

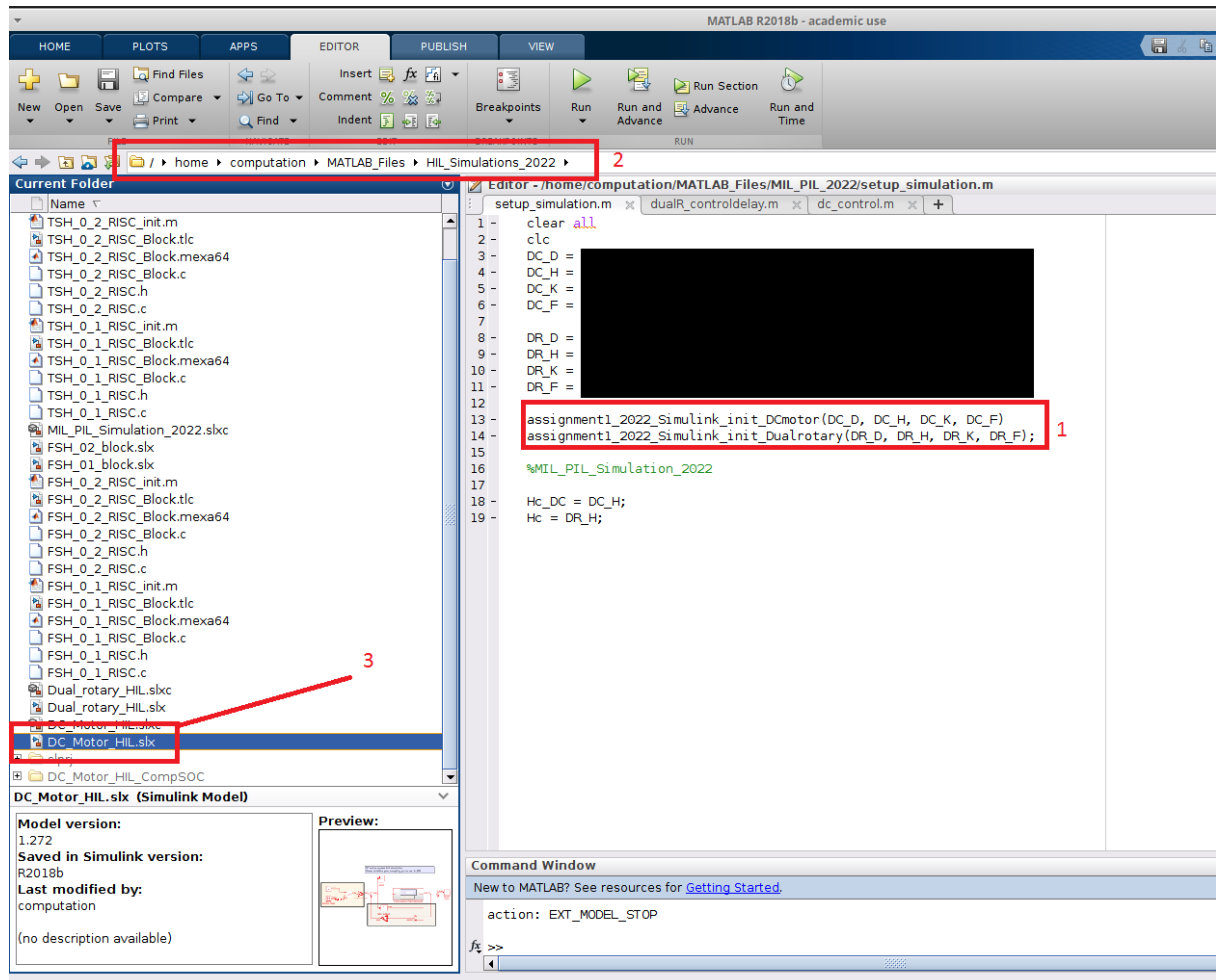
Prepare the initial variables (1 in the image below) :

assignment1_2022_Simulink_init_DCmotor(DC_D, DC_H, DC_K, DC_F);

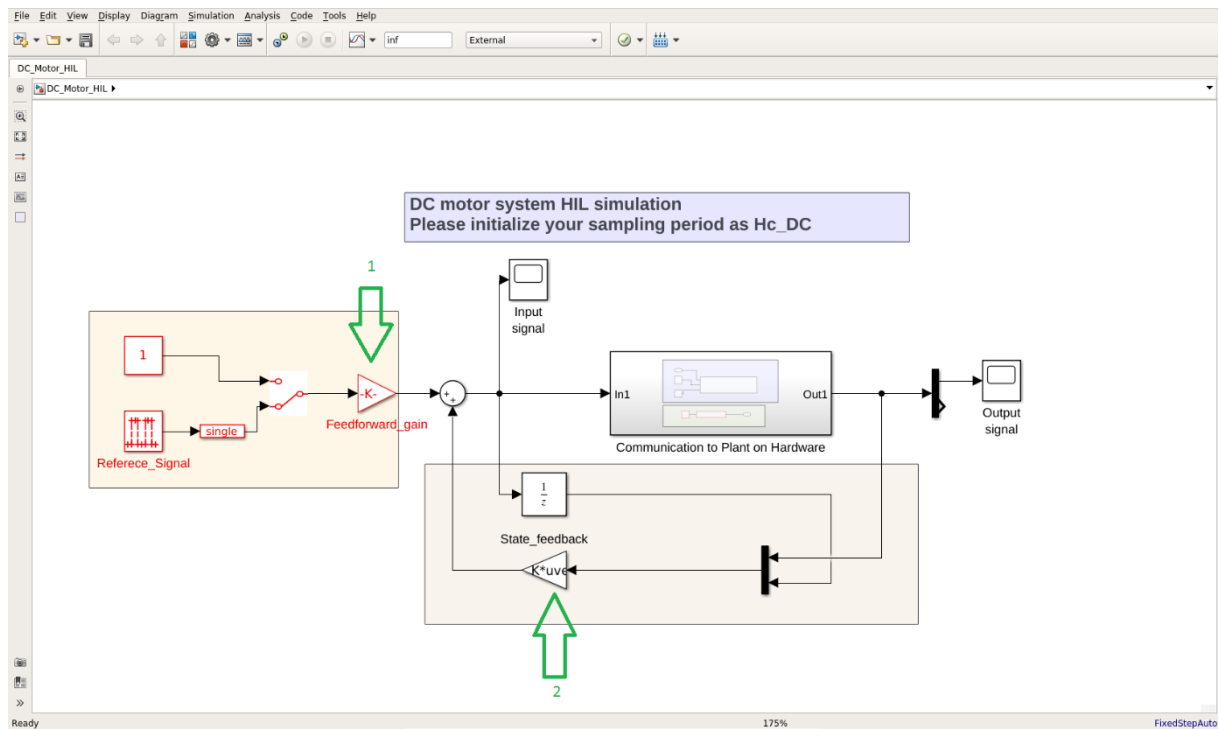
assignment1_2022_Simulink_init_Dualrotary(DR_D, DR_H, DR_K, DR_F);

Navigate to the HIL folder (2)

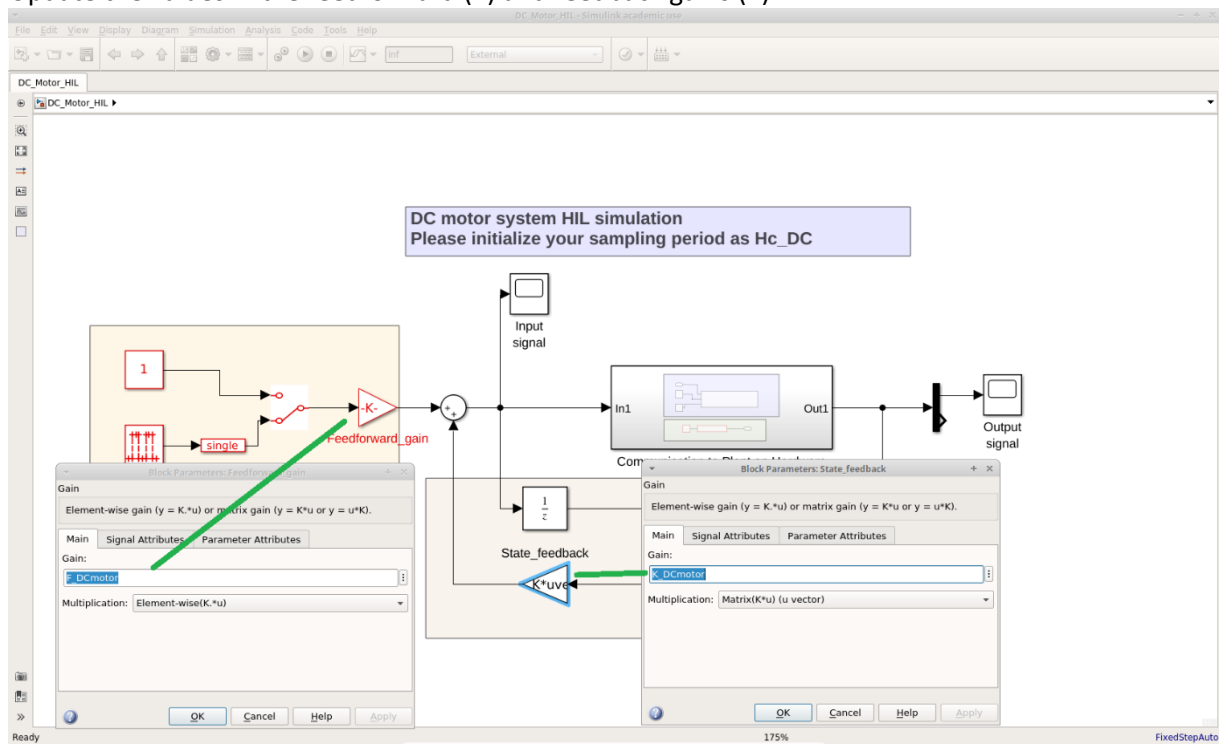
Double click the DC_Motor_HIL.slx (3)



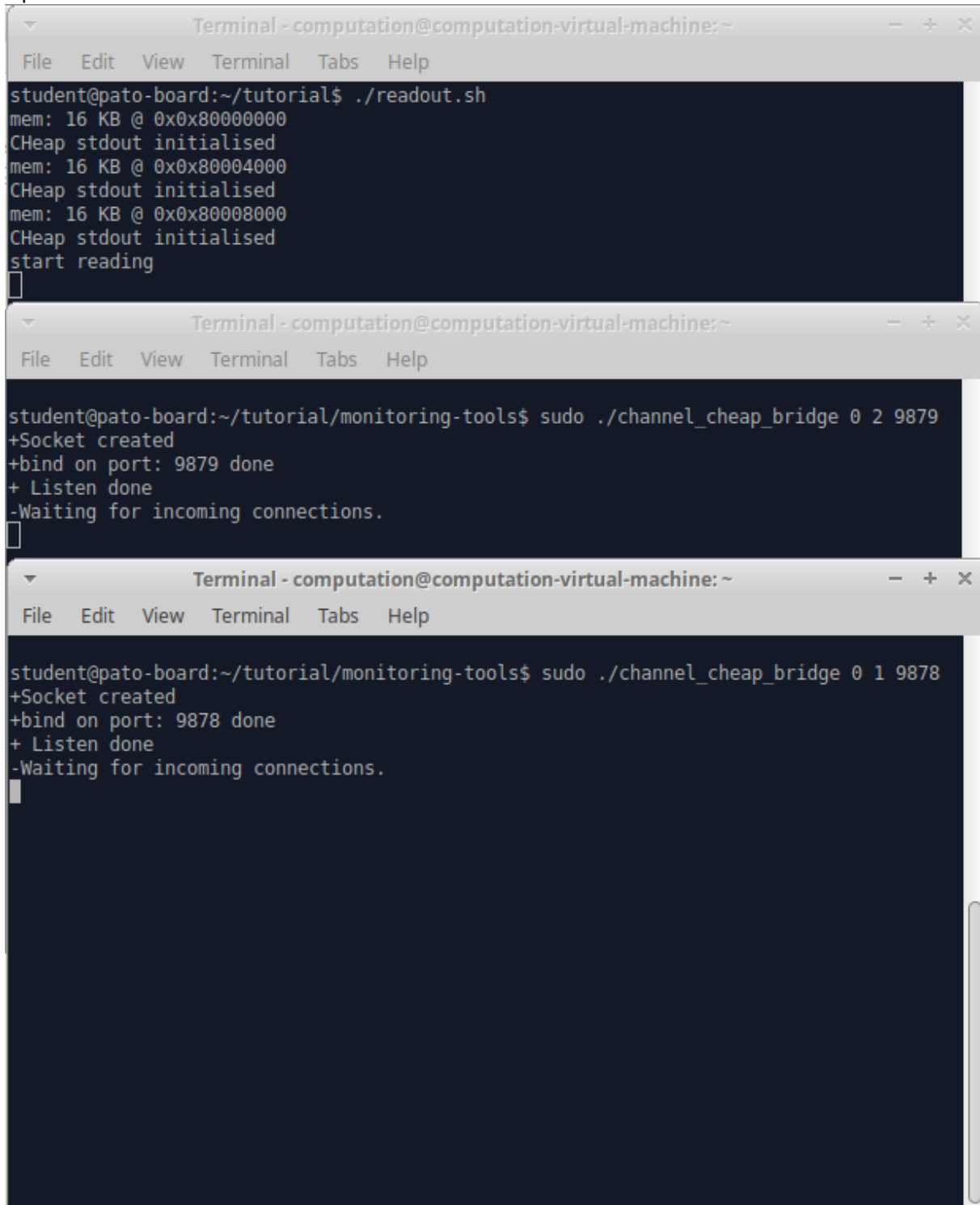
This gives the initial screen:



Update the values in the feedforward (1) and feedback gains (2)



Open 3 terminals to connect to the FPGA and execute one of the commands in each window



```
Terminal - computation@computation-virtual-machine: ~
File Edit View Terminal Tabs Help

student@pato-board:~/tutorial$ ./readout.sh
mem: 16 KB @ 0x0x80000000
CHep stdout initialised
mem: 16 KB @ 0x0x80004000
CHep stdout initialised
mem: 16 KB @ 0x0x80008000
CHep stdout initialised
start reading
█

Terminal - computation@computation-virtual-machine: ~
File Edit View Terminal Tabs Help

student@pato-board:~/tutorial/monitoring-tools$ sudo ./channel_cheap_bridge 0 2 9879
+Socket created
+bind on port: 9879 done
+ Listen done
-Waiting for incoming connections.
█

Terminal - computation@computation-virtual-machine: ~
File Edit View Terminal Tabs Help

student@pato-board:~/tutorial/monitoring-tools$ sudo ./channel_cheap_bridge 0 1 9878
+Socket created
+bind on port: 9878 done
+ Listen done
-Waiting for incoming connections.
█
```

To be sure no problems occur, reset stuff on the fpga with

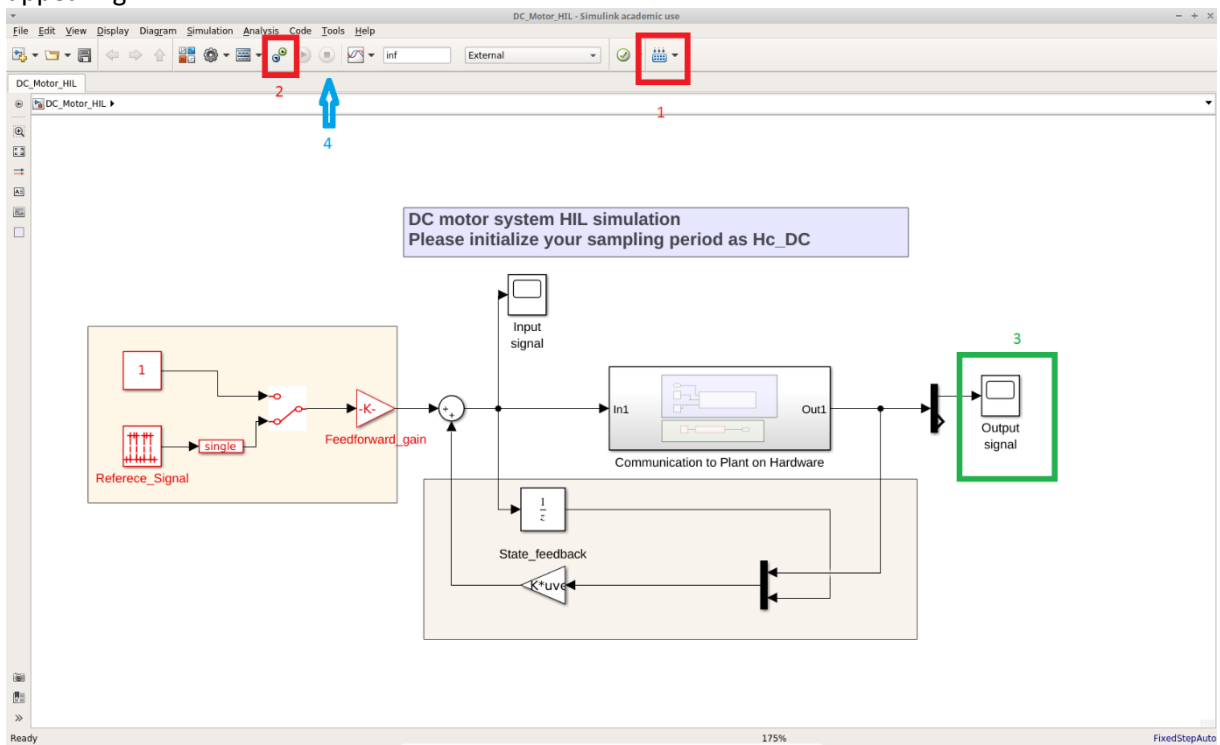
Go to the tutorial folder

`./rerun.sh`

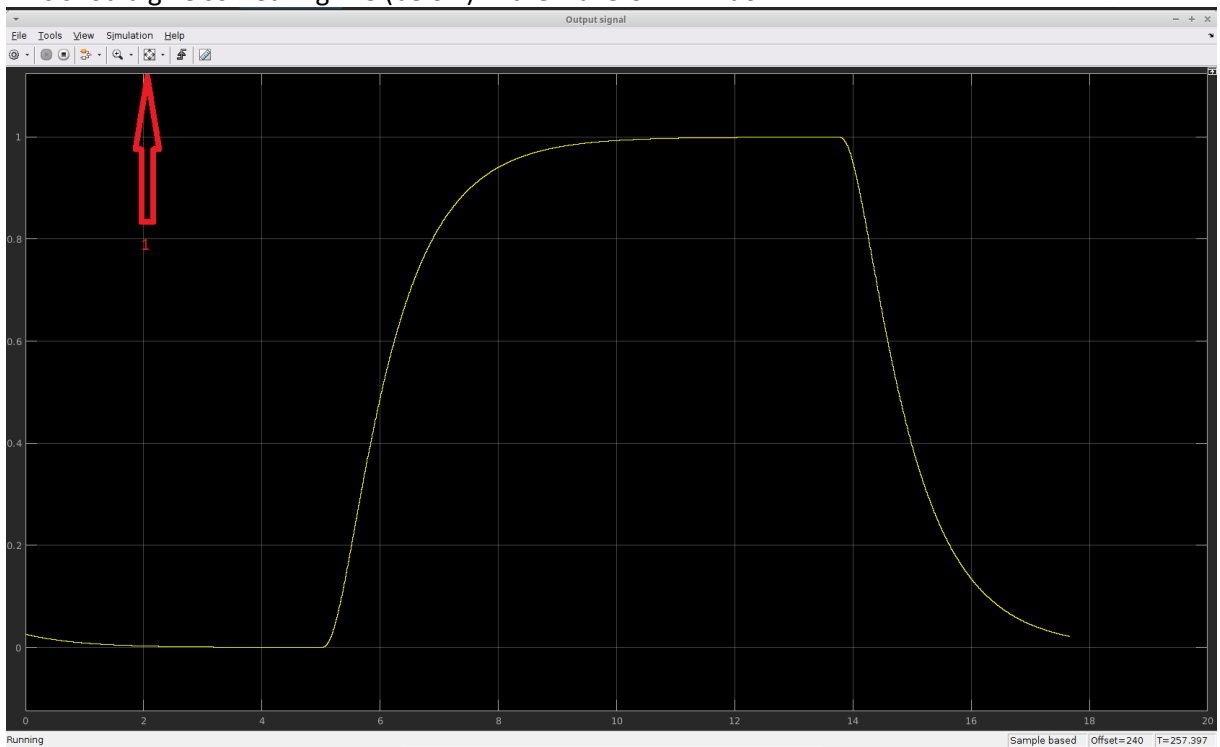
Screenshot

```
student@pato-board:~/tutorial$ ./rerun.sh █
```

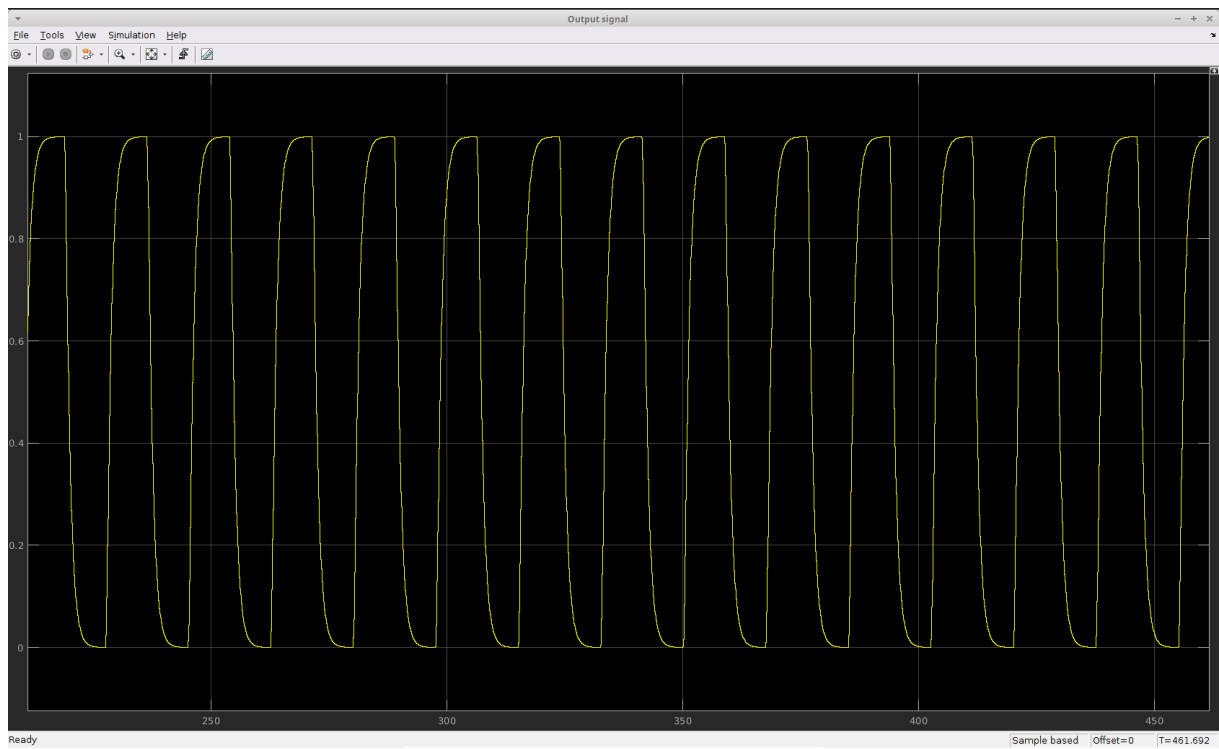
Build the model (1) and then connect to the target (2). After some time stop the simulation by pressing (4). Afterwards, double click the output signal block (3). You should see some waveform appearing.



This should give something like (below) in the waveform window:



Pressing the button (1) above will fit the waveform to the screen:



Note: the input waveform is a square wave.