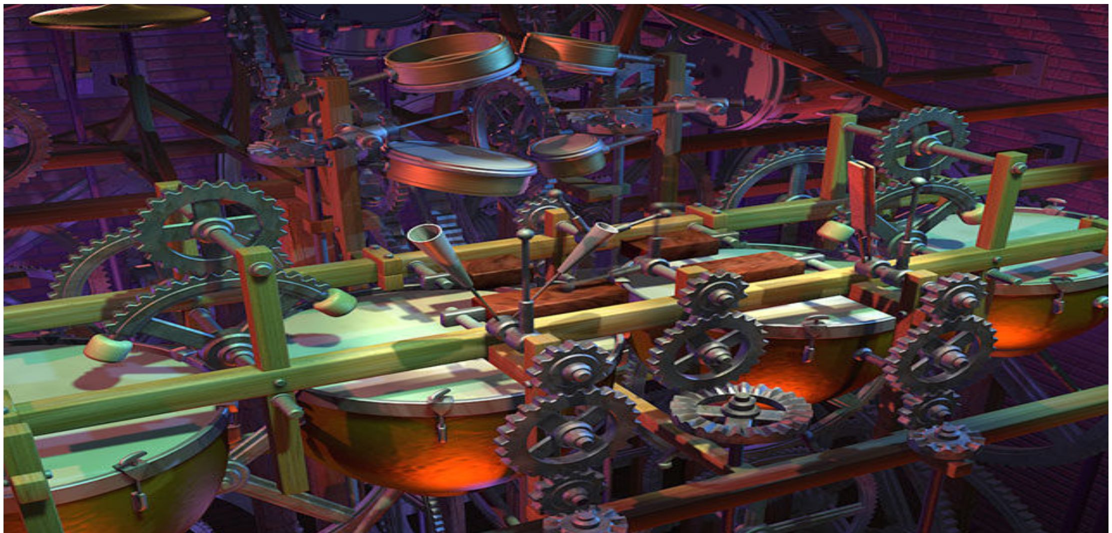


Figure 5: Extreme Animation of a Mechanical System[8]

High quality computer generated animation (preferably with audio) and movie quality audio/video certainly enhances any form of information. Figure 6 depicts still clips from a computer-generated video of an animation of the assembly of the Hirobo Helicopter. Highly specialized programming is required to create such a video, and this particular video was made using a combination of Solidworks, Pro/E, and 3D Studio Max.[9,10]

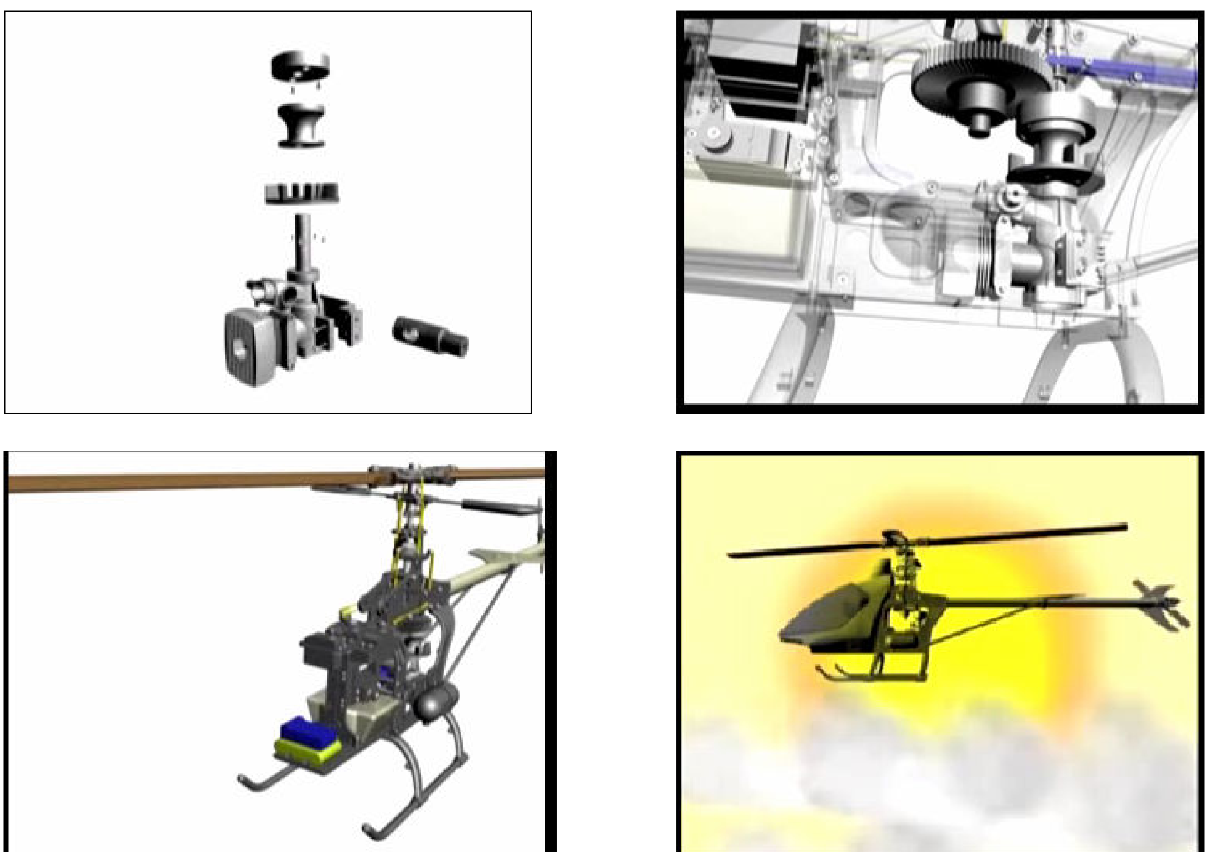
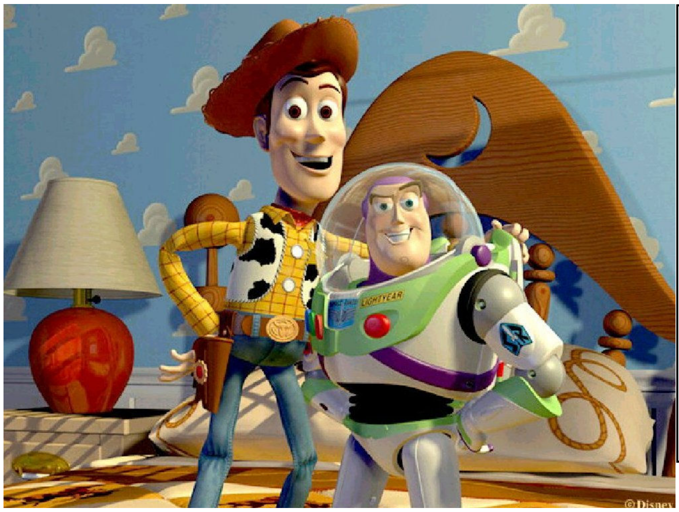


Figure 6: Sampling of scenes from the animated assembly of the Hirobo Helicopter[10]

The ultimate animation is the type seen in major motion pictures such as Pixar's Toy Story [11] as shown in Figure 7. Unfortunately, the cost of this sort of animation is reportedly in the range of $20 million per hour. To animate a complete 15 week course that meets three hours per week would therefore approach $900 million and is certainly prohibitive.



$20 million per hour

$20,000,000,000
x 3 hrs/wk
x 15 wks

Only $900 million per course.

Figure 7: Pixar's animated motion picture, Toy Story[11]

Animation - Back to Reality:

As previously stated, the task of creating simulations and animations is frequently a daunting, time-consuming task where regardless of the usefulness there is simply not enough time in a day to create such a worthwhile experience for students. Hiring outside firms to create the animations is typically prohibitive and certainly paying $20 million per hour[11] is a bit steep for most department budgets. However, there are unlimited possibilities for animating engineering systems using simple techniques available in Power Point.

Power Point Animation Mechanics:

Power Point may be used to easily animate any number of graphic scenarios as shown in each of the following Figures.

As shown in Figure 8, only two Power Point commands are required to establish a very useful base of animations:

1. *INSERT* (line, text box, shape, diagram, clip art, arrow, or picture)
2. *CUSTOM ANIMATION* (motion path, entry, emphasis, or exit)

The *INSERT* option is used to place objects on the screen, and then *CUSTOM ANIMATION* is used to animate the objects.

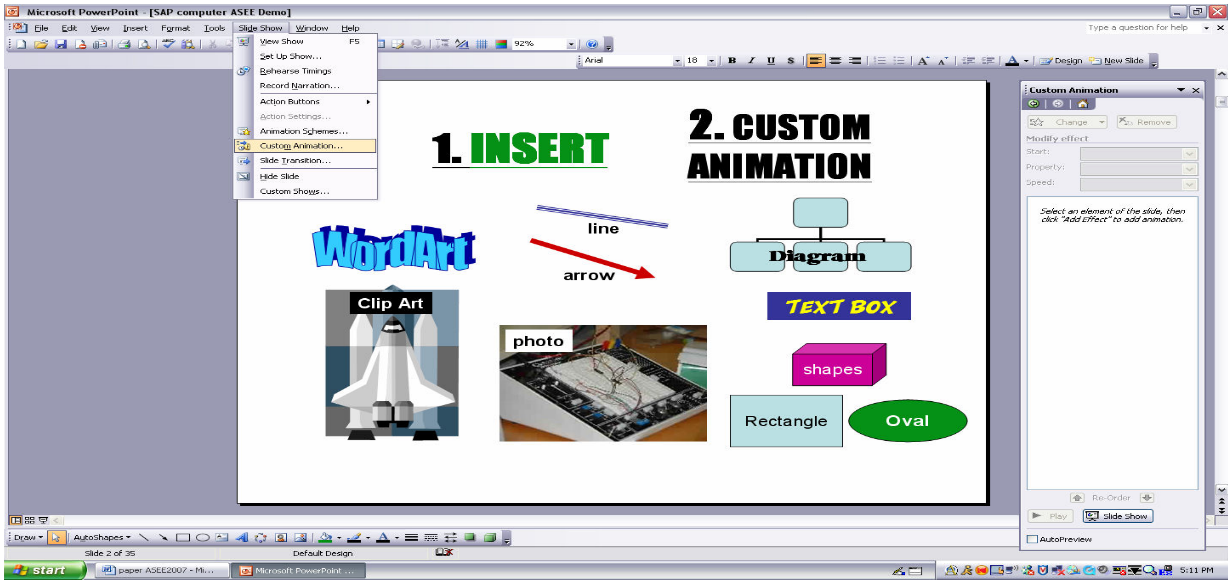


Figure 8: Only two commands are necessary – 1. *Insert* and 2. *Custom Animation*

Dynamic Fluids, Air, Data:

Creating an animation to simulate the movement of fluids, data, air, etc can be accomplished in only two steps as shown in Figure 9.

1. Insert graphic (clip art, drawing, etc.) and arrows
2. From the *Slide Show* menu option, use *Custom Animation* (choose *Motion Paths*) to simulate movement. Each arrow will move along its designated motion path (as shown with dotted lines in the Figure 9).

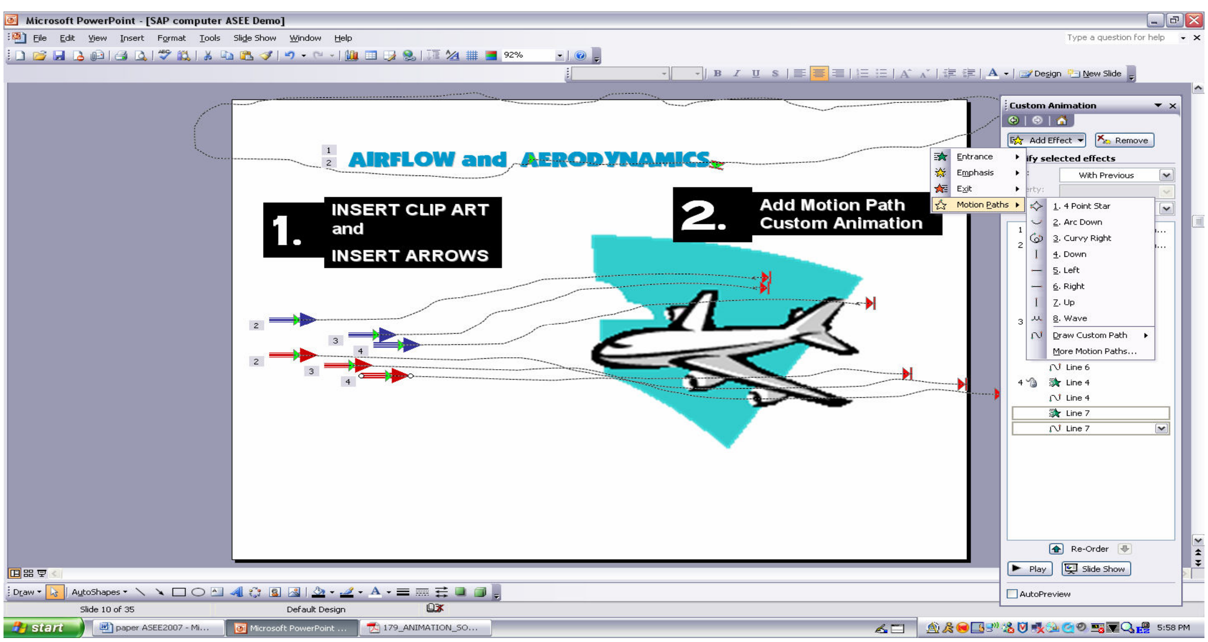


Figure 9: Airflow using animated motion paths

Animated Charts and Graphs:

Adding dynamic properties to static graphs and charts can be very captivating to the observer and can be accomplished in only two steps as shown in Figure 10.

1. Insert each bar of the graph as a rectangle.
2. Click on *Slide Show* and then *Custom Animation* (use *Entrance and Dissolve-in*). Each bar will appear beginning as tiny particles and evolving into the full bar.

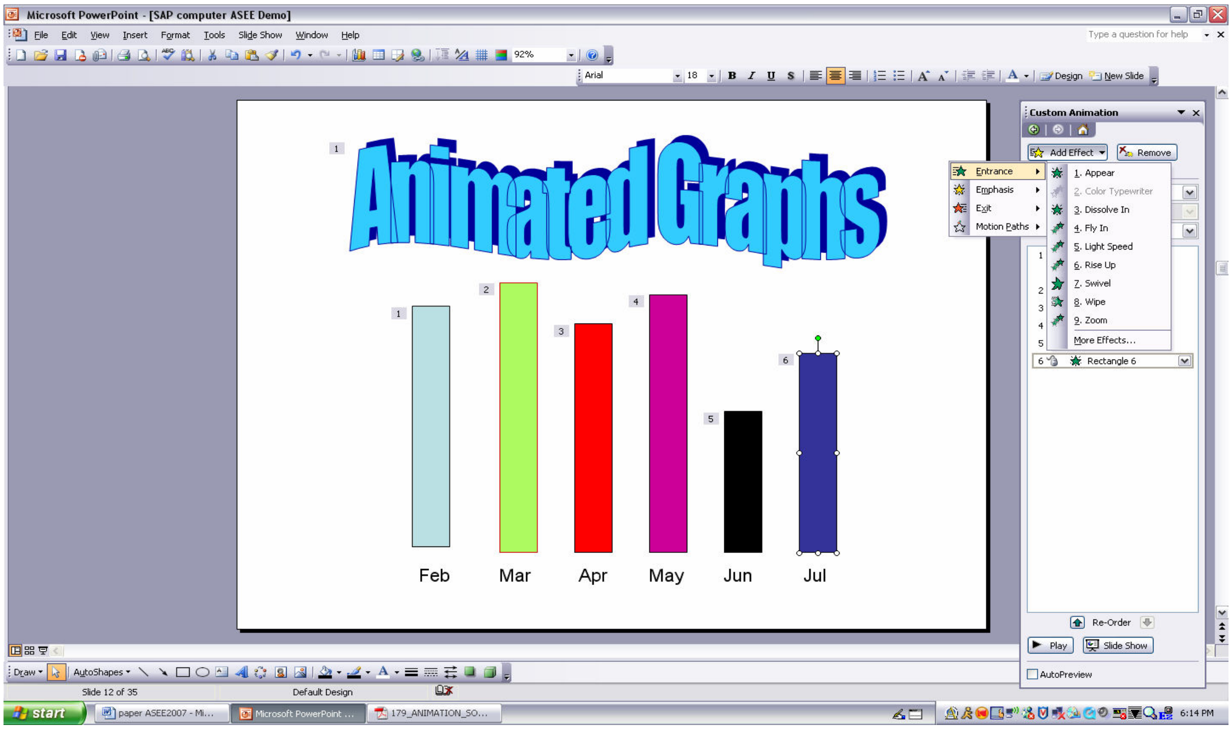


Figure 10: Animating charts and graphs

Transmission of Data, Signals, and Objects:

Animating the movement of data through a system, or the transmission of electricity, etc. can be accomplished in just two basic steps as shown in Figure 11.

1. Insert starting point graphic and ending point (clip art, drawing, etc)
2. Click on *Slide Show* and then choose *Custom Animation* (use *Motion Paths*) to simulate movement. The designated object (an email represented as a letter as shown in the figure) will traverse along the designated motion path (as shown with dotted line) from start to end.

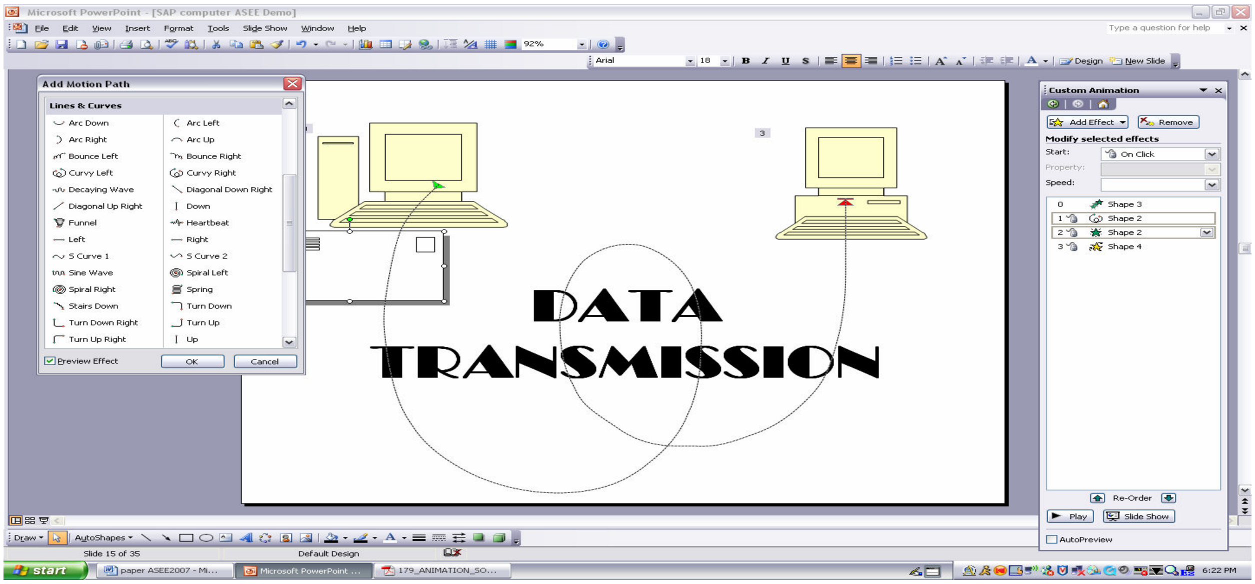


Figure 11: Transmitting Information (or electricity, etc)

Traversing a Path:

Traversing a path through a system can be accomplished in only three steps as shown in Figure12.

1. Insert a diagram of the system as the background (a map of the US in the figure).
2. Insert objects (clip-art of a racecar in the figure) to represent stops on the path.
3. Set up *Custom Animation* (use *Motion Paths*) to simulate movement of the first object to the second object, the second object to the third object, etc. Each object will traverse along the designated motion path (as shown with dotted lines) from start to end.

Figure 12: Traversing a designated path.

Engineering System Connectivity and Assembly:

Connectivity or assembly of elements in an engineering system can be accomplished in only three steps as shown in Figure 13.

1. Insert each object required for the assembly (computer peripherals in the figure).
2. Insert a connector between elements that are connected during assembly (a shape was used in the figure). Stretch the connector to show connections between objects.
3. Click on *Slide Show* and then choose *Custom Animation* to simulate adding components to the base (processor) diagram or object.
   a. The screen begins blank and then the processor appears. *(Custom Animation – Entrance – Dissolve in)*
   b. The connector to the monitor appears. *(Custom Animation – Entrance – Wipe)*
   c. The peripheral appears at the end of the connector. *(Custom Animation – Entrance – Dissolve in)*

   Steps *b and c* are then repeated for each connector and peripheral. Note there are nine objects in the diagram, and nine associated items in the Custom Animation process window shown on the right side of the screen in Figure 13.

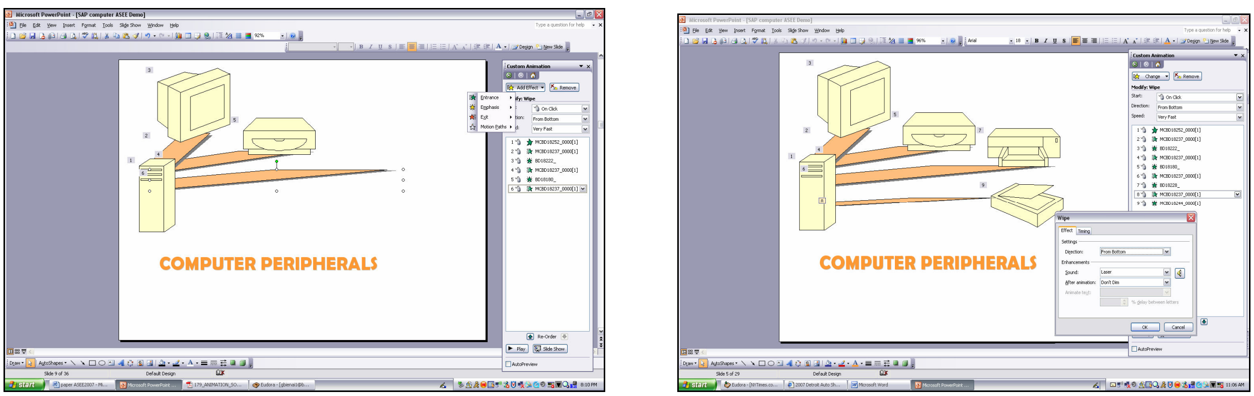


Figure 13: Two views in time of a slide animating the assembly of an engineering system

3-D Object Rotation and System Assembly:

Adding a third dimension to a diagram considerably enhances the reality of the object as shown in Figure 14. The use of the transparency feature, which creates a see-through effect of some of the objects, is also very useful (as shown in Figure 15).

1. Insert each object as a 2-dimensional plane.
2. Add depth (the third dimension) by clicking on the 3-D icon near the bottom center of the screen
3. Place the objects on the screen using *Custom Animation – Entrance – Dissolve In*
4. Set up the transparency effect by clicking on the object. A Window as shown in Figure 15 will appear and then the degree of transparency and color can be selected.
5. Rotate the objects using *Custom Animation – Emphasis - Spin (*and select the number of times to *repeat* the spin)

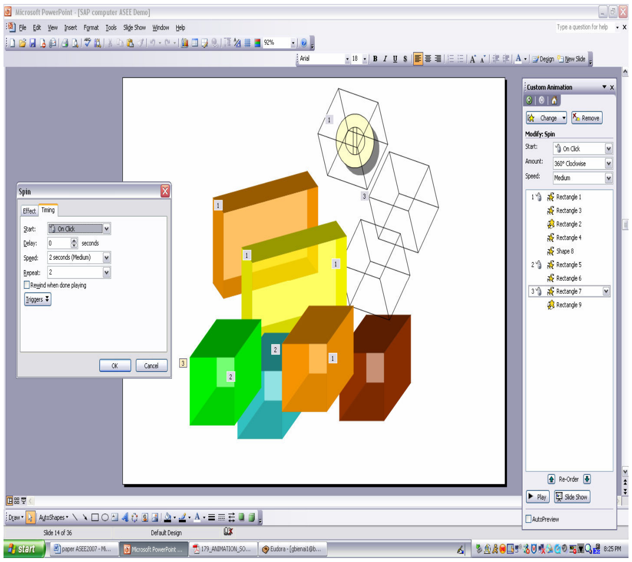


Figure 14: Rotation/assembly of 3-D objects.

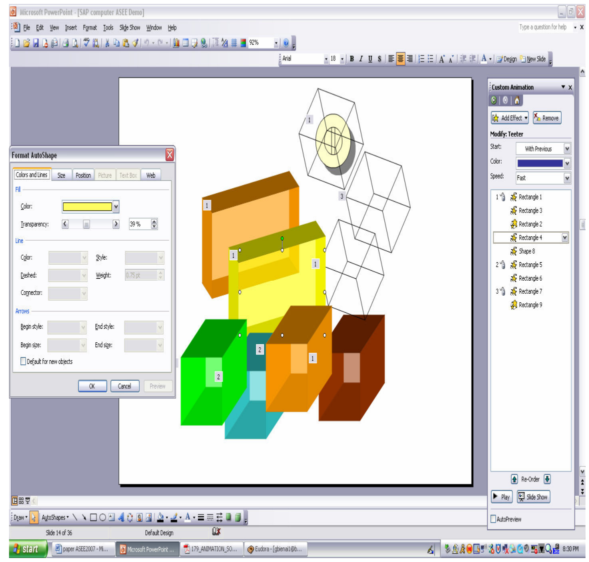


Figure 15: Selecting degree of transparency

Animation of a Complete Engineering System – Computer Engineering

The following is the development of an example of animating the architecture of a computer system. This analytical example actively demonstrates the physical interrelationships of individual components of a computer system (or any engineering system) as a program executes in real-time, clock-pulse by clock-pulse. This simulation/animation provides an effective opportunity for students to visualize the movement of data throughout the architecture of the system as a program executes in either a continuous or stop-and-go fashion at a speed determined by the analytical capabilities of the student user of the system. Even though the example is from the topic of digital systems, the explanation of the simulation/animation techniques is presented in a manner that is applicable across all engineering disciplines.

The example being developed (as shown in its final form in Figures 22, 23, and 24) is a computer engineering application and represents the computer architecture and the implementation of machine code instructions into supporting hardware, including

- RAM
- Control Unit
- Data Bus
- Program Counter
- ALU with two operands (registers)
- Output Register
- Instruction Register with Instruction Decoder

To begin the task of building the system, open a new Power Point presentation. Insert a text box, and then add color and text (as shown in Figure 16).

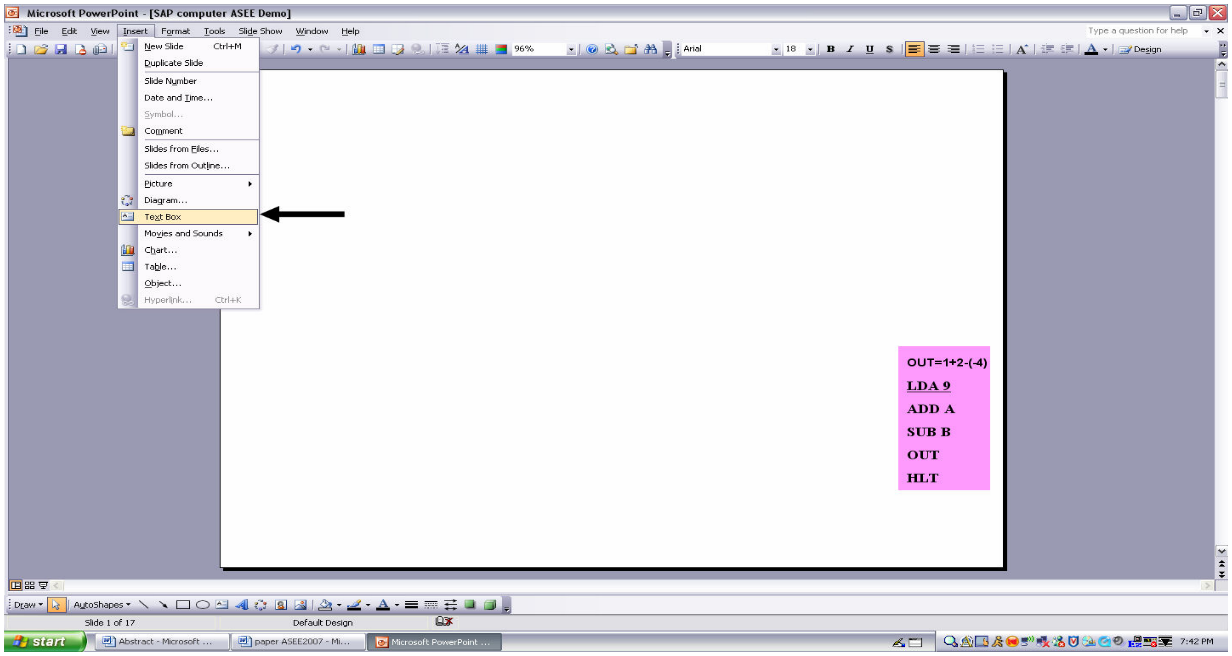


Figure 16: Slide 1 – begin with the first text box (an Assembly Language program)

Copy each slide forward to begin the next subsequent slide. Continue adding text boxes, one (or more) per slide as shown in Figure 17. Insert arrows of various sizes and colors to simulate directionality and data movement as shown in Figures 18 and 19.

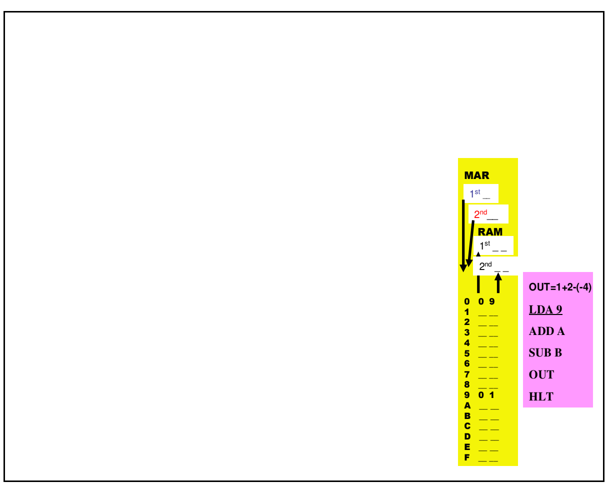


Figure 17: Slide 2 – Add the second text box (containing the box to represent the RAM and Memory Address Register (MAR)).

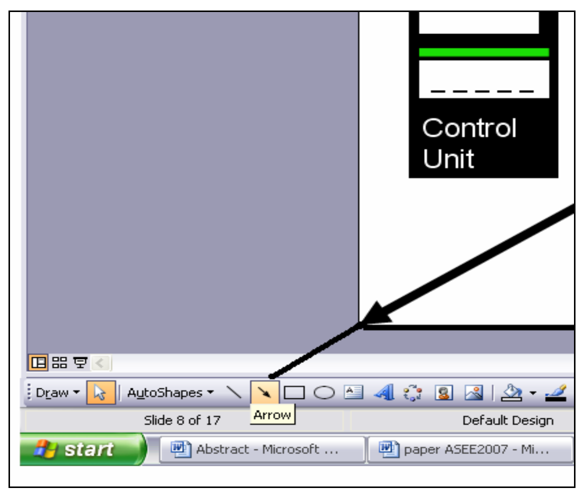


Figure 18: Insert arrows to represent data transmission throughout the system.

By Slide 8, the RAM, MAR, Control Unit, Data Bus, Instruction Register,Decoder, Program Counter, and associated data paths have been added as shown in Figure 19. By slide 11, all hardware components and associated data paths have been added as shown in Figure 20.

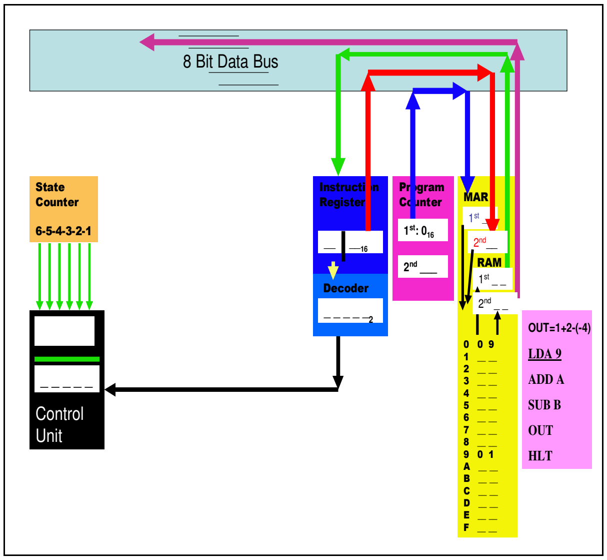


Figure 19: Slide 8

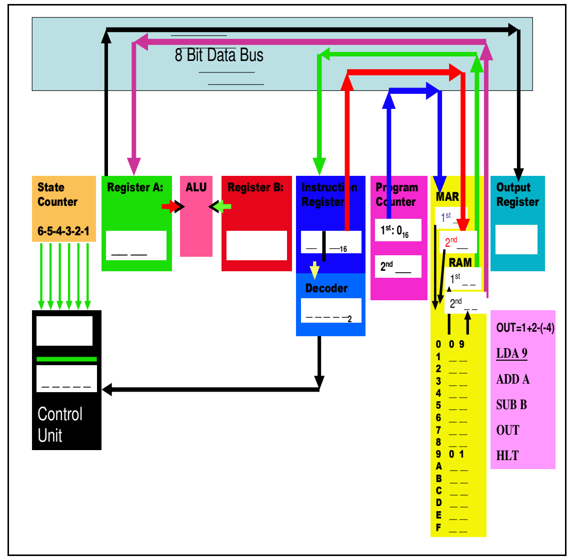


Figure 20: Slide 11 – complete diagram

Using *Custom Animation*, various elements and data paths are animated simply by clicking on the object and selecting from the animation options. As shown in Figures 21 and 22, each animation is numbered in a small box close to the object being animated and represents the order in which the animation will be executed. In Figure 21, note the numbers 1, 2, and 3 in boxes on the left side of the screen correspond to the selected animation menu items 1, 2, and 3 near the center of the screen.

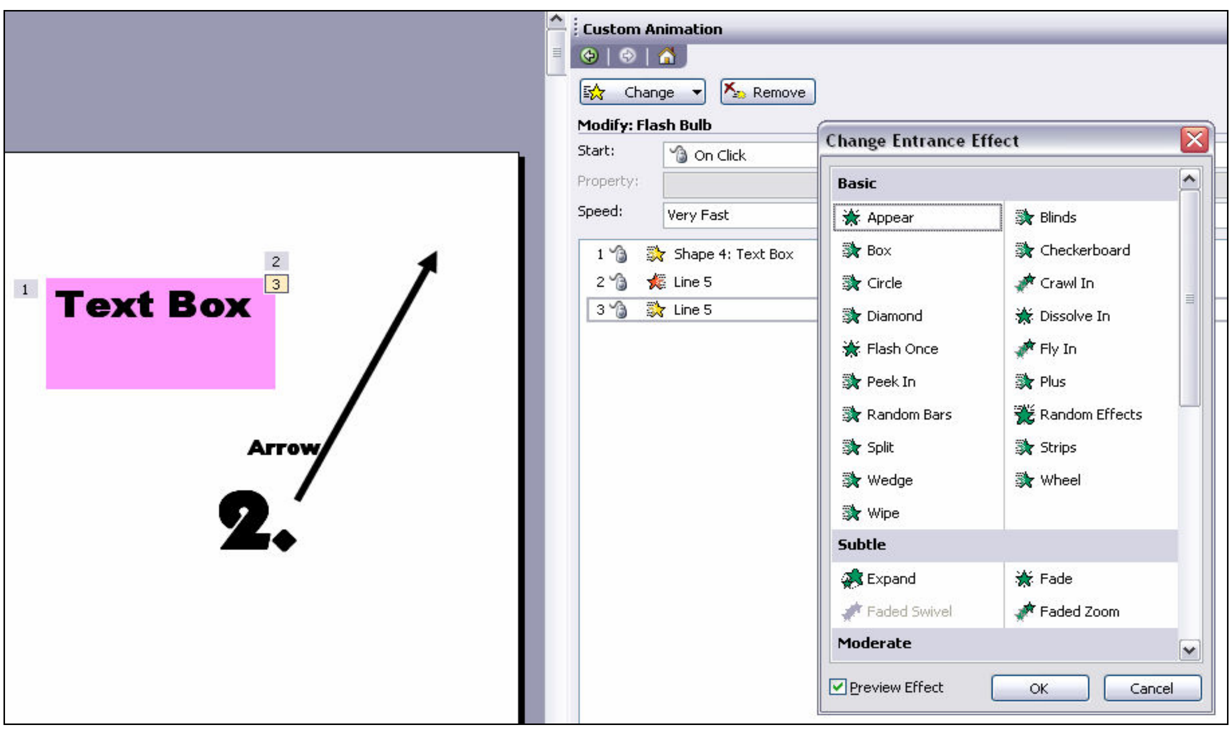


Figure 21: *Custom Animation* Pull Down Menu.

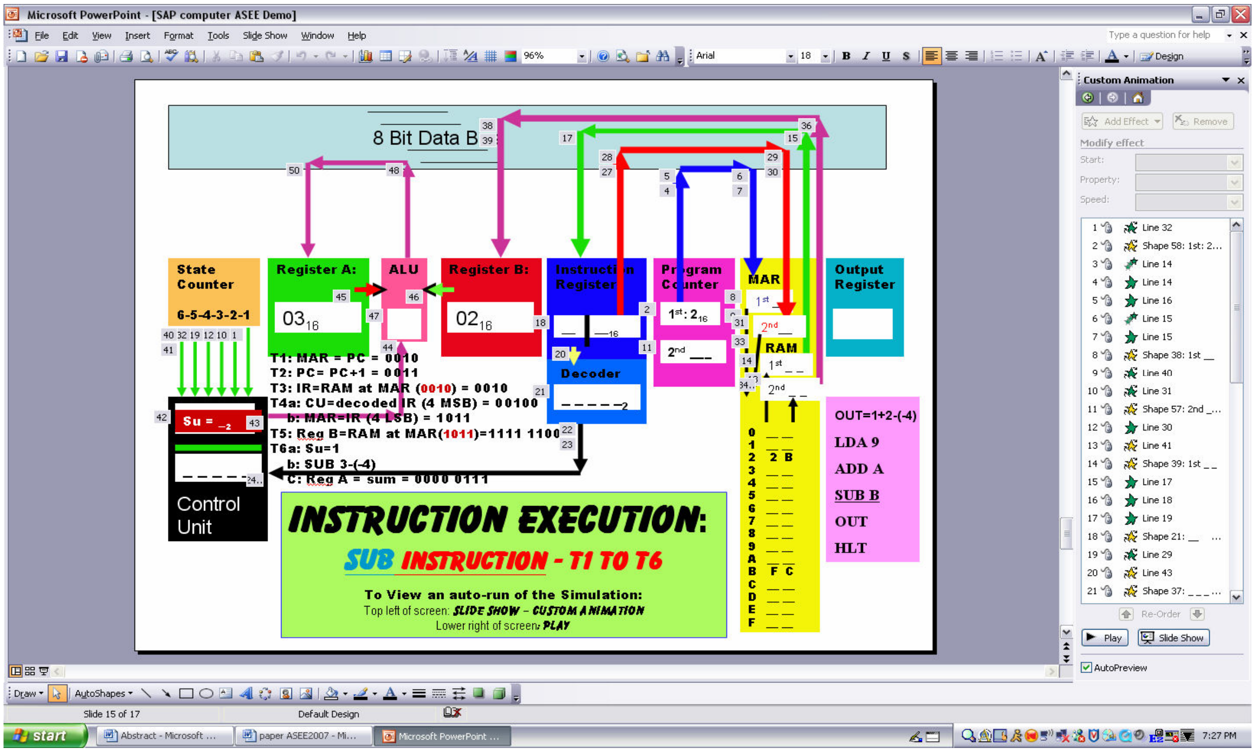


Figure 22: Full diagram, complete with animation.

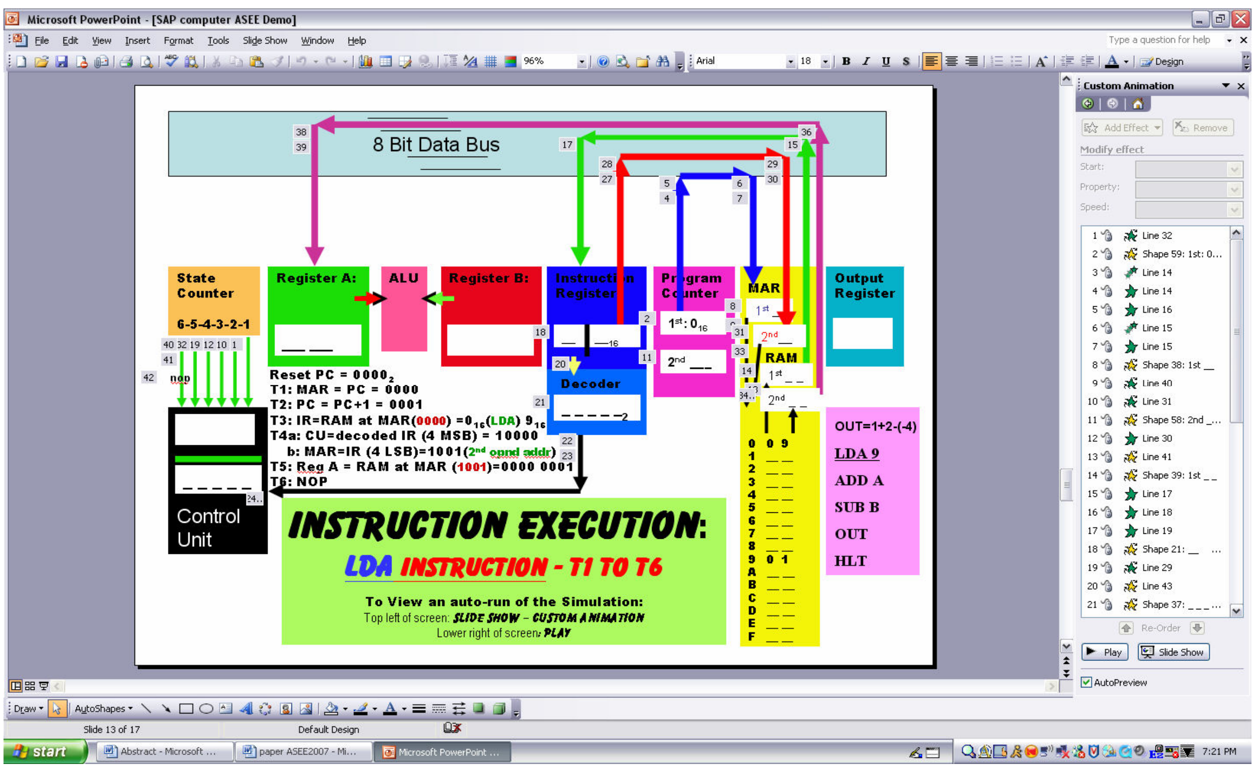


Figure 23: Data Paths for the LDA (Load Accumulator) instruction. Note the small numbers in boxes represent the order in which the animation will be executed.

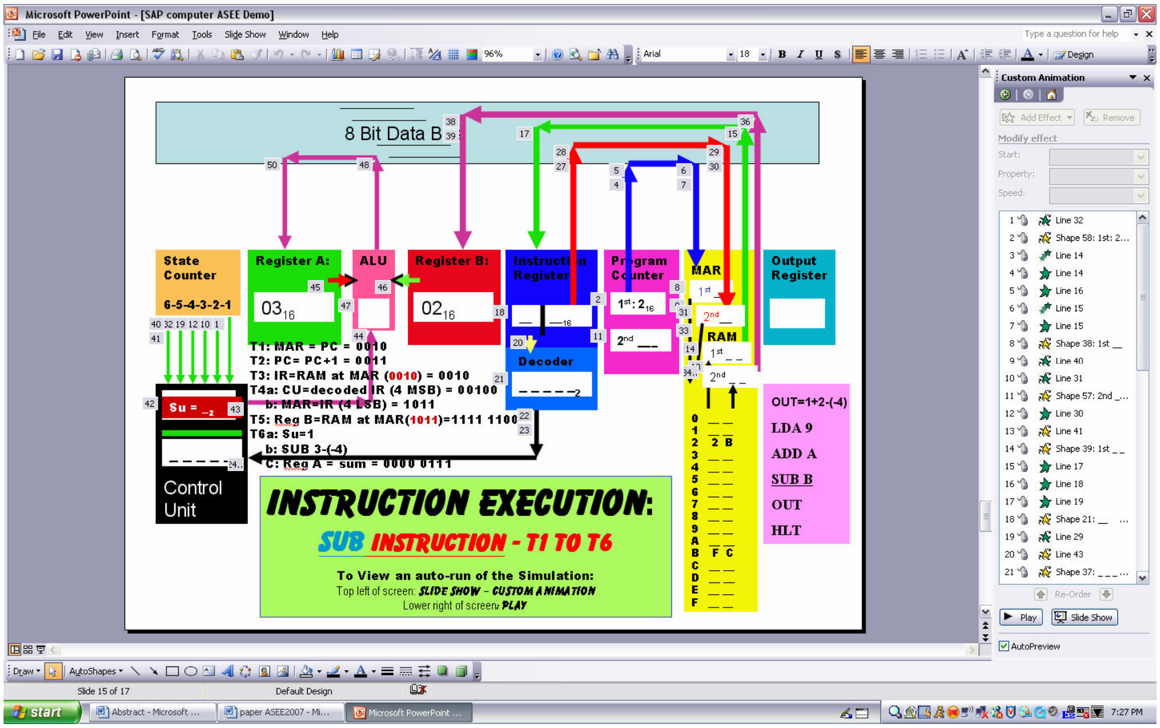


Figure 24: Each assembly language instruction is completely executed on one slide (one instruction per slide). Note that this slide shows the data paths and animation setup for the SUB instruction.

Conclusion:

The extraordinary value of providing a visual component to teaching and learning is well documented. The use of animation to bring life to otherwise lifeless information is a powerful tool and greatly enhances the student's ability to visualize engineering systems.

Power Point is a very common tool that can be used to easily simulate and animate these systems. Using just two basic commands, *insert* and *custom animation*, engineering systems can be brought to life. In just two or three steps, animations can be constructed to demonstrate:

- Dynamic fluids, air, data, etc.
- Animated charts and graphs
- Transmission of signals, data, or objects
- Traversing a path
- Engineering system construction and connectivity
- 3-D object rotation and system assembly

Adding animation to visual components only serves to enhance the learning experience. The use of animation allows students to view dynamic processes in various topics throughout all engineering disciplines in a manner that effectively engages the student in the learning process.

Bibliography:

1. R. Jacquot, C. Wright, R. Kubichek, Animation Software for the Teaching of Electrical Transmission Lines; *Proceedings of the 2006 ASEE Annual Conference*, Chicago, IL, 2006

2. R. Jacquot, C. Wright, T. Edgar and R. Kubichek, Clarification of Partial Differential Solutions `Using 2-D and 3-D Graphics and Animation, *Proceedings of the 2005 ASEE Annual Conference and Exposition*, Portland, OR, June 12-15, 2005.

3. R.G. Jacquot, C. Wright, T. Edgar and R. Kubichek, Visualization of Partial Differential Equation Solutions, *Computing in Science and Engineering*, vol. 8, no. 1, January/February 2006, pp.73-77

4. J. Watkins, G. Piper, K. Wedeward and E.E. Mitchell, Computer Animation: A Visualization Tool for Dynamic Systems Simulations, *Proceedings of the 1997 ASEE Annual Conference*, June 15-18, 1997, Milwaukee, WI

5. A. J. Valocchi and C.J. Werth, Web-Based Interactive Simulation of Groundwater Pollutant Fate and Transport, *Computer Applications in Engineering Education*, vol. 12, no. 2, 2004, pp.75-83.

6. R. Jacquot, University of Wyoming, College of Engineering, www.eng.uwyo.edu/classes/matlabanimate

7. www.fao.org/docrep/T0551E/t0551e05.htm

8. Animusic, Austin, Texas, http://www.animusic.com/company-contact-info.html

9. http://video.google.com/videosearch?q=solidworks+engineering+animation&hl=en

10. http://video.google.com/videoplay?docid=-2774823249721228235&q=solidworks+engineering+animation&hl=en

11. Pixar, http://www.pixar.com/featurefilms/ts/theater/teaser_480.html