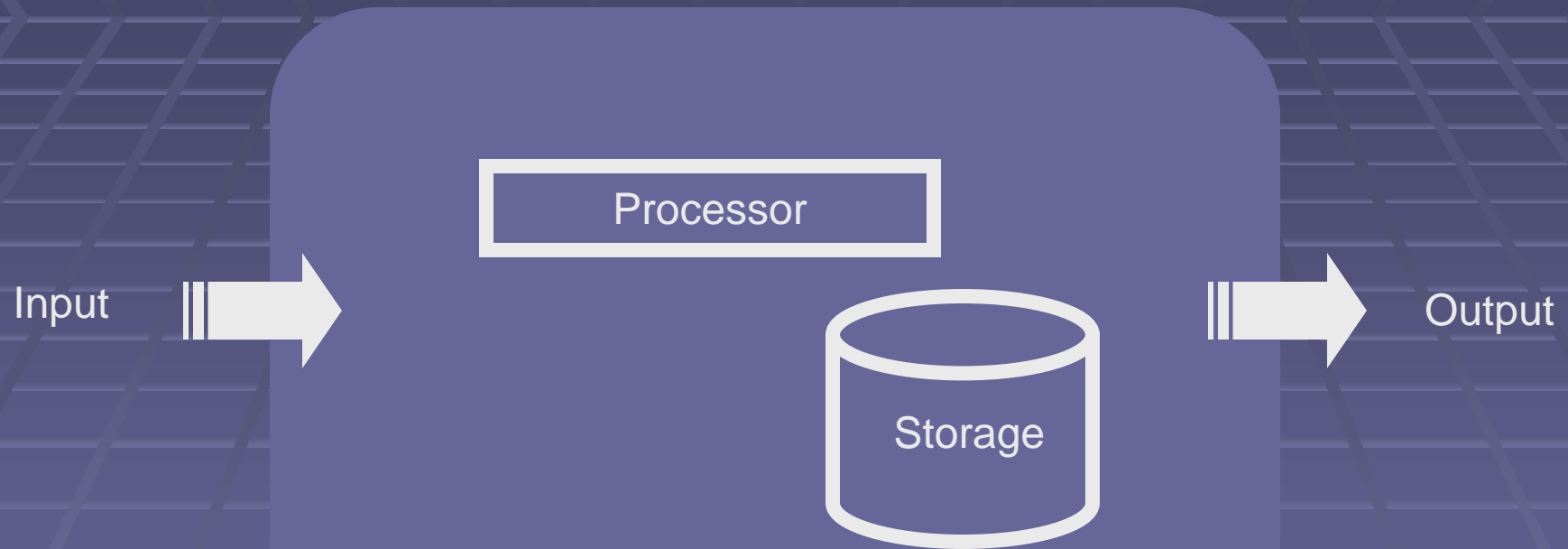


Role of Computers in Engineering

Izad Khormaei
Engineering Department
Clark College
1/22/2005

What is a computer?



Operating Systems: MS Windows, Mac OS, Linux, ...

Where are computers
used?

Computer Usages

- Everyday Living
- Office Productivity
- Engineering Specialized Tools
- Engineering Applications Software

Computers for Everyday Life

- Cars
- Cell Phones
- Home Appliances
- Watches
- Games
- Buildings
- Vending Machines
- ...

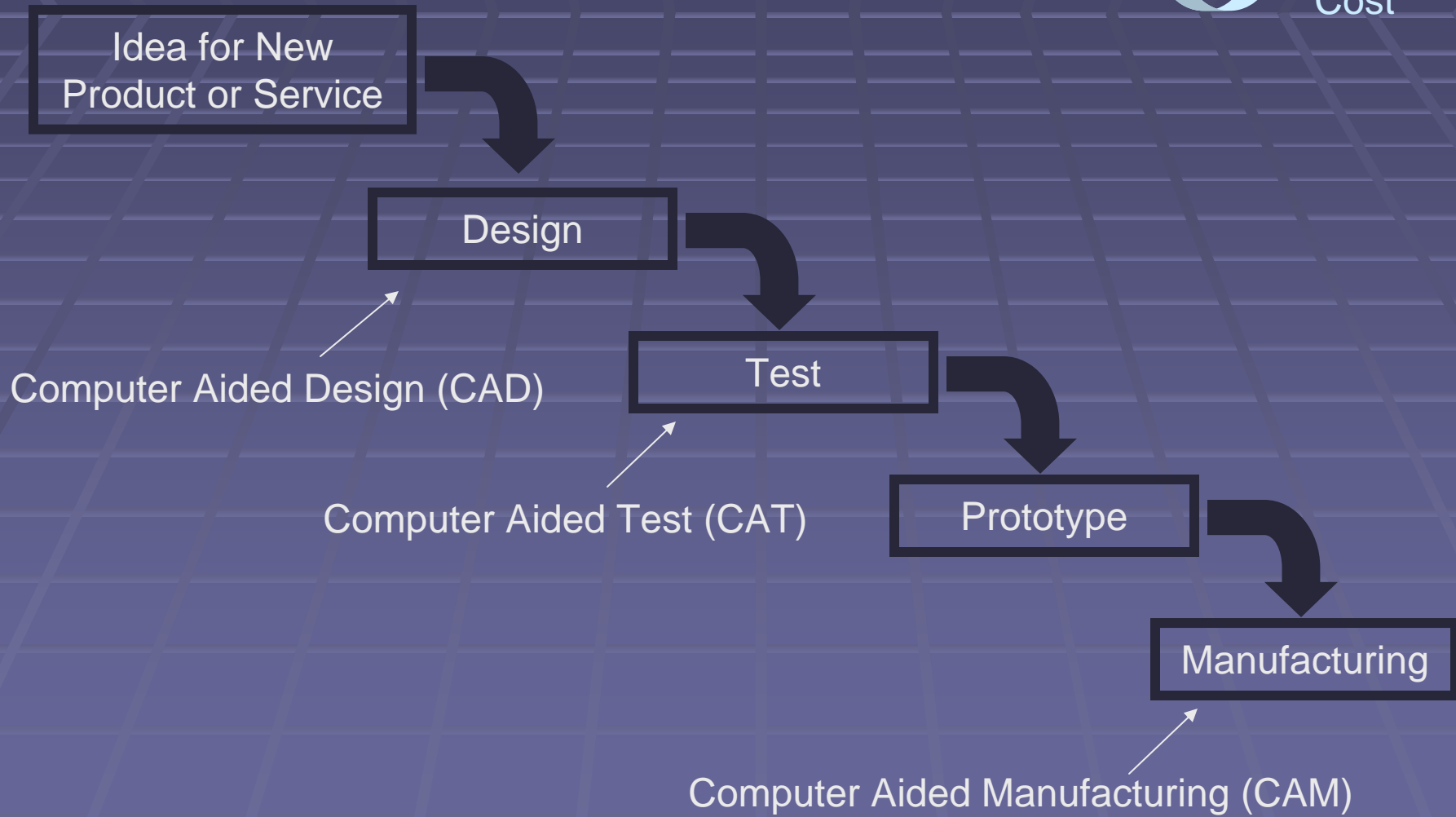
Office Productivity

- Word Processing (MS Word)
- Spread Sheet (MS Excel)
- Presentation (MS PowerPoint)
- Database (MS Access)
- Project Planning (MS Project)
- ...

Engineering Specialized Tools

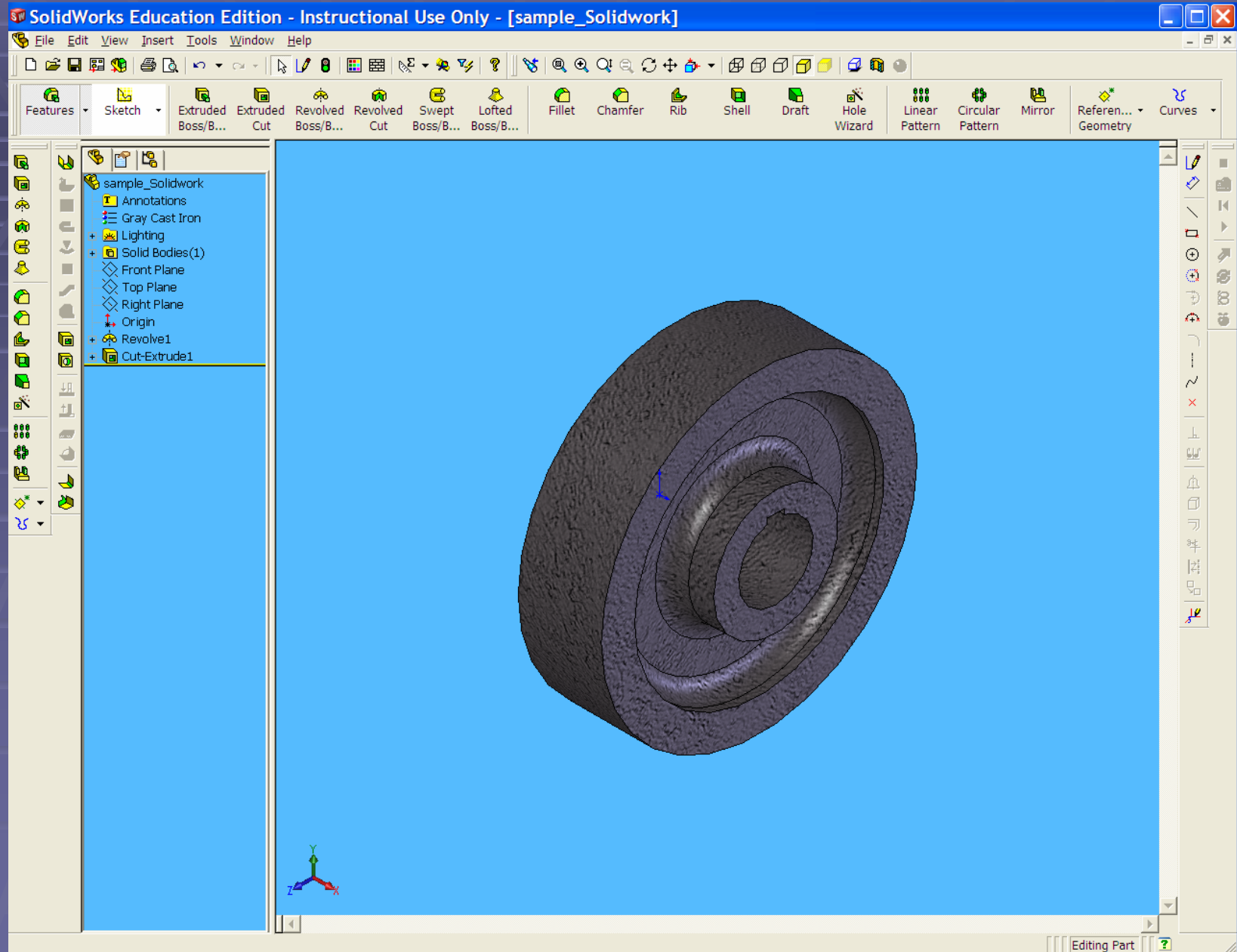
- Calculators (HP 45 and TI 89)
- Surveyor Tool
- Oscilloscope
- Strain Gauge
- Laser Leveler
- Pneumatic Machines
- ...

Engineering Applications Software

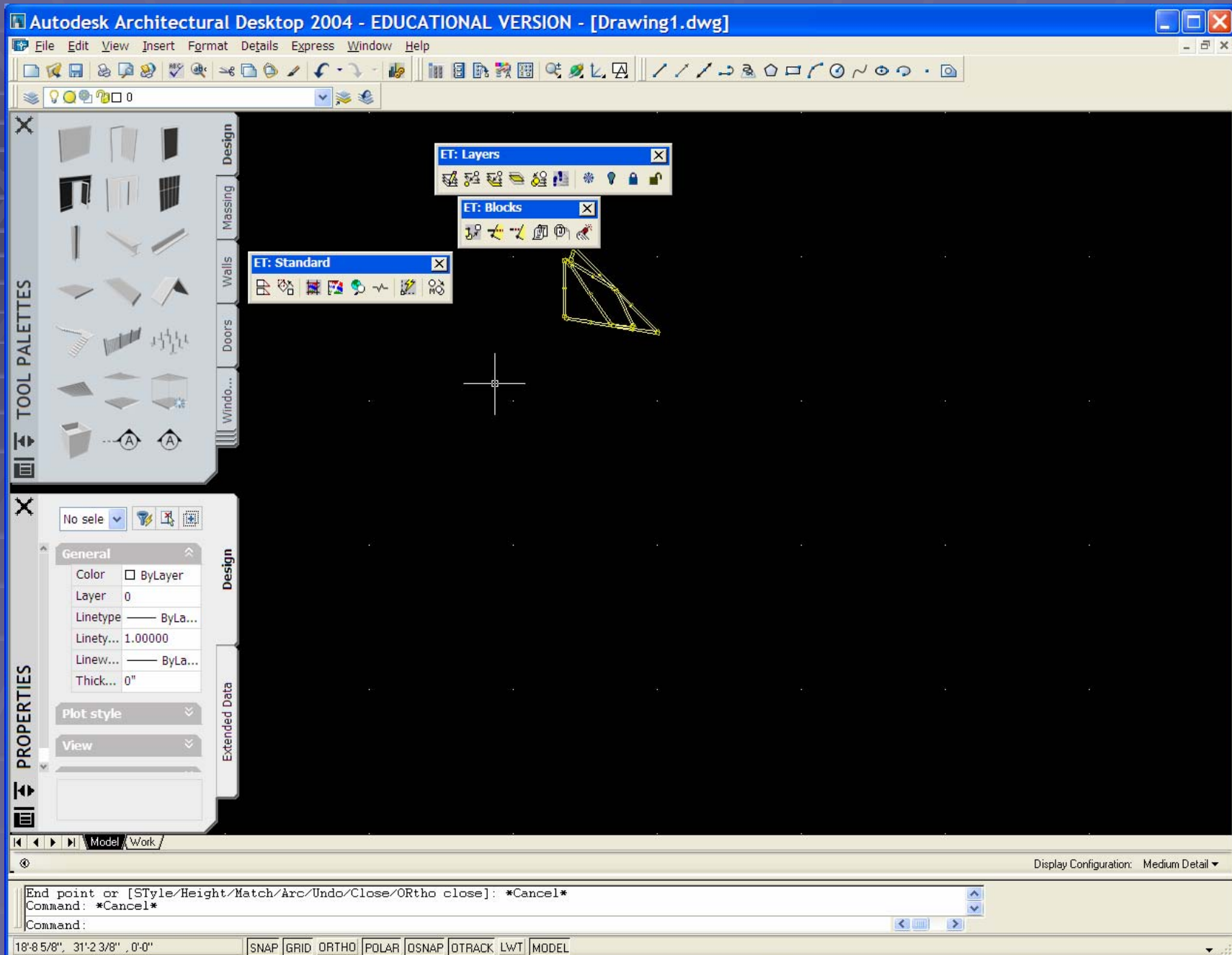


SolidWorks, Parametric Solid Modeler

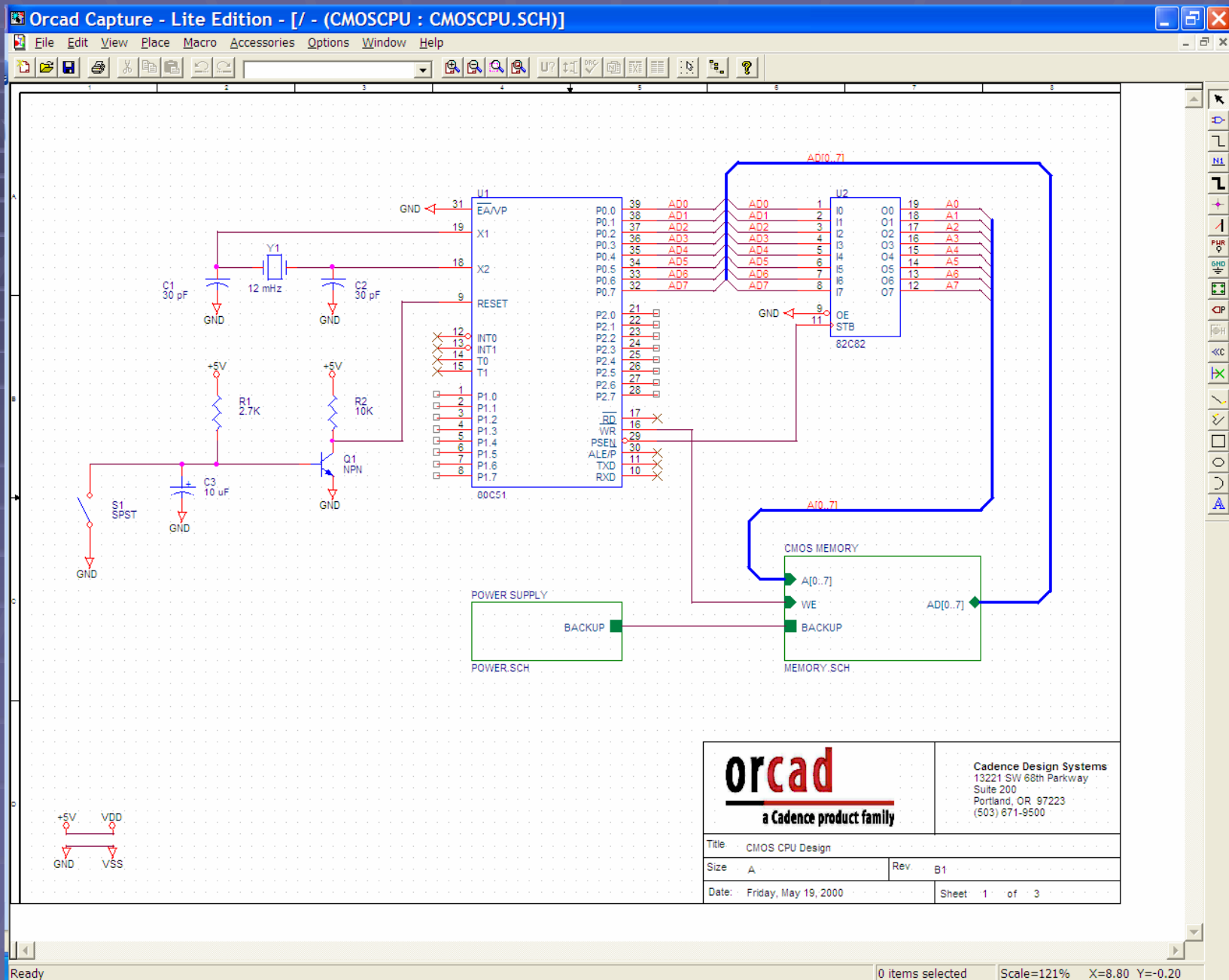
“Changing parameter values will change the drawing”



AutoCAD, Computer Aided Drafting



PSpice, Electronic and Digital Design/Testing Environment



MatLab, Engineering Programming Environment

The screenshot displays the MATLAB software interface. The main window is titled "MATLAB" and shows the "Command Window" with the current directory set to "C:\MATLAB6p5\work". The "Workspace" window is empty. The "Script Editor" window is open to the file "C:\MATLAB6p5\toolbox\matlab\demos\splashdemo.m". The code in the script editor is as follows:

```
1 function splashdemo
2 % SPLASHDEMO Plot the L-shaped membrane with semitranspare
3
4 % Copyright 1984-2002 The MathWorks, Inc.
5 % $Revision: 1.3 $ $Date: 2002/04/08 20:04:53 $
6
7 L = 40*membrane(1,25);
8
9 % Generate a circular alphasdata for the mesh.
10 [j,i] = meshgrid(1:size(L,1));
11 alphadata = sqrt((i-4).^2 + .8*(j+4).^2);
12
13 % This alphasmap causes there to be a thin band that
14 % is semi-transparent, as opposed to the entire surface
15 fig = figure;
16 set(fig, 'Color', 'white',...
17 'AlphaMap',...
18 [ zeros(1,40), ...
19 linspace(0,.5,15), linspace(.5,1,8), ...
20 ones(1,30)]);
21
22 axes('CameraPosition', [-193.4013 -265.1546 220.4819],...
23 'CameraTarget',[26 26 10], ...
24 'CameraUpVector',[0 0 1], ...
25 'CameraViewAngle',9.5, ...
26 'DataAspectRatio',[1 1 .9],...
27 'Position',[0 0 1 1], ...
28 'Visible','off', ...
29 'Xlim',[1 51], ...
30 'Ylim',[1 51], ...
31 'Zlim',[-13 40]);
32
33 % This surface represents a white opaque surface with edge l
34 s1 = surface(L, ...
35 'FaceColor','white', ...
36 'FaceLighting','none', ...
37 'Clipping','off',...
38 'EdgeColor',[.8 .8 .8]);
39 % This represents the colored surface.
```

The "Figure No. 1" window shows a 3D surface plot of the L-shaped membrane. The surface is colored yellow and blue, with a semi-transparent band along the edge. The plot is viewed from a perspective angle, showing the depth of the surface.

VHDL, Hardware Description Language

The screenshot displays the Xilinx Project Navigator interface. The main window shows the VHDL code for an entity named `LFSR_B`. The code defines a generic `cycleB0` (integer, default 26) and `cycleB20` (integer, default 21), and a port `width` (integer, default 1). The entity has a port `Clk` of type `std_logic`, and two enable signals `Enable` and `Fill_En` of type `std_logic`. It also has an output port `New_Fill` of type `std_logic_vector` and a delay output port `DelayB0` of type `std_logic_vector`. The code includes attributes for `clock_node` and a process `main` that implements the LFSR logic. The process `main` is triggered by the clock edge and updates `int_sigB0` and `int_sigB20` based on the enable signals and the current state of the LFSR. The code also includes a process `delayB0_int` that updates the delay output.

```
1 library ieee;
2 use ieee.std_logic_1164.all;
3
4 entity LFSR_B is
5 generic (cycleB0 : integer := 26;
6         cycleB20 : integer := 21;
7         width : integer := 1);
8 port ( Clk :in std_logic;
9       Enable : in std_logic;
10      Fill_En : in std_logic;
11      New_Fill : in std_logic_vector(width -1 downto 0);
12      DelayB0 :out std_logic_vector(width - 1 downto 0));
13
14 attribute clock_node :boolean;
15 attribute clock_node of Clk : signal is TRUE;
16
17 end LFSR_B;
18
19 architecture LFSR_B_ARCH of LFSR_B is
20
21 signal Data_In_B : STD_LOGIC_VECTOR(width -1 downto 0);
22 signal DelayB20 : STD_LOGIC_VECTOR(width -1 downto 0);
23 signal DelayB0_int : STD_LOGIC_VECTOR(width -1 downto 0);
24 type my_type is array (0 to cycleB0 -1) of std_logic_vector(width -1 downto 0);
25 signal int_sigB0 :my_type;
26 type my_type2 is array (0 to cycleB20 -1) of std_logic_vector(width -1 downto 0);
27 signal int_sigB20 :my_type2;
28
29 begin
30
31 main :process (Clk)
32 begin
33 if Clk'event and Clk = '1' then
34   if (Enable = '1') then
35     int_sigB0 <= Data_In_B & int_sigB0(0 to cycleB0 - 2);
36     int_sigB20 <= Data_In_B & int_sigB20(0 to cycleB20 - 2);
37   end if;
38   if (Fill_En = '0') then
39     Data_In_B <= DelayB20 xor DelayB0_int;
40   else
41     Data_In_B <= New_Fill;
42   end if;
43 end if;
44 end process main;
45
46 delayB0_int <= int_sigB0(cycleB0 -1);
47 delayB20 <= int_sigB20(cycleB20 - 1);
```

Warning: This process is used to display the running command log file that records some application command lines.
If you haven't run any process yet, this file couldn't be generated. Please run some processes to create the running command log file.

Questions / Comments

Almost Done!