**Assignment 2**

Data have been collected from a laboratory scale system that operates in a similar way as a common hand held hair dryer. Air is blown through a tube after being heated at the inlet of the tube. The input to the process is the power applied to a mesh of resistor wires that constitutes the heating device. The measured variable is the air temperature at the outlet, measured in Volts by a thermocouple sensor. The experimental data consists of 1000 data points with a sampling interval of 80 ms. The data can be loaded from the MATLAB prompt by using the command:

>> load dryer2

This command loads the input vector ***u*2** and the output vector ***y*2**. You are required to:

1. Use the **iddata** command to create a suitable data object. Pre-process the data by removing the mean values of the variables using the function **dtrend**. Divide the data into an estimation data set and a validation data set, each containing 500 samples. Estimate an ARMAX model with the following structure:

****

where

****

Give the resulting polynomial coefficients. Validate the model by plotting the output predicted by the model and comparing it with the measured output for the validation data set. Also, compute and plot the auto-correlation of the residuals and the cross-correlation between the residuals and the input. Comment on the quality of the identified model.

1. Estimate another ARMAX model as given in Equation (1), but now with the following polynomial degrees:

****

Give the resulting polynomial coefficients. Validate the model by plotting the output predicted by the model and comparing it with the measured output for the validation data set. Also, compute and plot the auto-correlation of the residuals and the cross-correlation between the residuals and the input. How does this model compare with the one identified in (a)?

**Hint:** the MATLAB **armax** function can be used to estimate ARMAX models. It takes two arguments. The first argument is a data object obtained from using the command iddata. The second argument is a vector with the polynomial degrees and pure delay (*nk*): [*na*, *nb*, *nc*, *nk*].