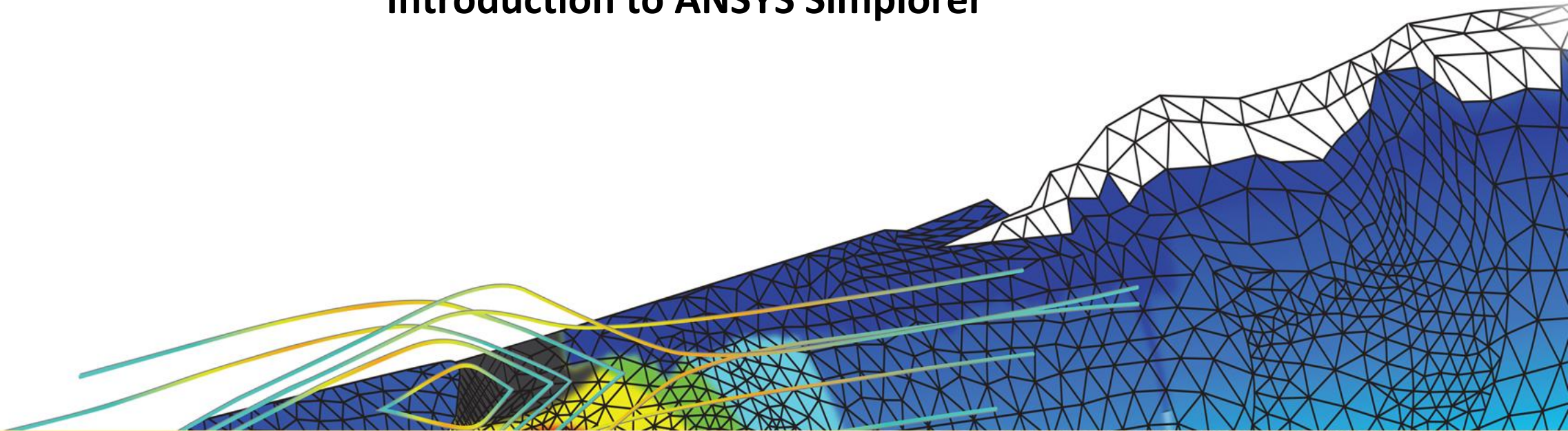




# Workshop 2.1: DC Motor control

## Introduction to ANSYS Simplorer





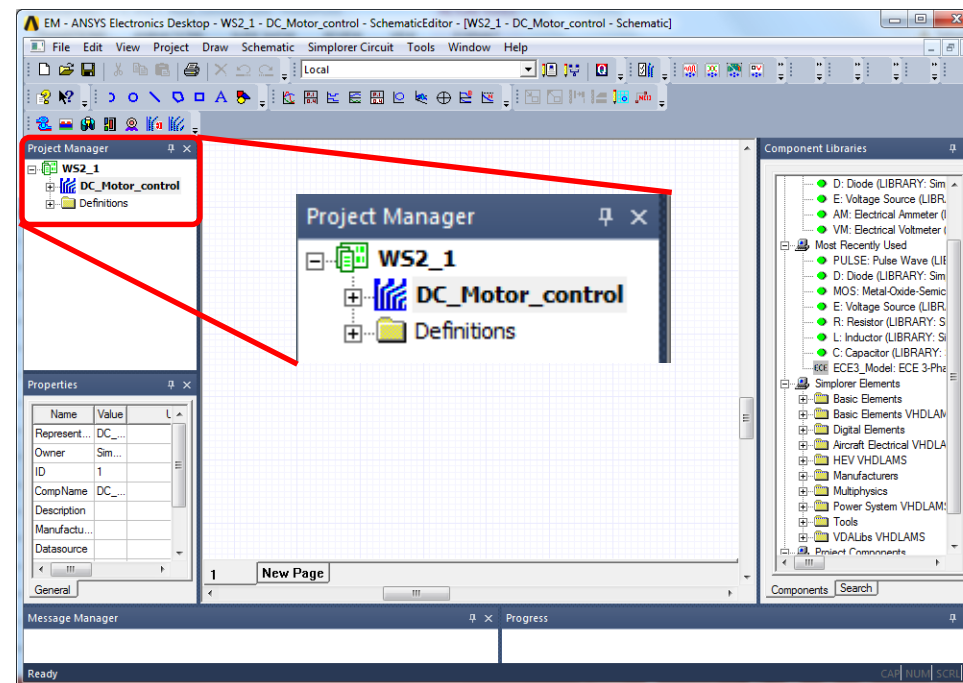
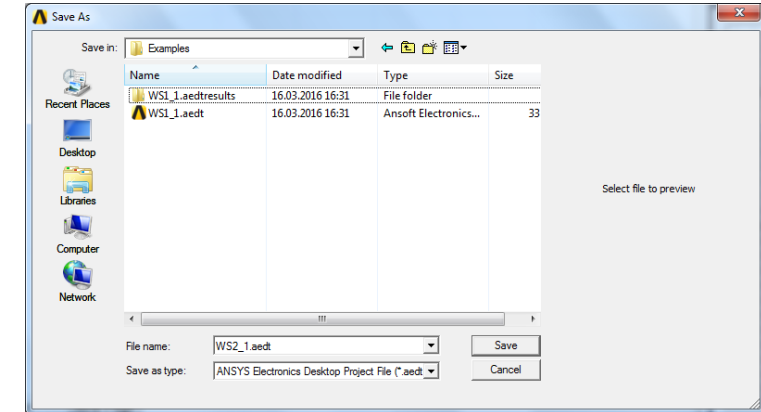
# Overview

- **DC Motor Control**

- In this example we will build a simple AC/DC Diode rectifier feeding a DC Permanent Magnet Motor. We will also build a complete control block scheme intended to drive the motor in case of start up and sudden load changes
- In particular we will learn:
  - How to create and use Datasets to specify component properties
  - How to connect components without physical wire connections
  - How to pass information between components
  - How to use different blocks to create a complex control structure
  - How to plot different quantities (signals) in the same graph

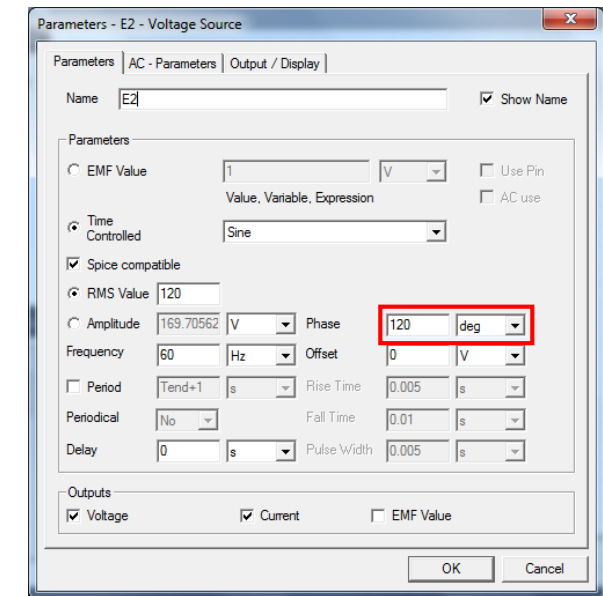
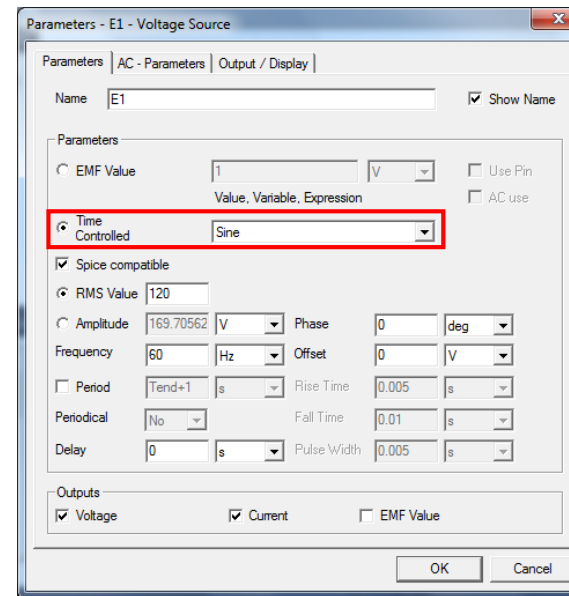
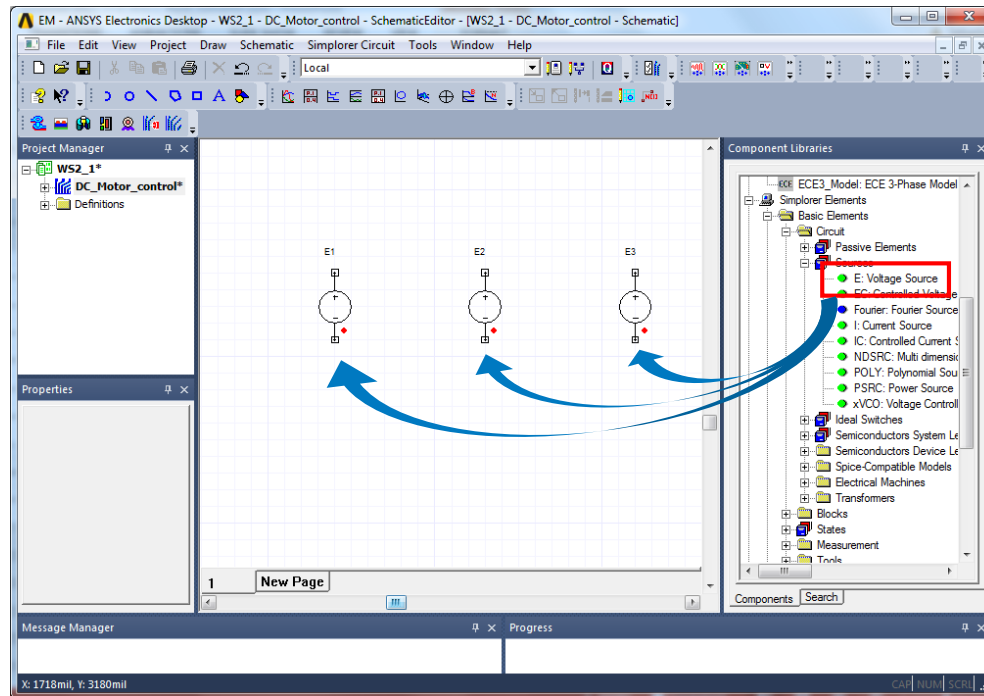
# Insert a Simplorer Design

- Launch the Electronics Desktop 2016
  - Save the Project as **WS\_2\_1.aedt**
  - Insert a Simplorer Design using the icon 
  - Rename the Design as **DC\_Motor\_Control**
  - Save again the project using the icon 



# Insert Components – Voltage Sources

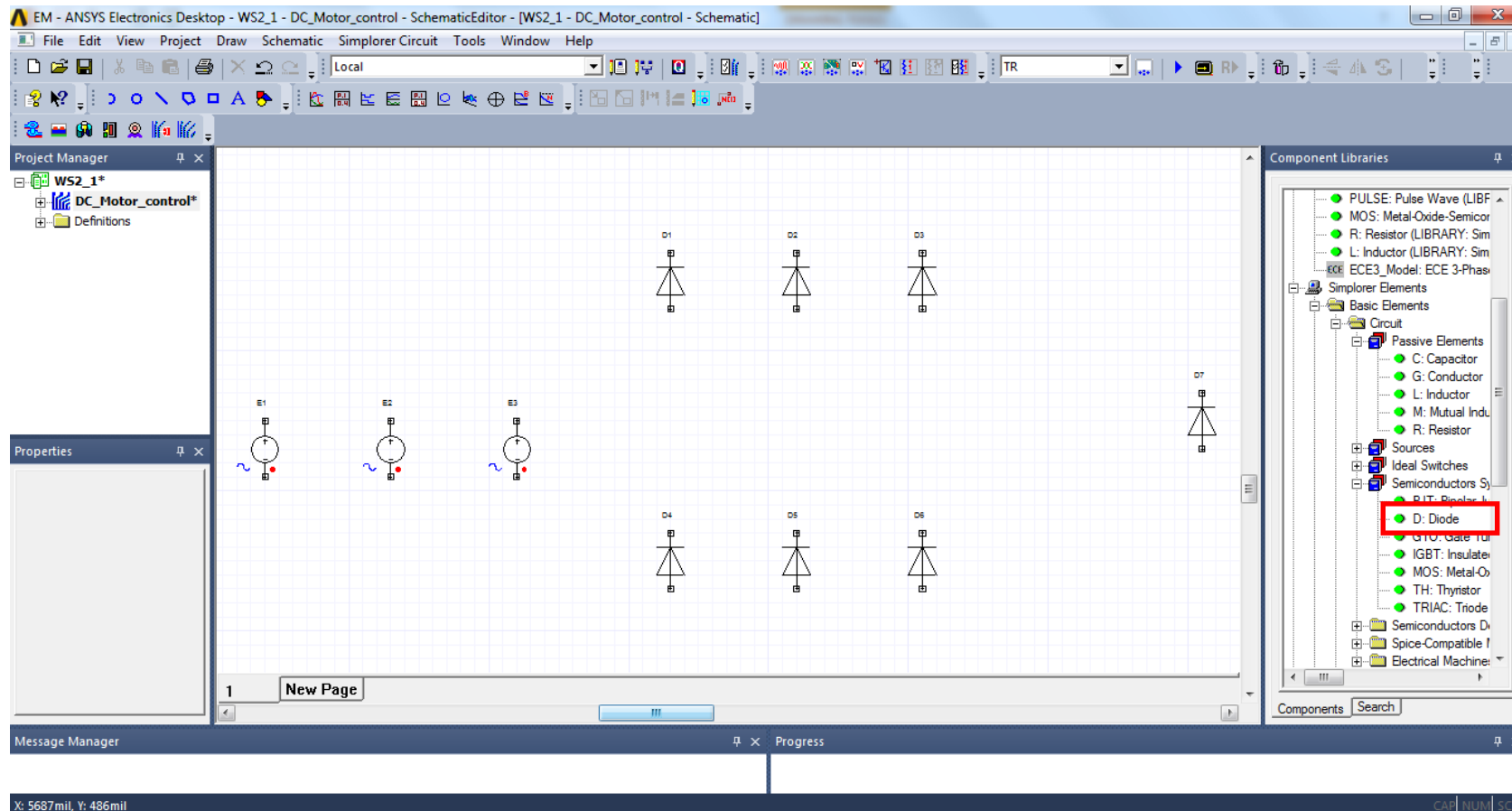
- In Component Libraries window *Simplorer Elements* → *Basic Elements* → *Circuit* → *Sources*
- Select the **E: Voltage Source**, drag and drop it three times into the Schematic
- Double click on **Voltage Source E1**, select **Time Controlled** → **Sine** to make it sinusoidal
- Spice Compatible: ☒ Checked
- Select **RMS Value** to be 120 Vac and **Frequency** to be 60Hz, Press **OK**
- Repeat the procedure for E2 and E3, adding **Phase 120 deg** and **240 deg** respectively





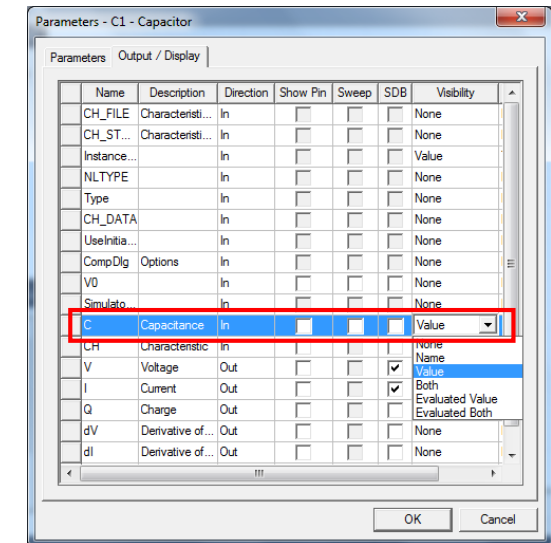
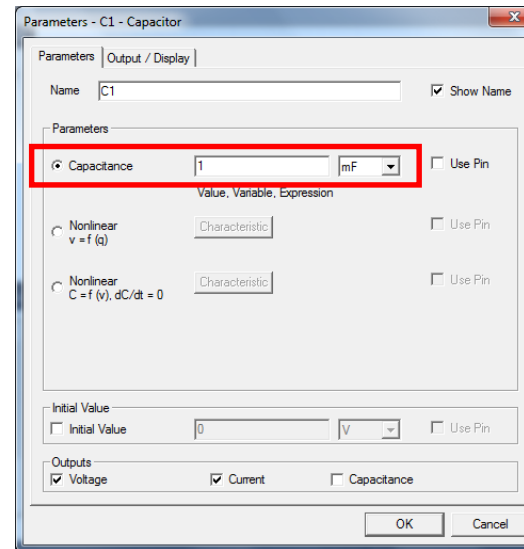
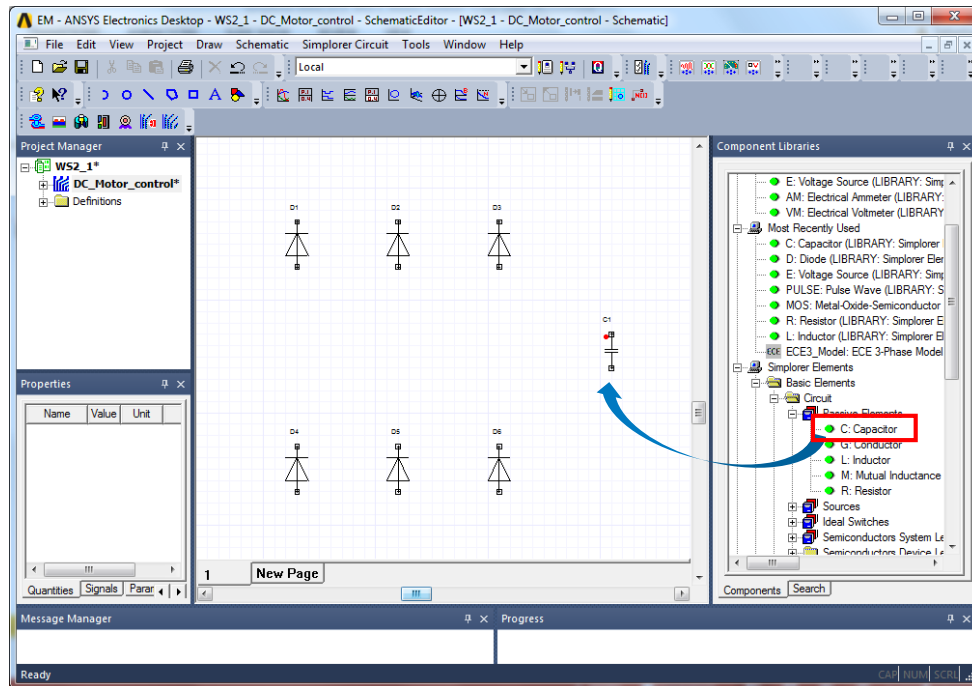
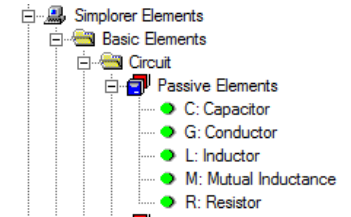
# Insert Components – Diodes

- In Comp. Libraries window *Simplorer Elements* → *Basic Elements* → *Circuit* → *Semiconductor system Level*
- Select the Diode, drag and drop it seven times into the Schematic
- Press **Esc** key to exit the insert mode and use the shortcut **Ctrl+D** to fit all



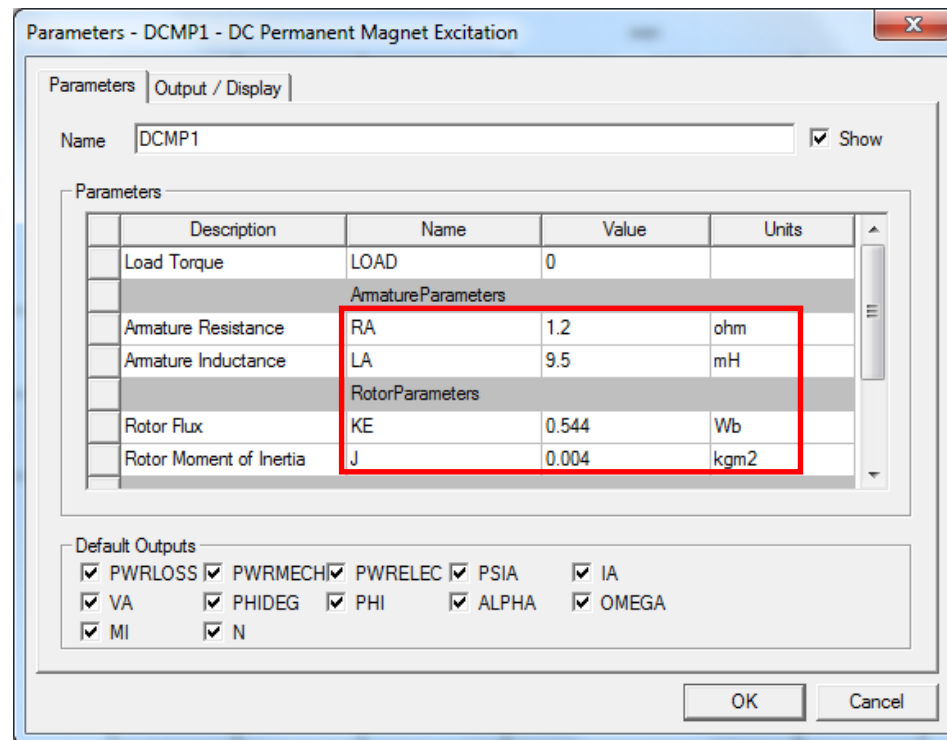
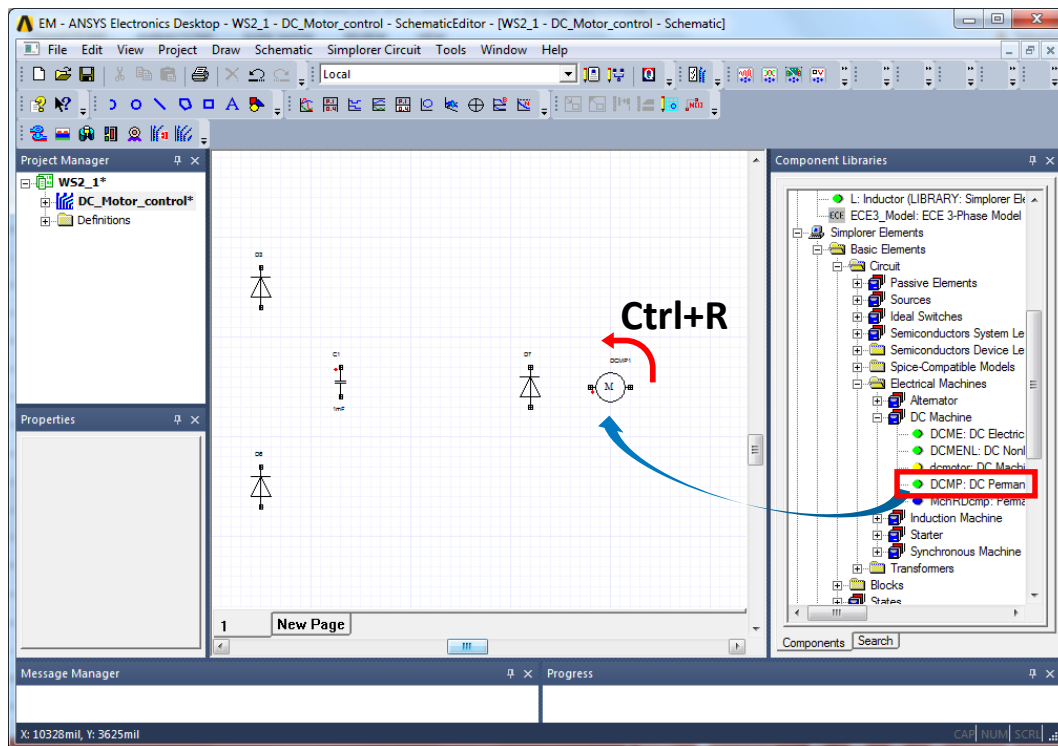
# Insert Components - Capacitor

- In Component Libraries window *Simplorer Elements* → *Basic Elements* → *Circuit* → *Passive Elements*
- Select the **Capacitor**, drag and drop it into the Schematic
- Press **Esc** key to exit the insert mode
- Double click on the Capacitor, change the value to **1 mF**
- In the **Output/Display Tab** under Visibility, select Value for Capacitance



# Insert Components – DC Motor

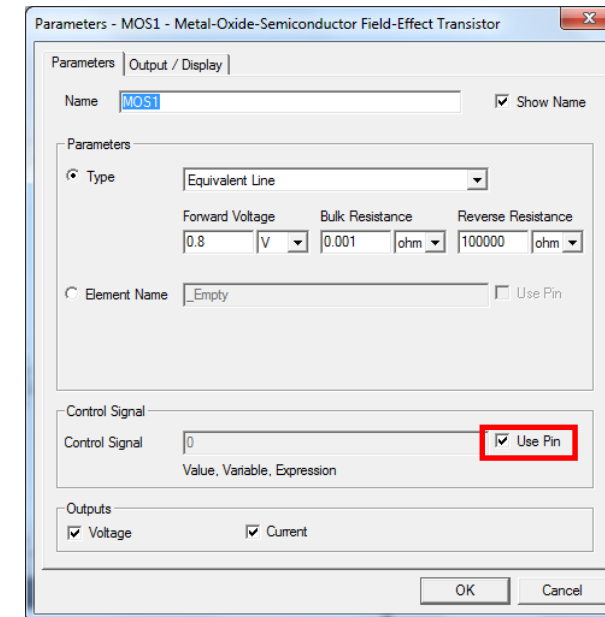
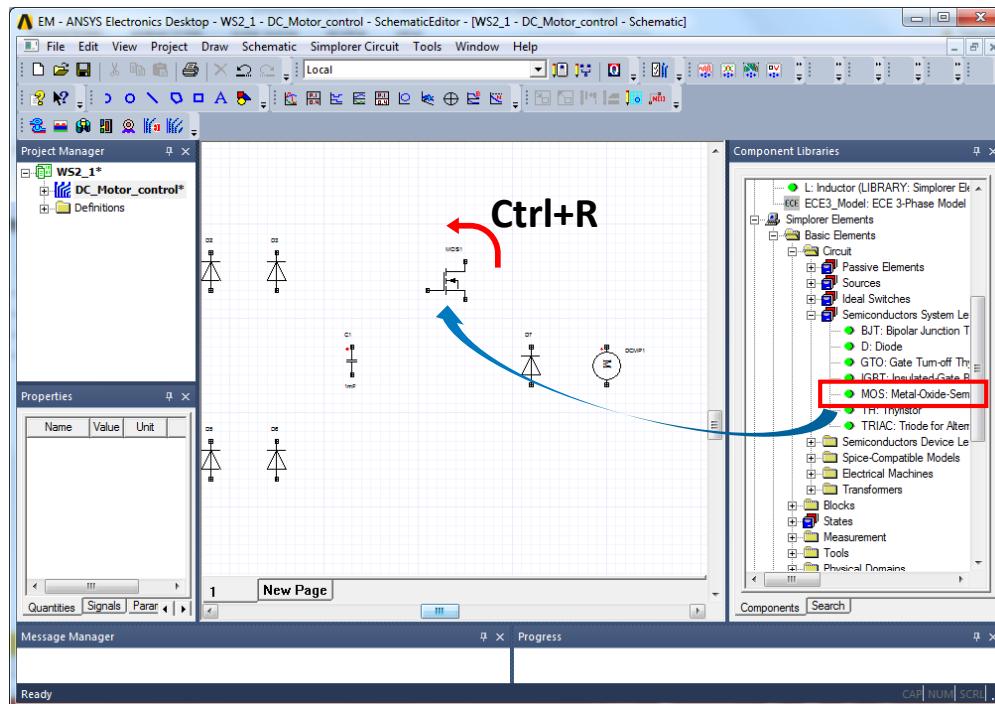
- Select **Simplorer Elements** → **Basic Elements** → **Circuit** → **Electrical Machines** → **DC Machine**
- Select the **DCMP**, drag and drop it into the Schematic. Press **Esc** key to exit the insert mode
- Use 3 times the shortcut **Ctrl+R** to rotate **DCMP** component 270 deg counterclockwise
- Double click on DCMP and set  $RA = 1.2\ \Omega$ ,  $LA = 9.5\ \text{mH}$ ,  $KE = 0.544\ \text{Wb}$ ,  $J = 0.004\ \text{Kgm}^2$ , as reported in Figure



# Insert Components


- **Electronics Switch (Mosfet)**

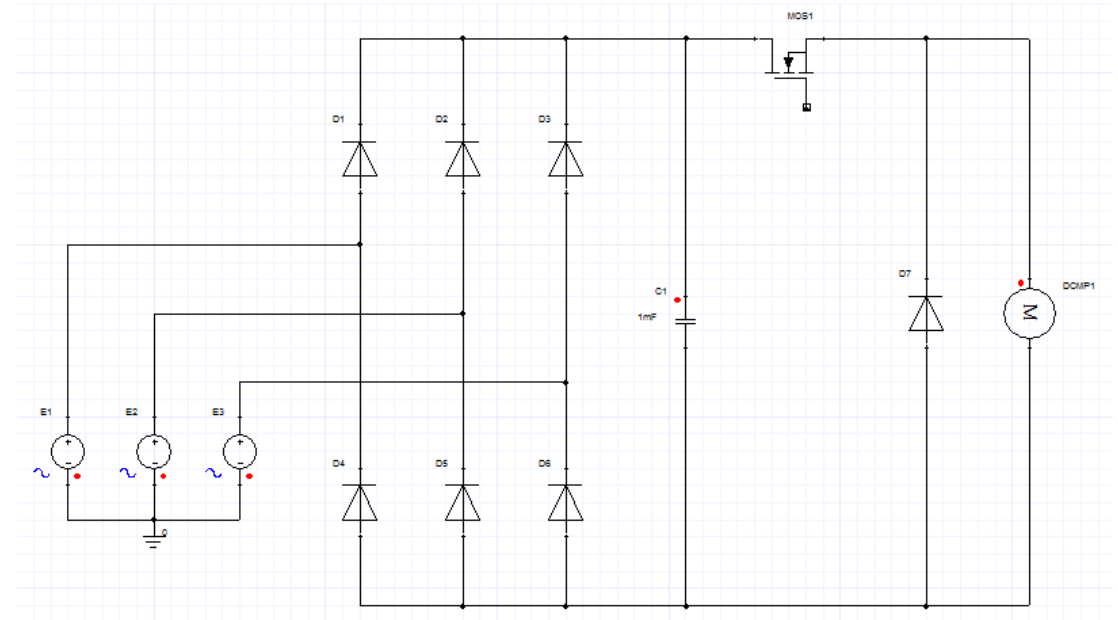
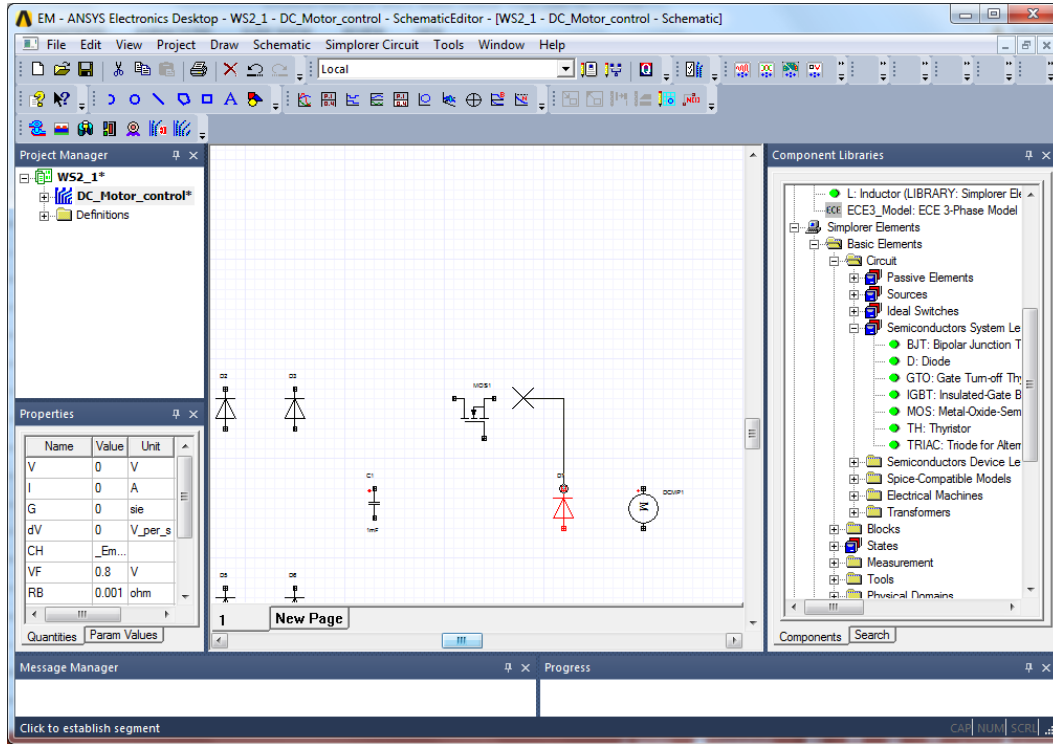
- In Comp. Libraries window *Simplorer Elements* → *Basic Elements* → *Circuit* → *Semiconductor system Level*
- Select the Mosfet (MOS), drag and drop it into the Schematic
- Select the MOS and use the shortcut **Ctrl+R** to rotate it 90 degrees counterclockwise
- Be sure **Use Pin** is checked





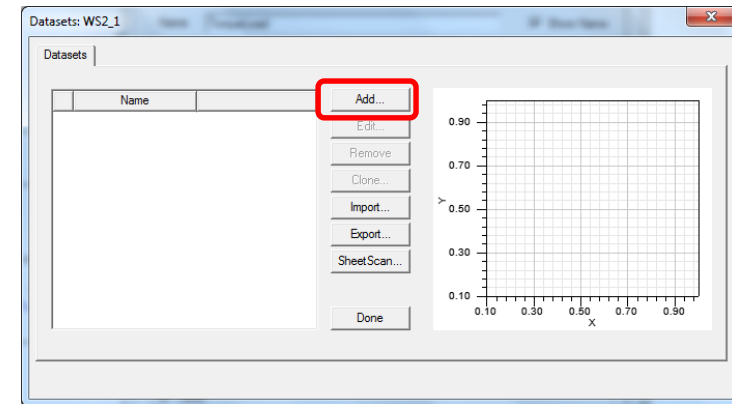
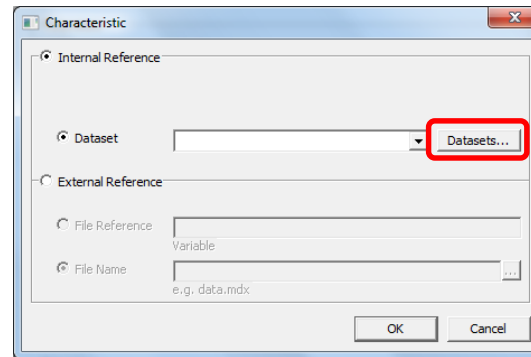
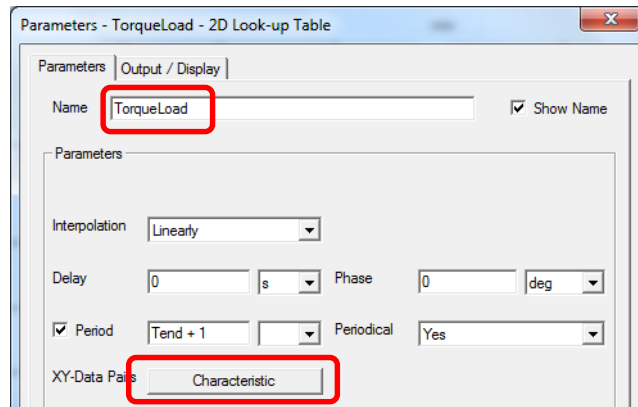
# Connect Components

- Place the mouse over one terminal of a component. The mouse pointer changes its shape becoming a cross. Press the **LMB** and move the cross till the connecting terminal of next component
- Add the **Ground node** clicking the icon  and placing it into the Schematic
- Connect all the components till completing the circuit as in figure



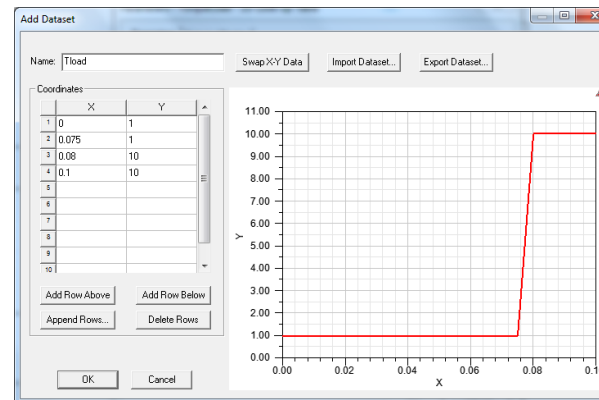
# Creating Dataset

- A dataset will be now created intended to be used as representing the DC motor load torque
- In Comp. Libraries window *Simplorer Elements* → *Basic Elements* → *Tools* → *Time Functions*
- Select the **DATAPAIRS** component, drag and drop it into the Schematic
- Double click on the DATAPAIRS block, change the name to **TorqueLoad**, then click on the **Characteristic** button. Select **Datasets...** button, then **Add** (to create new Dataset)



- Change the name to **Tload** and enter time/value pair per the following

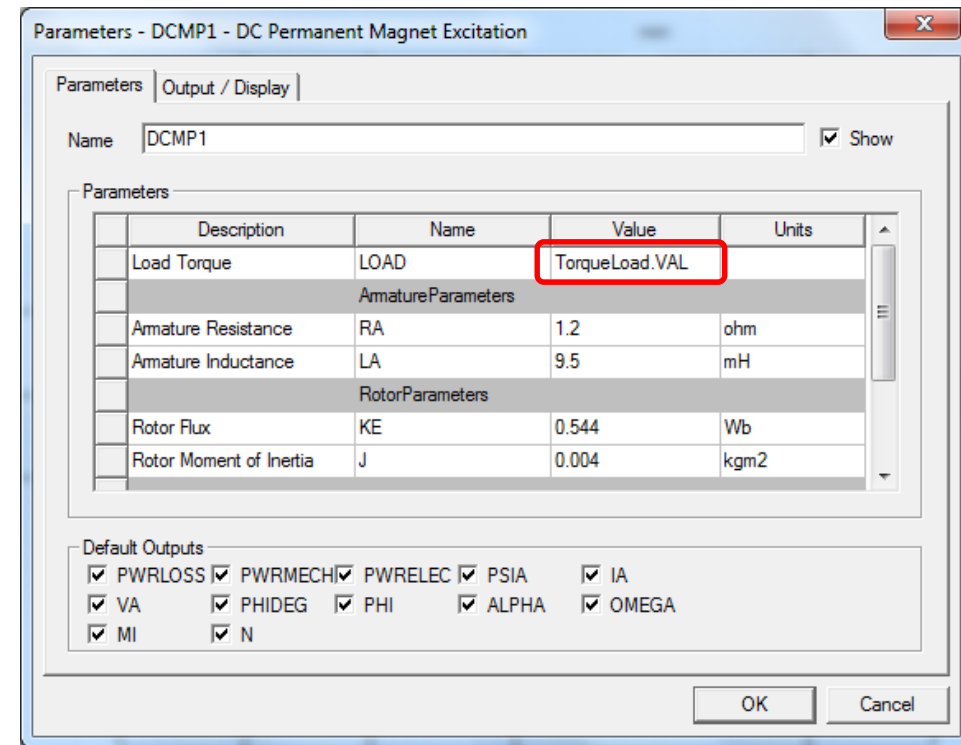
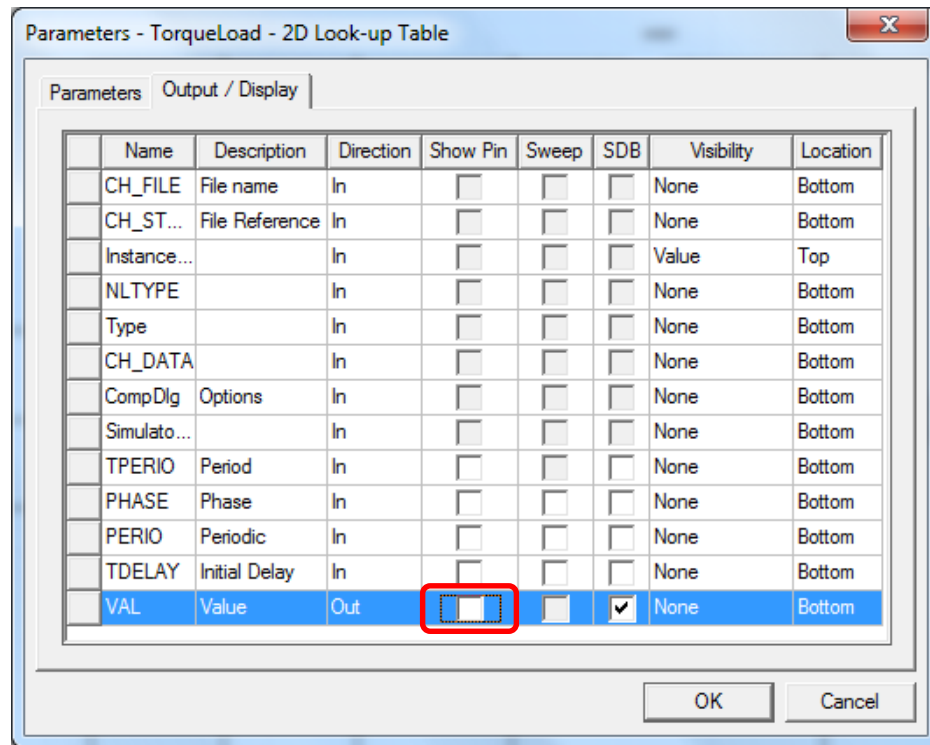
0	1
0.075	1
0.08	10
0.1	10



- Press **OK**, then **Done**, then **OK**, then **OK** to get out of all windows

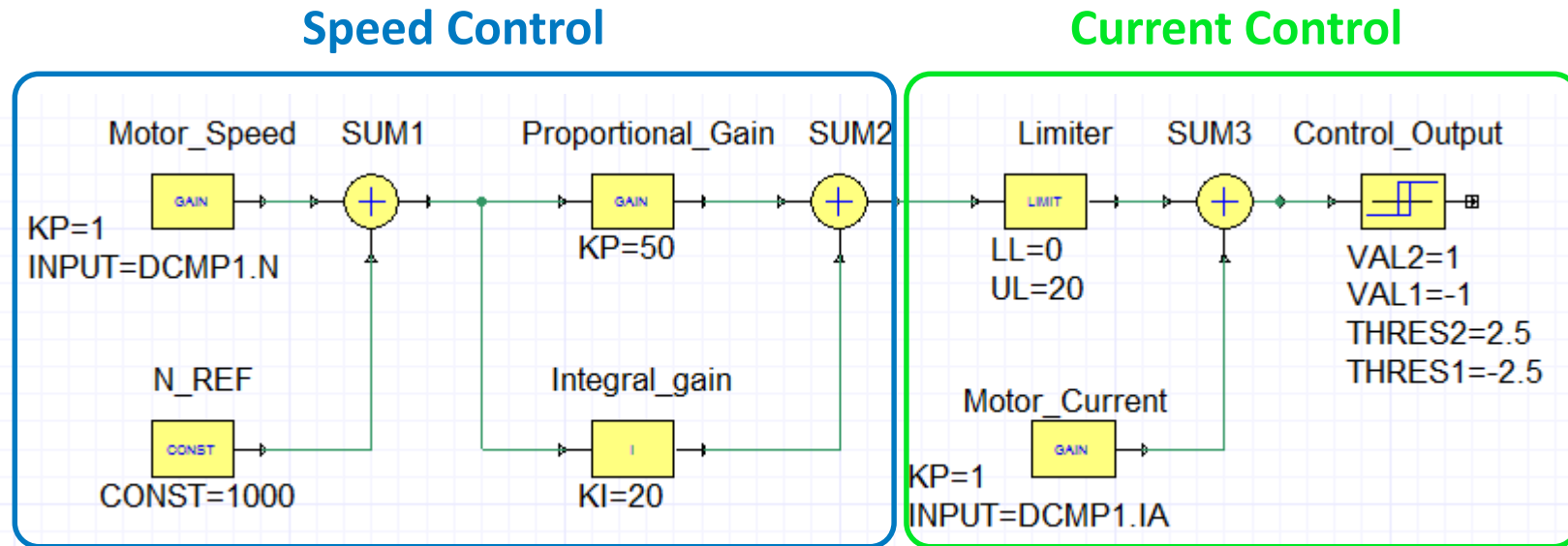
# Add New Load Torque to DC Motor

- Double click again on the **TorqueLoad** table lookup block, select the **Output/Display** tab, un-check the box for the output **Show Pin** for the output value **VAL**
- Double click on the **DC Motor** component and define the load input to be the output of the just created table lookup block specifying **TorqueLoad.VAL**



# Control Scheme

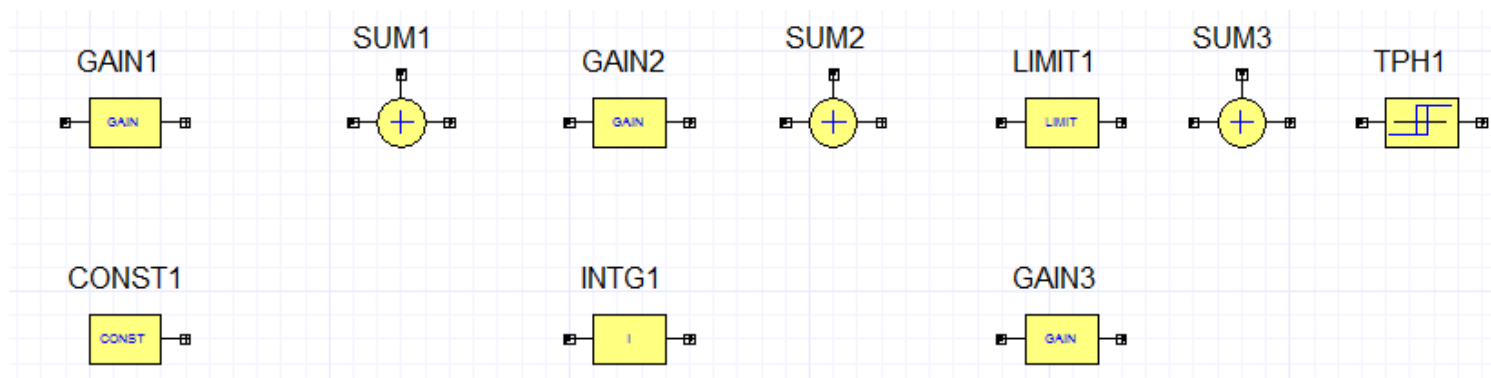
- The control scheme is composed by an outer **Speed Control** loop coupled together with an inner **Current Control** loop
- The two controls have different goals:
  - The Speed Control drives the motor to a fixed speed of **1000 rpm**
  - The Current Control makes the current not exceeding a maximum value, confining it into a bound of  **$\pm 2.5$  A** around **20 A**
- The complete scheme to be implemented is shown in the following figure:



# Insert Blocks

- Control Scheme Blocks

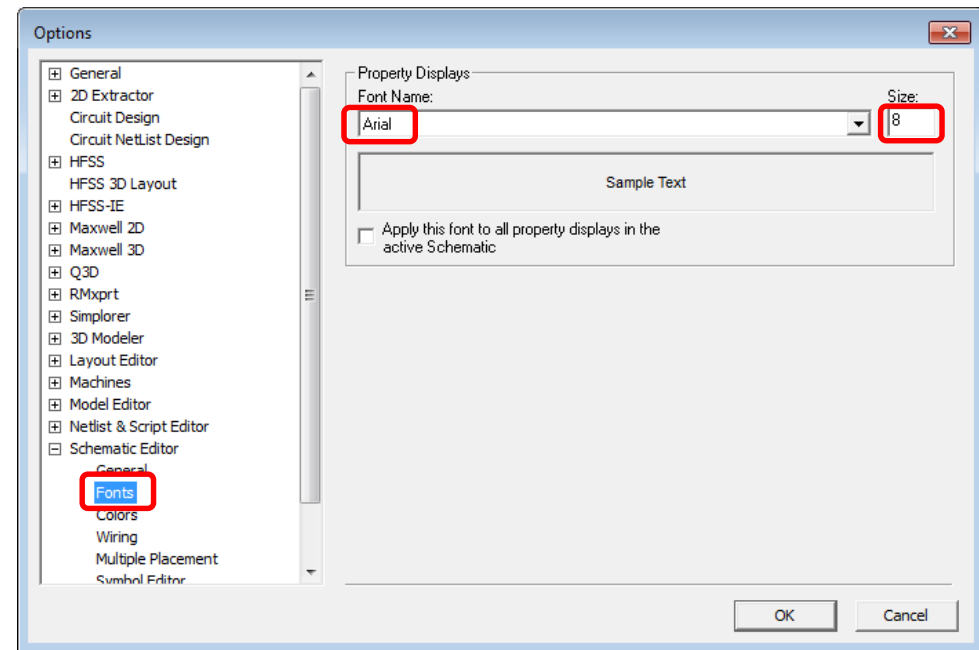
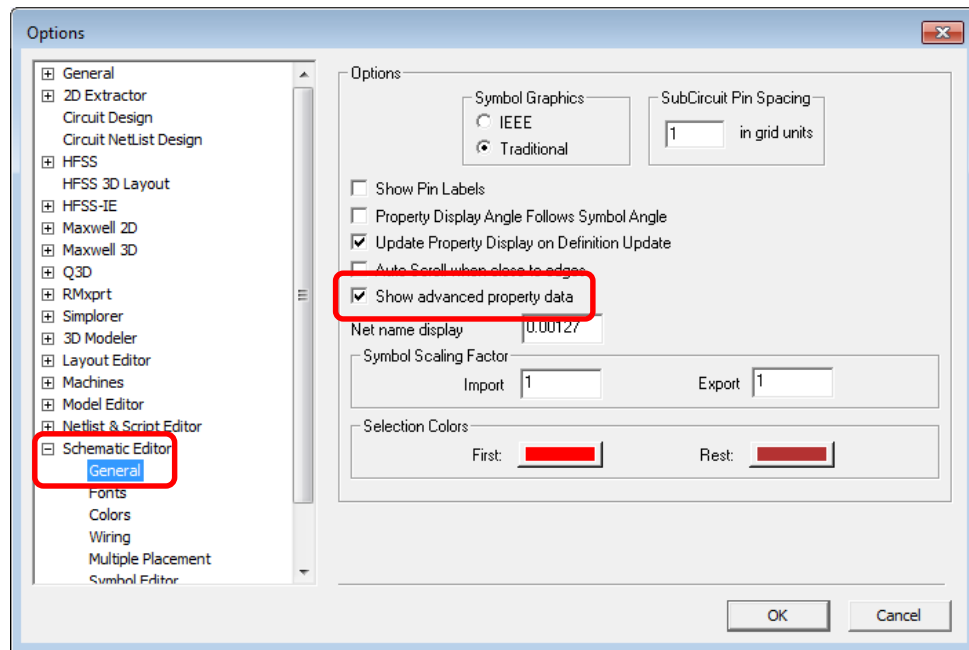
- In Component Libraries window *Simplorer Elements* → *Basic Elements* → *Blocks* → *Continuous Blocks*
- Select the **GAIN** block and drag and drop it 3 times into the Schematic
- Select the **INTG** (Integrator) block and drag and drop it into the Schematic
- In Component Libraries window *Simplorer Elements* → *Basic Elements* → *Blocks* → *Sources Blocks*
- Select the **CONST** (Constant Value) block and drag and drop it into the Schematic
- In Comp. Libraries window *Simplorer Elements* → *Basic Elements* → *Blocks* → *Signal Processing Blocks*
- Select the **SUM** block and drag and drop it 3 times into the Schematic
- Select the **TPH** (Two-Point Element with Hysteresis) block and drag and drop it into the Schematic
- Select the **LIMIT** (Limiter) block and drag and drop it into the Schematic
- Arrange the 10 blocks as in the figure:





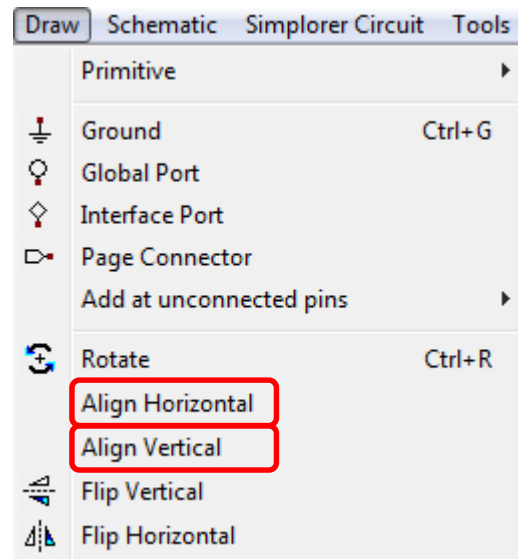
# Component Visibility and Properties

- For a better component visibility on the schematic it is suggested to:
  - Check the Advanced Properties going to menu item **Tools → Options → General Options → Schematic Editor → General** and eventually check box “Show advanced property data”
  - Under **Tools → ... → Schematic Editor → Fonts** select the desired Font type and Font size. You can also check box “Apply this font ... active Schematic” to modify the visibility of elements already placed onto the schematic. When ready, press OK.



# Components Alignment

- To make the block schemes and in general every circuit more readable, it is often helpful to use the Alignment capabilities Simplorer offers:
  - To Align elements/blocks horizontally, first select the elements/blocks to be aligned and then go to the menu item *Draw → Align Horizontal*
  - To Align elements/blocks vertically, first select the elements/blocks to be aligned and then go to the menu item *Draw → Align Vertical*



# Modify Blocks Properties

- Gain Blocks

- Double click on GAIN1 and change the name to **Motor\_speed**
- In the **Output/Display Tab**, uncheck the **Show Pin** for **INPUT**

- Parameters Tab

- INPUT: **DCMP1.N** (DC Motor Speed)
- KP: **1**
- TS: **0**

- Double click on GAIN2 and change the name to **Proportional\_Gain**

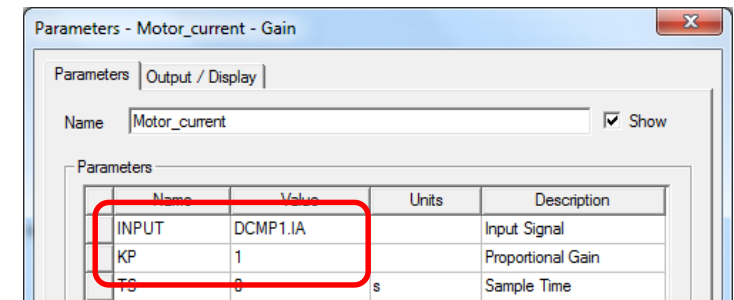
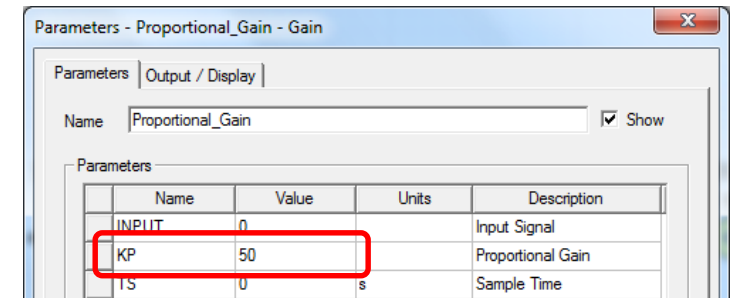
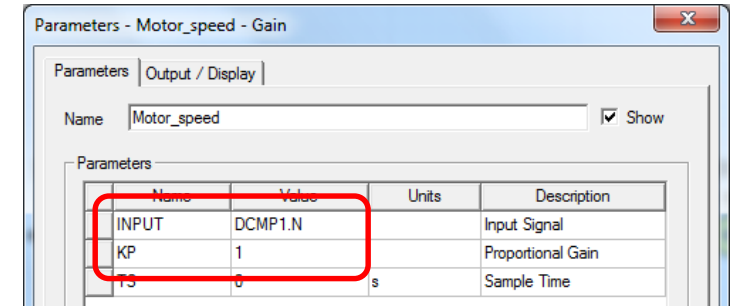
- Parameters Tab

- KP: **50**

- Double click on GAIN3 and change the name to **Motor Current**
- In the **Output/Display Tab**, uncheck the **Show Pin** for **INPUT**

- Parameters Tab

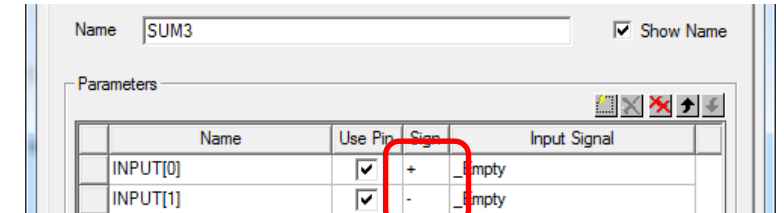
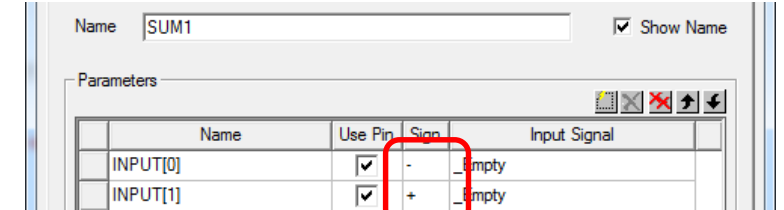
- INPUT: **DCMP1.IA** (DC Motor Current)
- KP: **1**
- TS: **0**



# Modify Blocks Properties

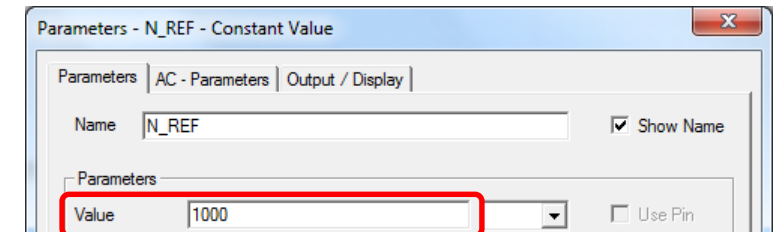
- SUM Blocks

- Select SUM1, SUM2 and SUM3 pressing Ctrl button
- *RMB → Flip Vertical*
- Double click on SUM1 and set the Sign for INPUT[0] to “-”
- Leave SUM2 as it is
- Double click on SUM3 and set the Sign for INPUT[1] to “-”



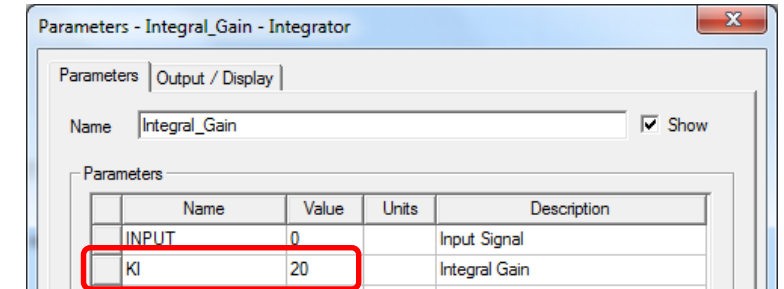
- CONST Block

- Double click on CONST1 and change the name to **N\_REF**
- Value: **1000**



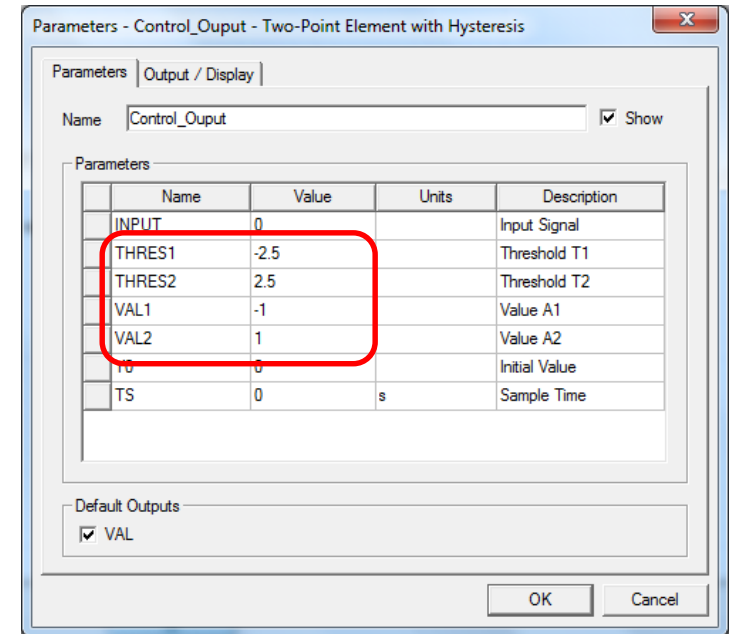
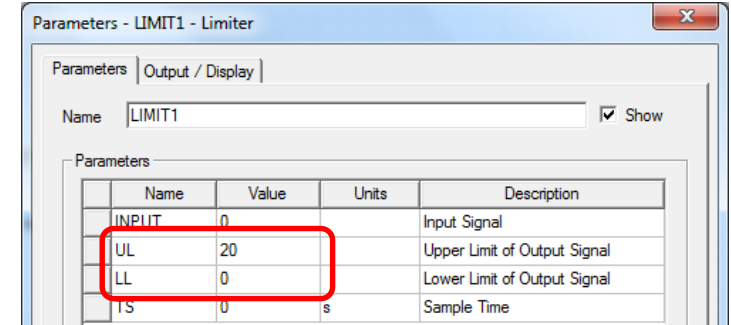
- INTG Block

- Double click on INTG1 and change the name to **Integral\_Gain**
- Parameters Tab
  - KI = **20**



# Modify Blocks Properties

- **Limiter Block**
  - Double click on LIMIT1 and change the name to **Limiter**
  - Parameters Tab
    - UL: **20**
    - LL: **0**
- **TPH Block (Schmitt Trigger)**
  - Double click on TPH and change the name to **Control\_Output**
  - Parameters Tab
    - THRES1: **-2.5**
    - THRES2: **2.5**
    - VAL1: **-1**
    - VAL2: **1**

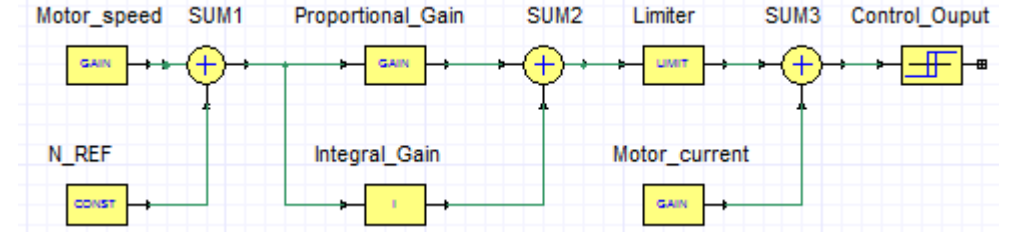




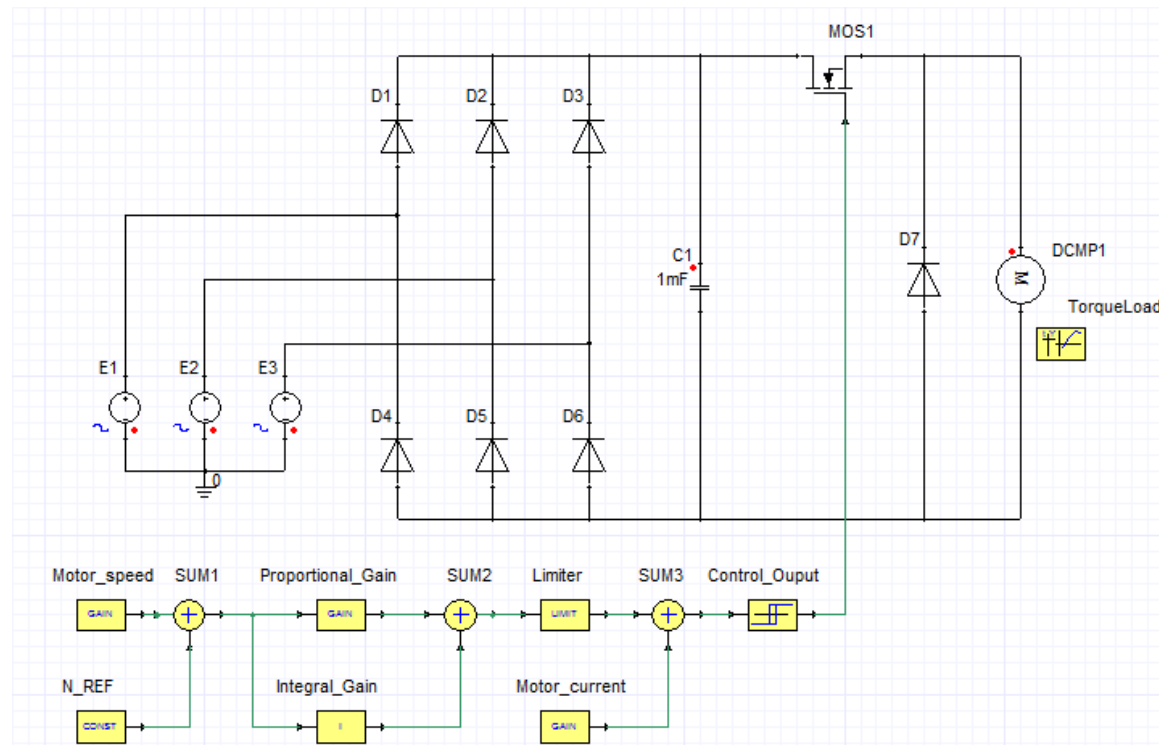
# Connect Blocks and Control Scheme

- Connect Blocks

- Connect all the 10 blocks as shown in figure

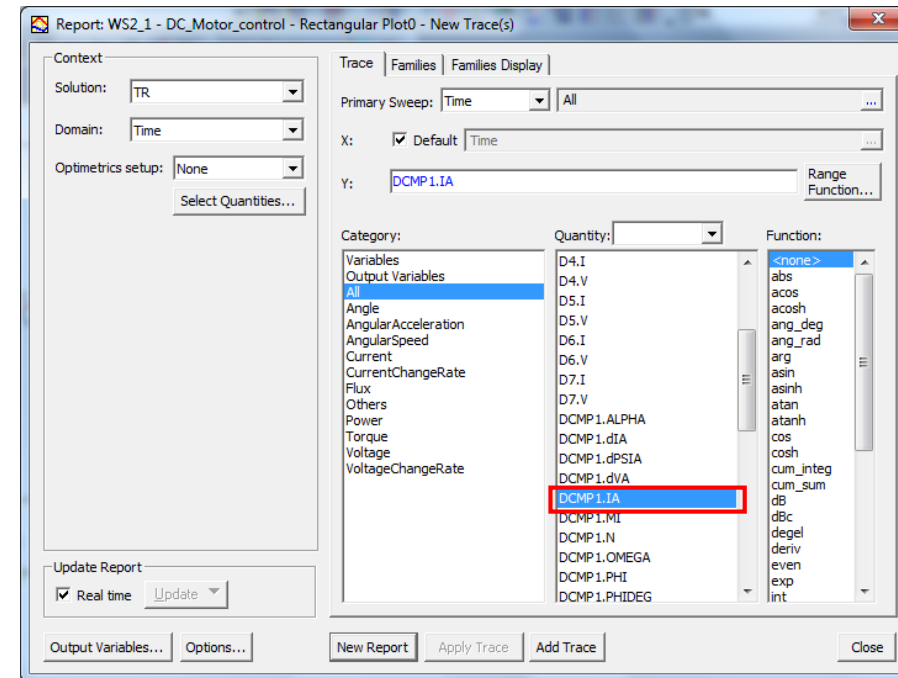
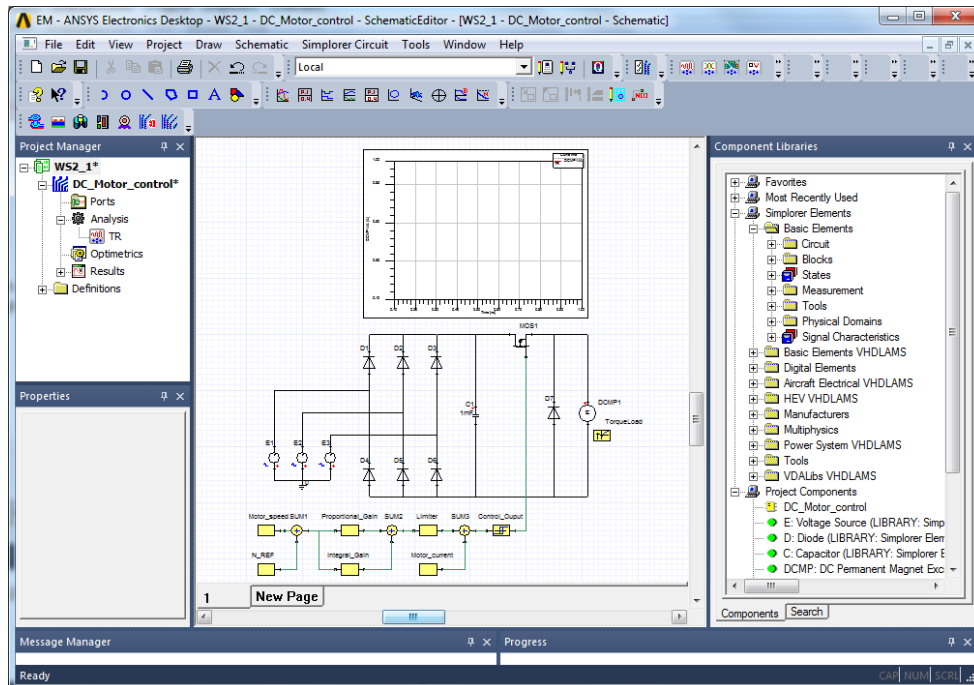


- Connect the **Control\_Ouput** block to the Mosfet's Gate



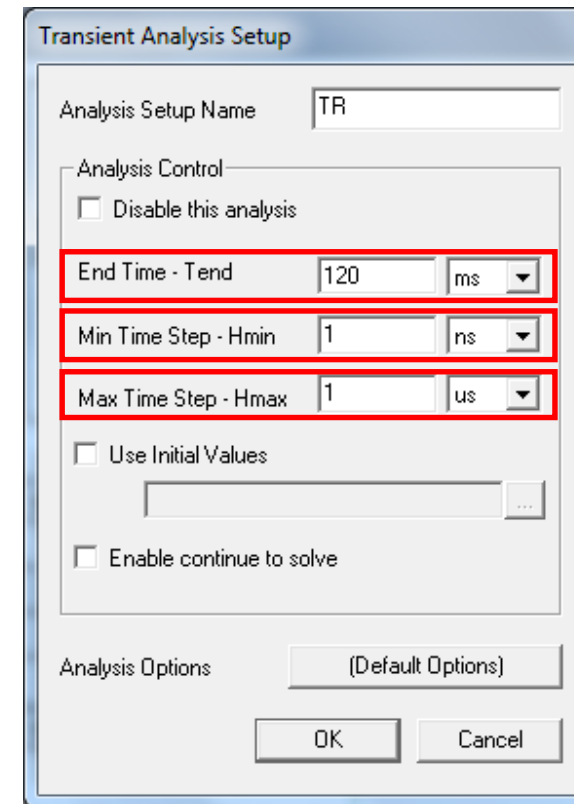
# Prepare the Postprocessing

- Select the menu item **Draw** → **Report** → **Rectangular Plot** and place the plot in the Schematic, for example above the Circuit
- Automatically the **New Trace** window pops-up
- Select the Current flowing through the DCMP Motor by selecting the quantity **DCMP1.IA**
- Click on the **Add Trace** button and then **Close**



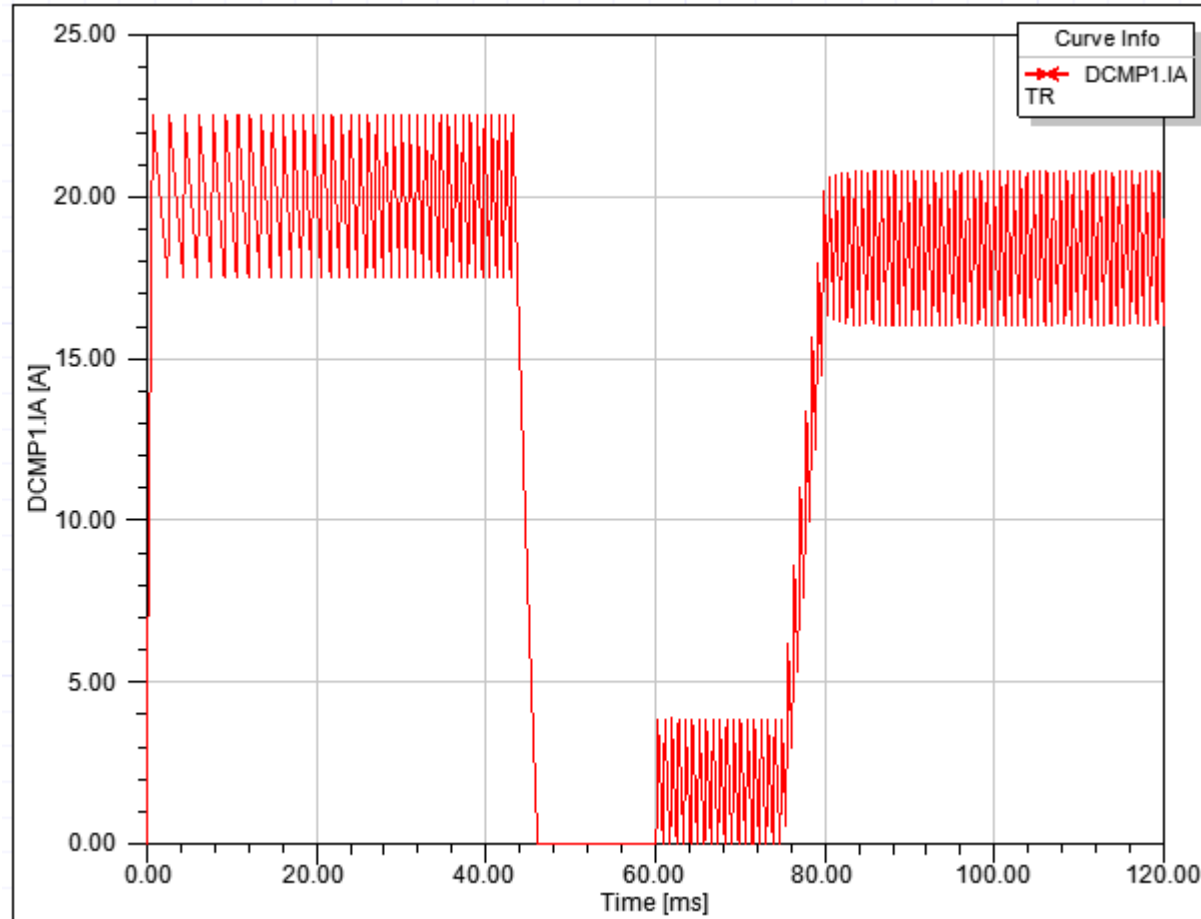
# Setup the Simulation Analysis

- In the present example is not possible to easily predict the Mosfet switching frequency due to the used control technique
- In order to be sure to represent in a proper way all the possible transients, in this case we set very small values for both Hmin and Hmax
- In the Transient Analysis Setup window:
  - Tend: 120 ms
  - Hmin: 1 ns
  - Hmax: 1  $\mu$ s
  - Press OK



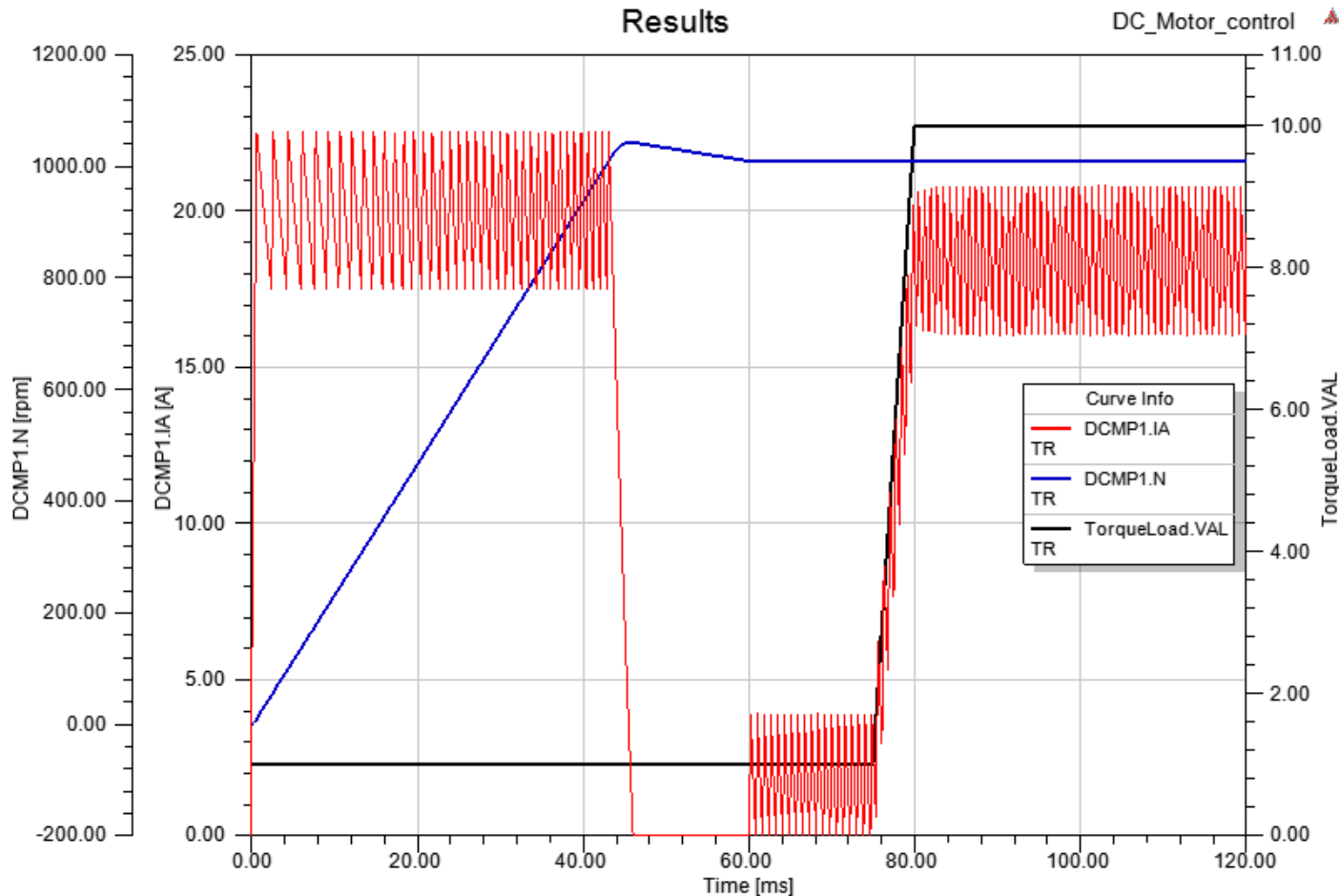
# Analyze and View Results

- Select the menu item *Simplorer Circuit* → *Analyze to run the Simulation*
- The final result for the current flowing through the DC Motor should look very similar to the following Figure:



# Further Results and Comments

- Select the Plot Report **RMB** → **Modify Report**
- Select the Quantities **DCMP1.N** and **TorqueLoad.VAL** and press **Add Trace**



- Note as the motor ramps up to the desired speed, the motor current is limited
- Once the motor reaches desired speed, the motor current is reduced to maintain speed
- When the torque load is increased again it causes the motor current to increase to support the increased torque load, maintaining the speed constant



# Saving the Project

- This completes the workshop
- Save the file with the name **WS\_2\_1** in the working folder