1 About

This is a help manual for the Controller Design Tool, developed by *Davíð Örn Jóhanneson* [davidj11@ru.is] for Reykjavik University. The program is intended to be used as a teaching tool for modern system control theory.

2 Window view

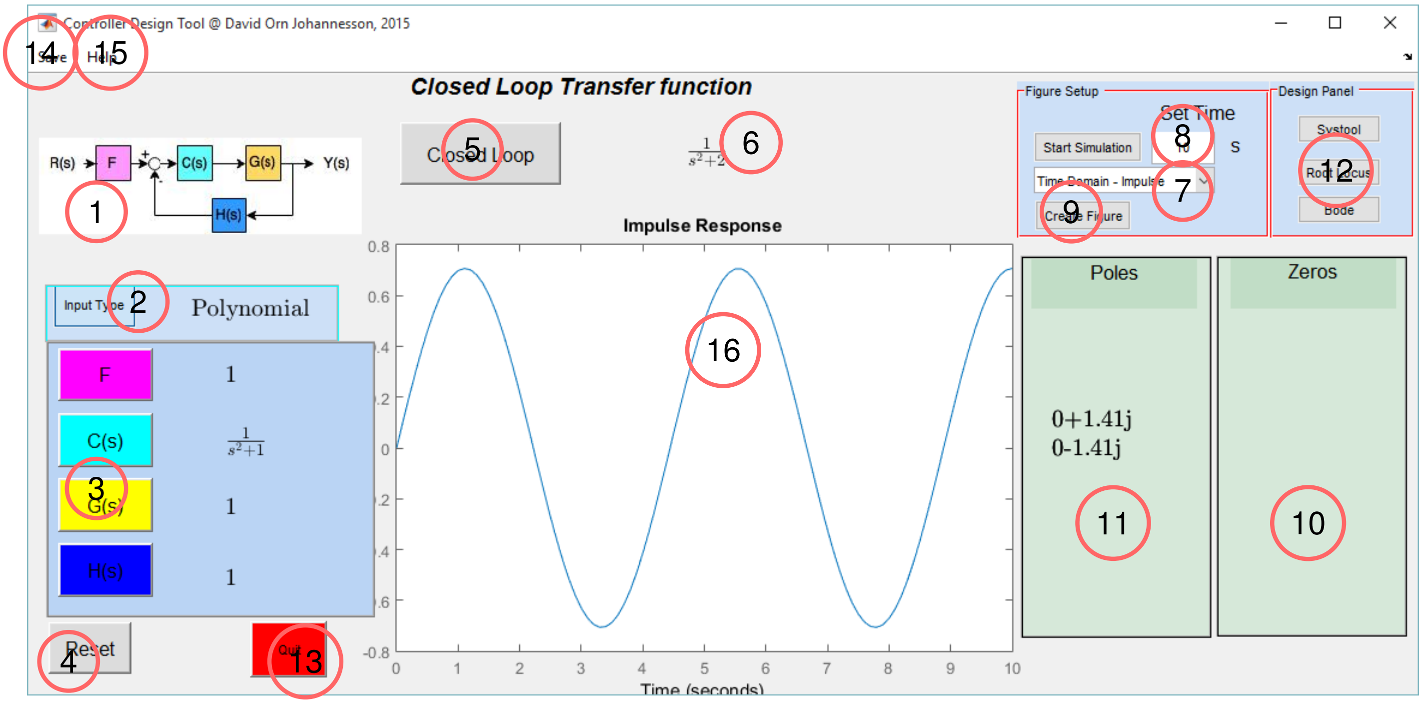

Figure 1: The layout for CDT

1. This is the layout of the system in question. It has four blocks. F is a multiplication factor, $C(s)$ is the controller, $G(c)$ is the plant to be controlled. $H(s)$ is the feedback block.

2. Here the input type can be controlled. Input types are *Polynomial* and *Pole/Zero* input. The *polynomial* input is typical to *Matlab*, for example

$s^3 + 2s^2 - 1 = [1, 2, 0, -1]$, etcetera. The *pole/zero* input is special to this program. The input should be the inverse signal of the pole/zero location. For example, given a pole $(s - 1)$ the input should be 1. If a sole integrator then the input should be 0, see Figure 2 for further detail.

3. By clicking these boxes a question dialogue opens up to enter the respective inputs to describe the system in question.

4. The reset button sets all blocks to one and clears the figure

5. Choose whether the system has feedback and is closed loop or if it should be open loop. Effectively turning off the feedback circuit or setting $H(s) = 0$.

6. Here the calculated transfer function is shown, the text shows whether the system is currently examining the closed or open loop system.

7. When changing the type of response to examine, changing popup-menu, in the figure set to *Time Domain - Impulse* the *Start Simulation* button must be pressed to receive an updated figure.

8. Here the time length of the time domain response can be changed, this is in seconds.

9. This creates a new figure outside the program to work on further.

10. The zeros of the transfer function are shown here

11. The poles of the transfer function are shown here

12. The buttons on the *Design Panel* open the *sisotool* tool from matlab. Type `help sistool` in the matlab command window for further reading.

13. The quit button closes the application and clears the workspace.

14. The save button allows the user to save the current system to a `.mat` file in the workspace.

15. The help button opens this document.
16. The main axis shows the figure defined in the popup-menu (8). It gets updated whenever the blocks change but not when the popup-menu is changed, for that press the *Start Simulation* button. The user can also right press the figure for additional options.

3 Input Dialogue

When the block buttons (3)are pressed an question dialogue opens up. In this box the values for the system are chosen. The values depend on weather the input is polynomial or pole/zero input. Each block button holds its own transfer function.
Figure 2 shows inputs for the resulting output in Figure 1. Note that the boxes both give the same results.

3.1 Polynomial input

The left box in figure 2 shows the question dialogue. There are three inputs

$$f(\text{k,num, den}) = \text{k} \cdot \frac{\text{num}}{\text{den}}$$

Where the values num and den are the coefficients for the polynomials of the numerator and denominator

$$\text{den} = [1,\ 0,\ 1] = s^2 + 1$$

3.2 Pole/Zero input

The question dialogue that opens appears similar note that the box asks for K, Zeros and Poles. The K is still a multiplication factor and here the poles and zeros are entered as they appear in the S plane

$$f(\text{K}, \text{Zeros}, \text{Poles}) = K \frac{\prod_k (s + z_k)}{\prod_n (s + p_n)}$$

The example as in the left half of figure 2 is as follows

$$\text{Poles} = [-i,\ i] = (s+i)(s-i) = s^2 + 1$$

When working with imaginary numbers the program will return an error if all imaginary numbers don't have conjugates.

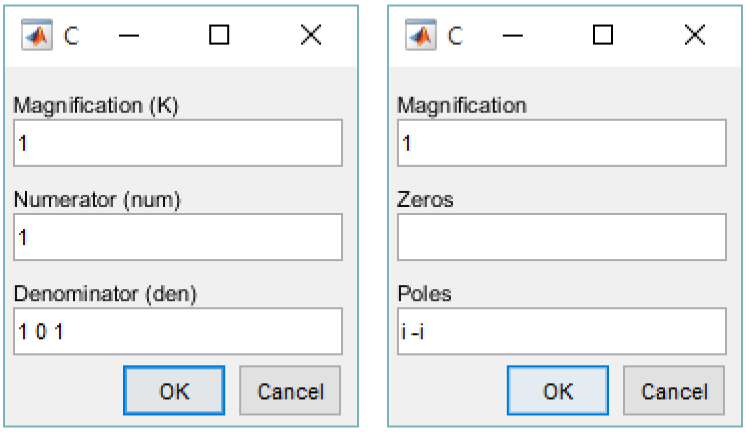

Figure 2: Question dialogue boxes, polynomial input and pole-zero input respectively