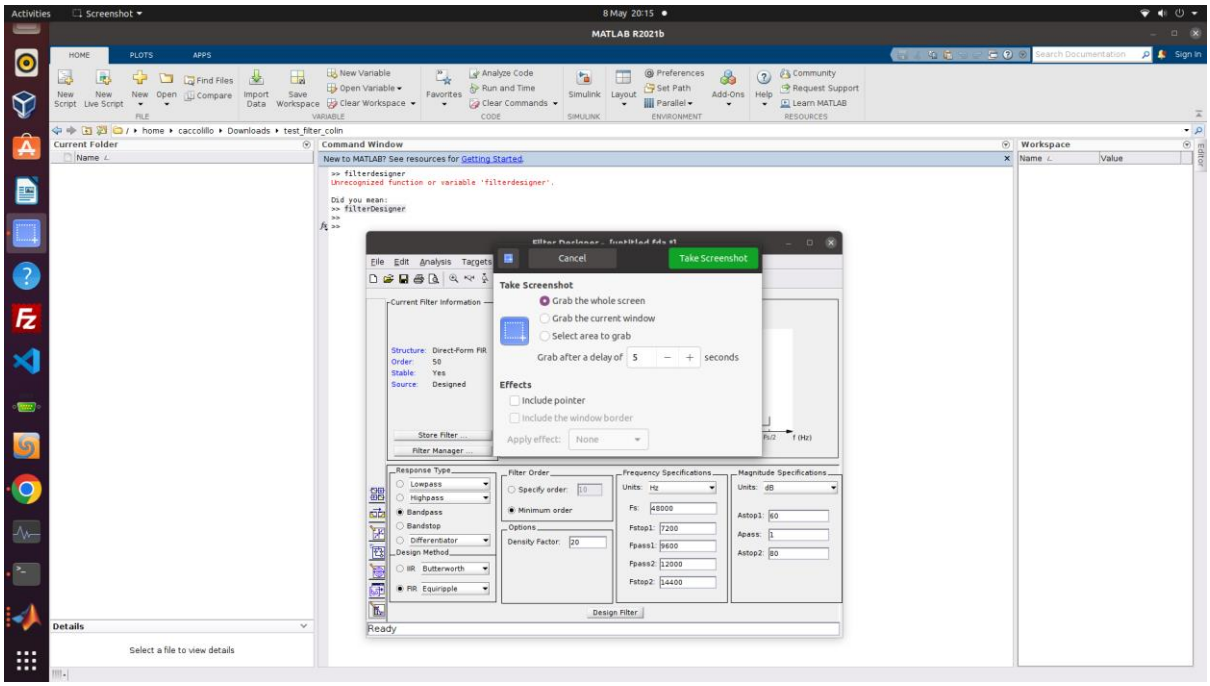
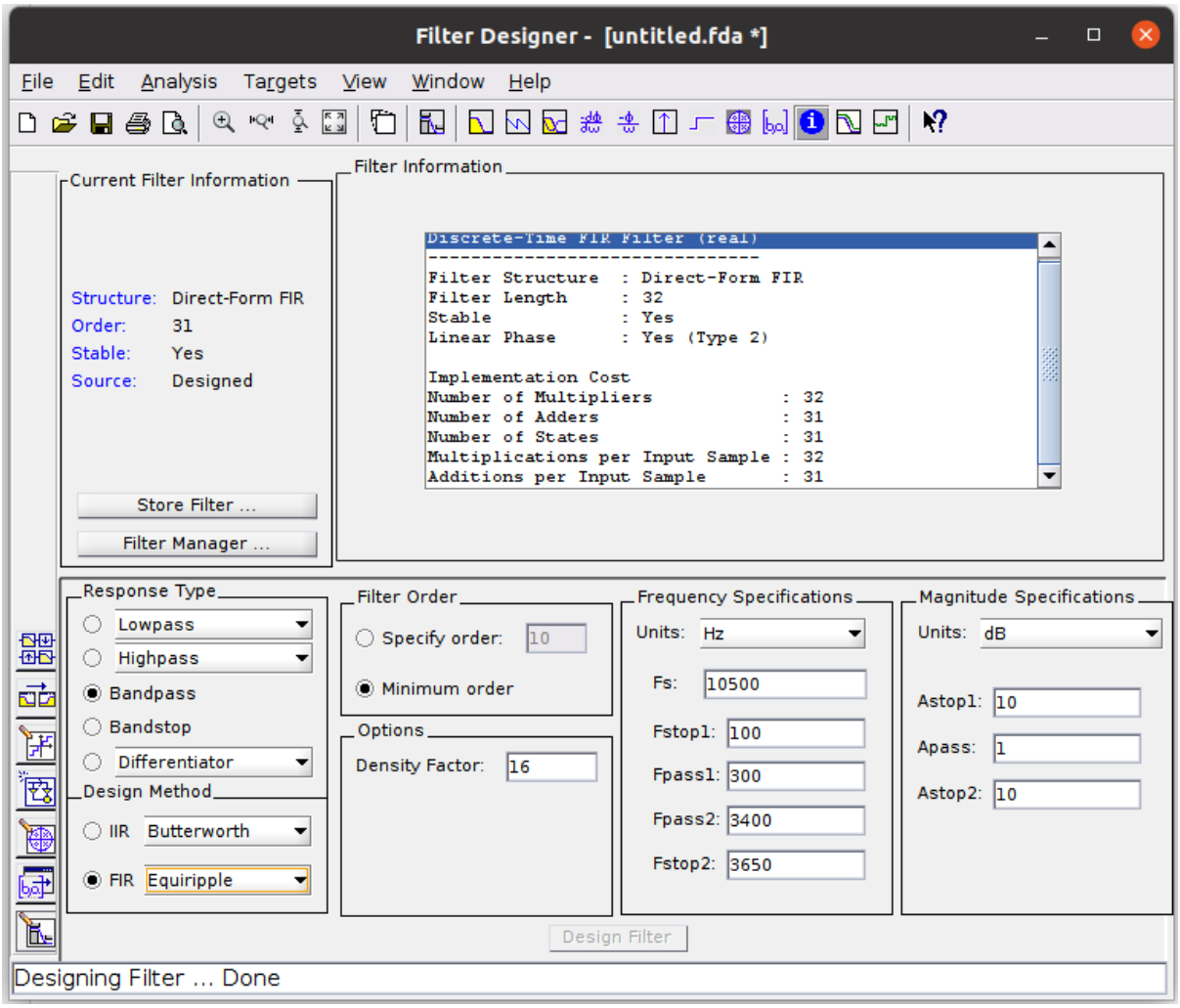
Opening the GUI

Start Matlab and type “filterDesigner”, the filter designer GUI shows up:



Set the desired parameters:



And press “design filter” to get the coefficients.

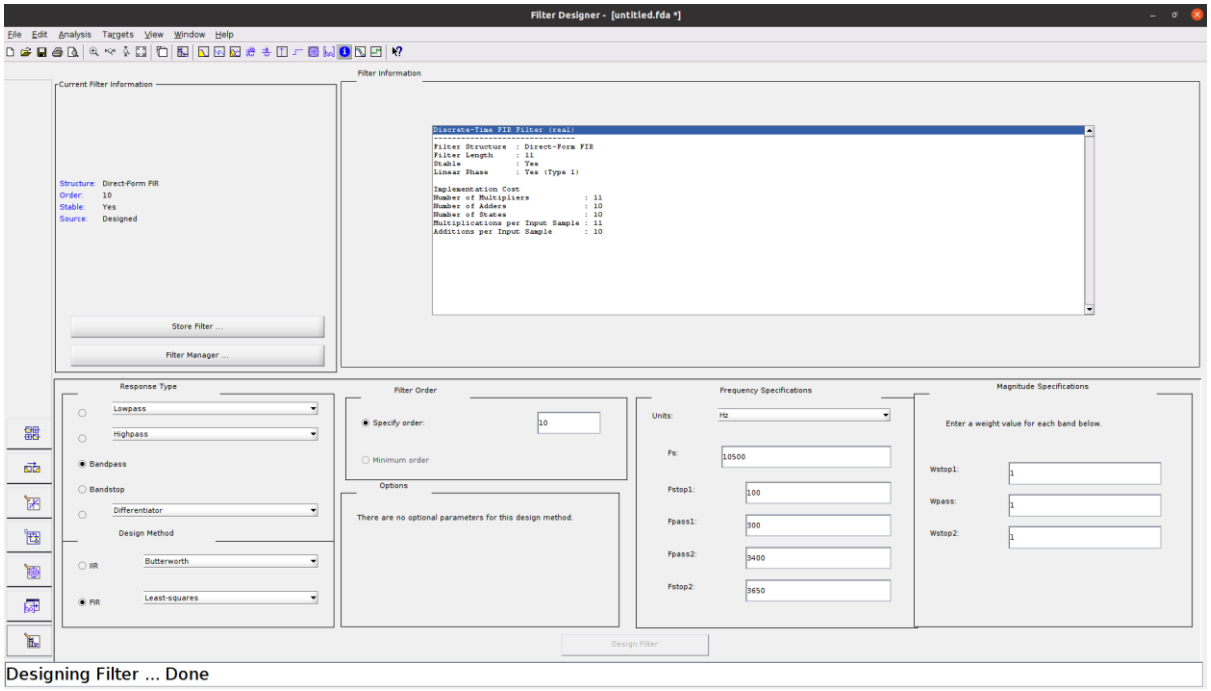
Once done, in file->generate matlab code, matlab scripts describing the filter can be used:

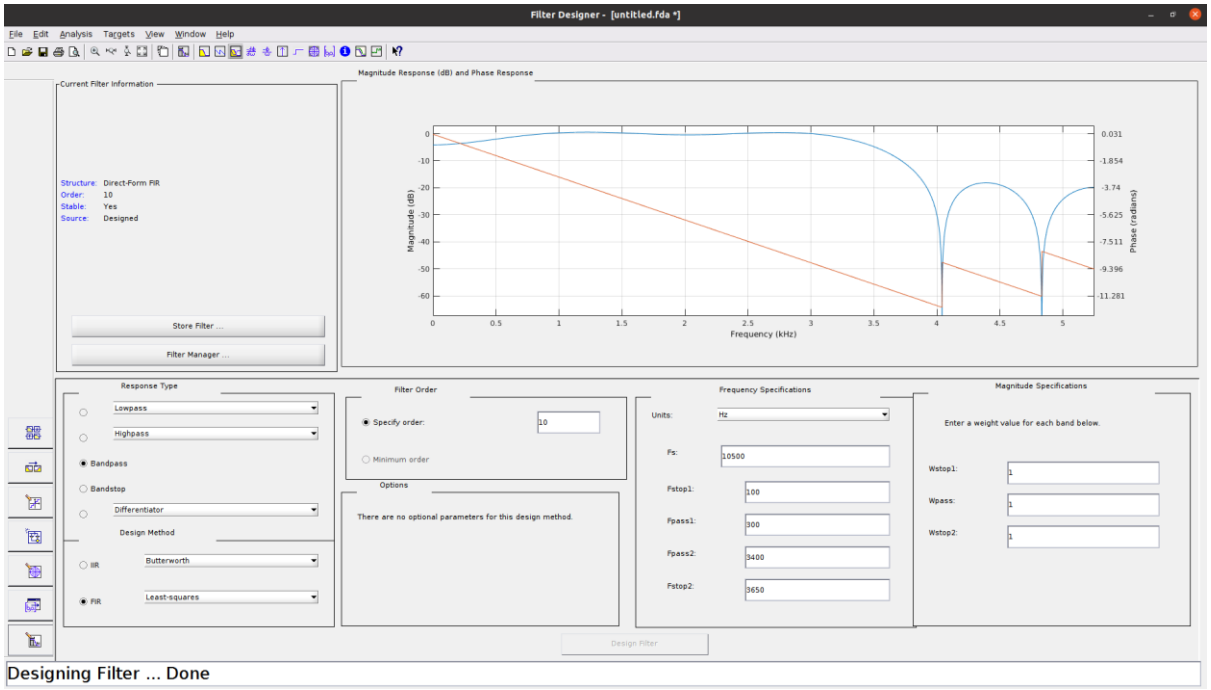
Editor - /home/caccolillo/Downloads/test_filter_colin/design_filter.m

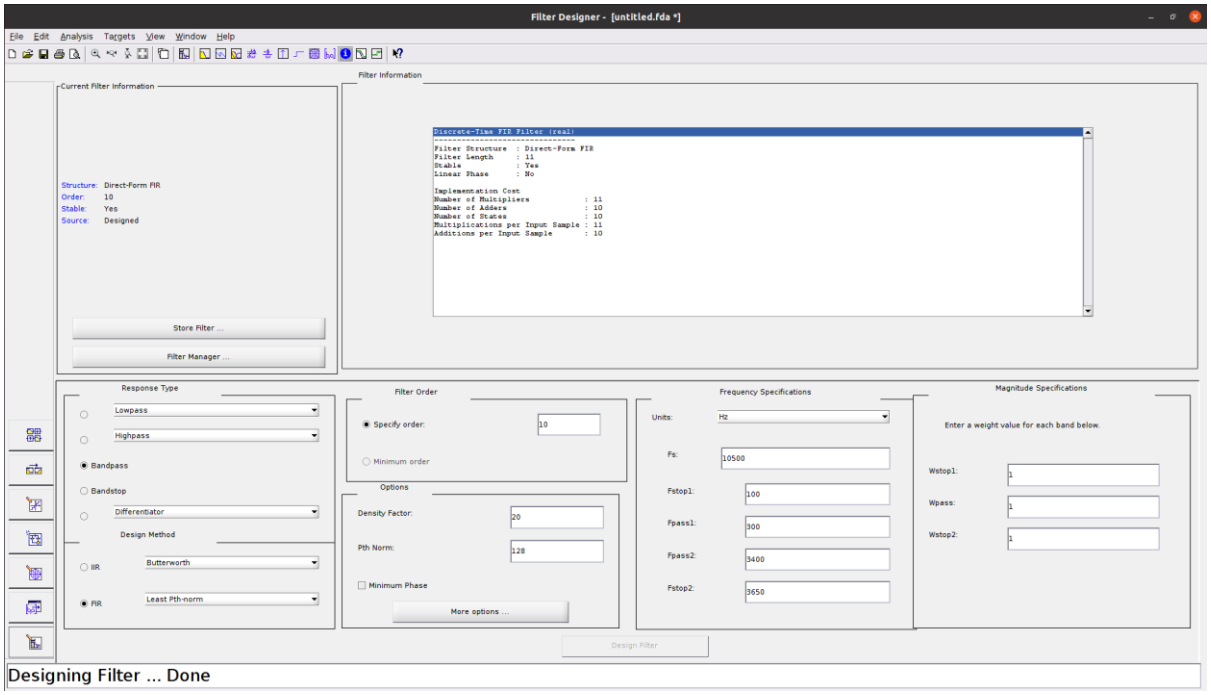
design_filter.m

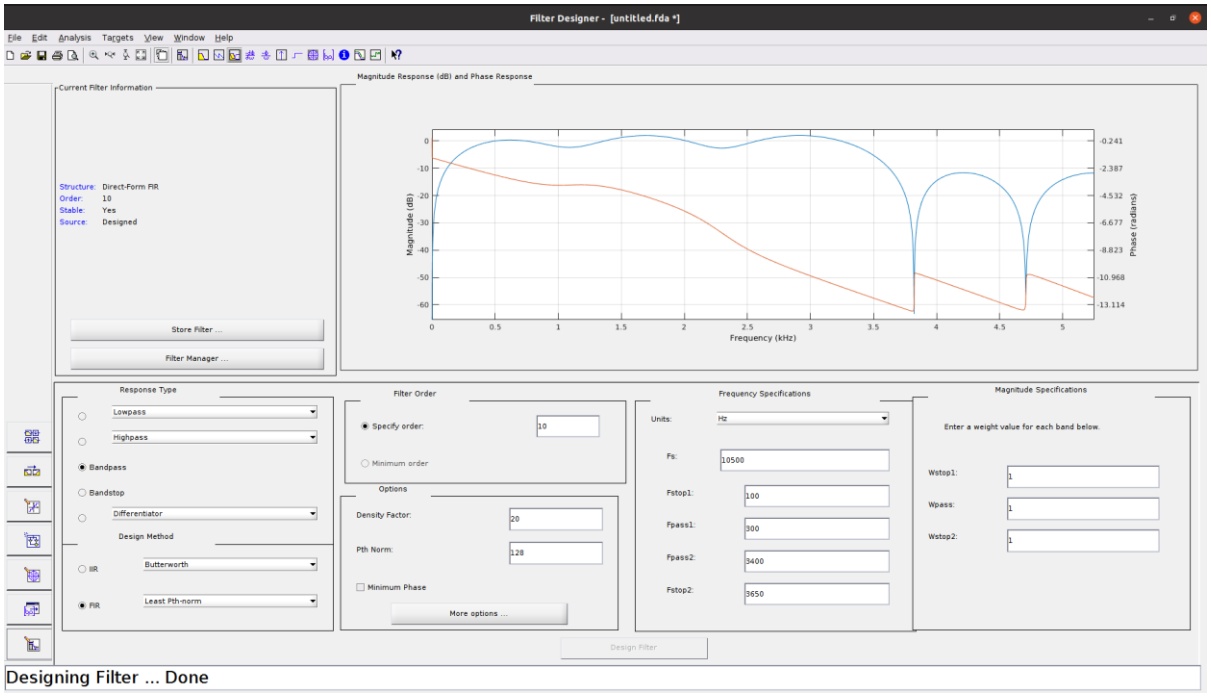
```
function Hd = design_filter
%DESIGN_FILTER Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.11 and DSP System Toolbox 9.13.
% Generated on: 08-May-2025 20:23:37

% Equiripple Bandpass filter designed using the FIRPM function.

% All frequency values are in Hz.
Fs = 10500;  % Sampling Frequency

Fstop1 = 100;              % First Stopband Frequency
Fpass1 = 300;              % First Passband Frequency
Fpass2 = 3400;             % Second Passband Frequency
Fstop2 = 3650;             % Second Stopband Frequency
Dstop1 = 0.31622776602;    % First Stopband Attenuation
Dpass  = 0.057501127785;   % Passband Ripple
Dstop2 = 0.31622776602;    % Second Stopband Attenuation
dens   = 16;               % Density Factor

% Calculate the order from the parameters using FIRPMORD.
[N, Fo, Ao, W] = firpmord([Fstop1 Fpass1 Fpass2 Fstop2]/(Fs/2), [0 1 ...
                          0], [Dstop1 Dpass Dstop2]);

% Calculate the coefficients using the FIRPM function.
b  = firpm(N, Fo, Ao, W, {dens});
Hd = dfilt.dffir(b);

% [EOF]
```

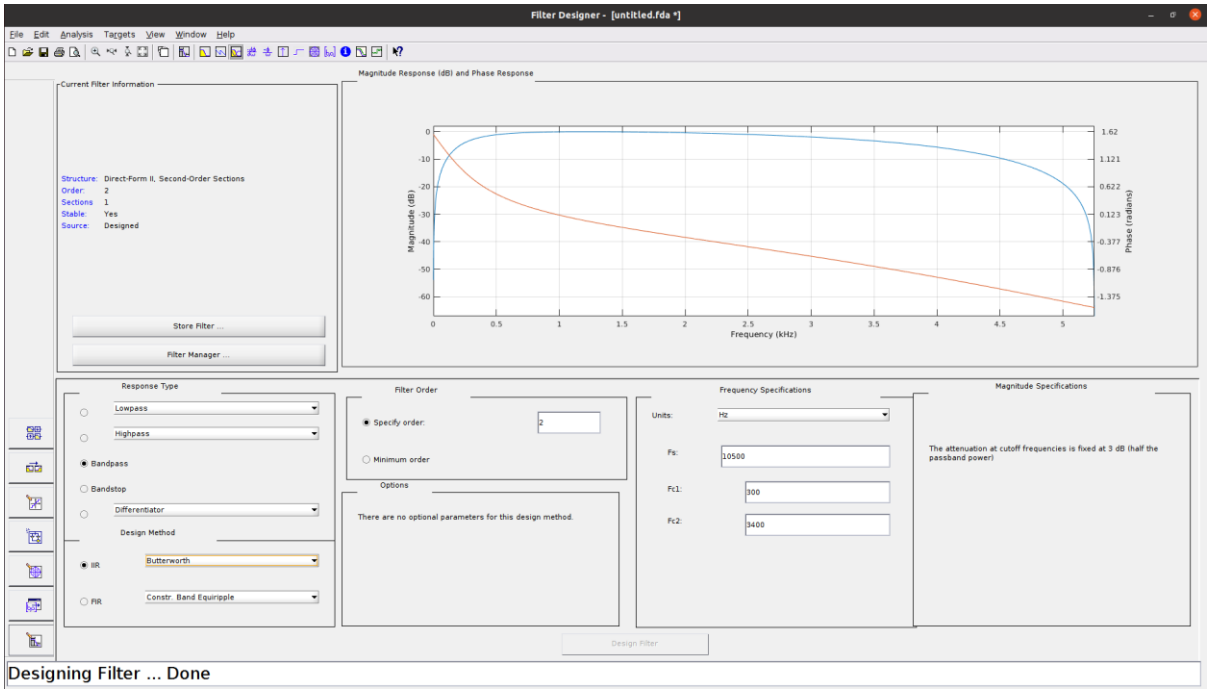
Changing filter topology, we can get a lower HW complexity:

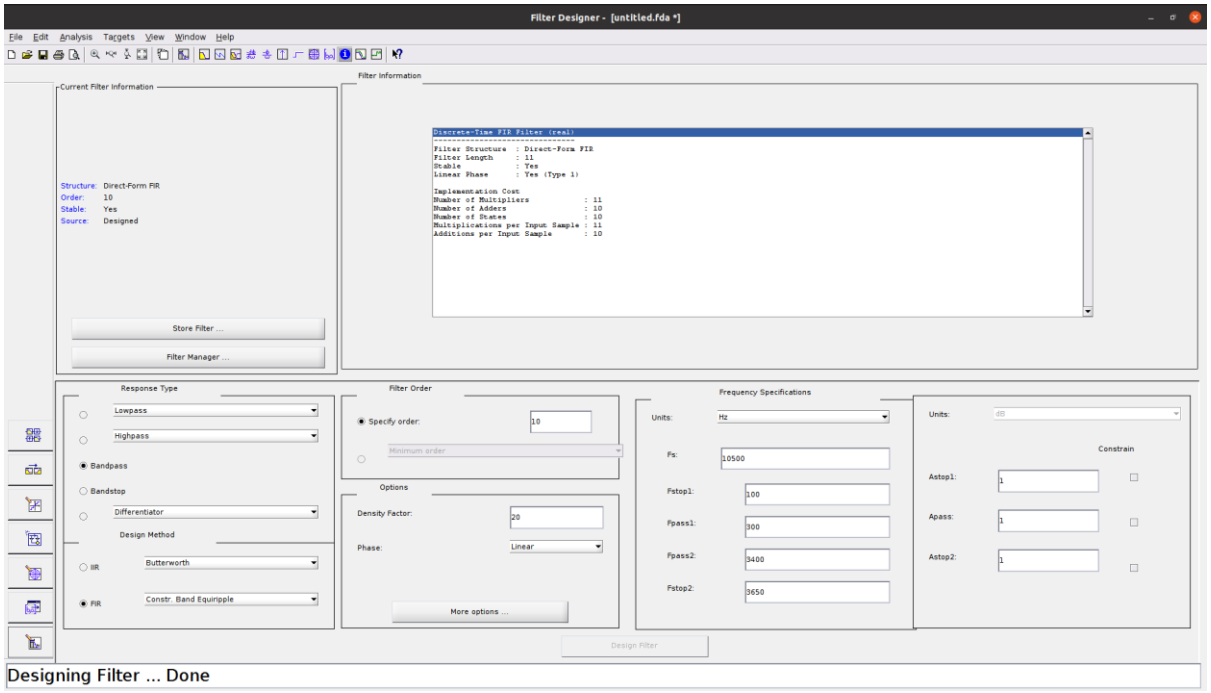

Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 10
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Filter Information
Discrete-Time FIR Filter (real)
Filter Structure : Direct-Form FIR
Filter Length : 11
Stable : Yes
Linear Phase : Yes (Type 1)
Implementation Cost
Number of Multipliers : 11
Number of Adders : 10
Number of States : 10
Multiplications per Input Sample : 11
Additions per Input Sample : 10
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR
Butterworth
FIR
Least-squares
Filter Order
Specify order:
10
Minimum order
Options
There are no optional parameters for this design method.
Frequency Specifications
Units:
Hz
Fs:
10500
Fstop1:
100
Fpass1:
300
Fpass2:
3400
Fstop2:
3650
Magnitude Specifications
Enter a weight value for each band below.
Wstop1:
1
Wpass:
1
Wstop2:
1
Design Filter
Designing Filter ... Done

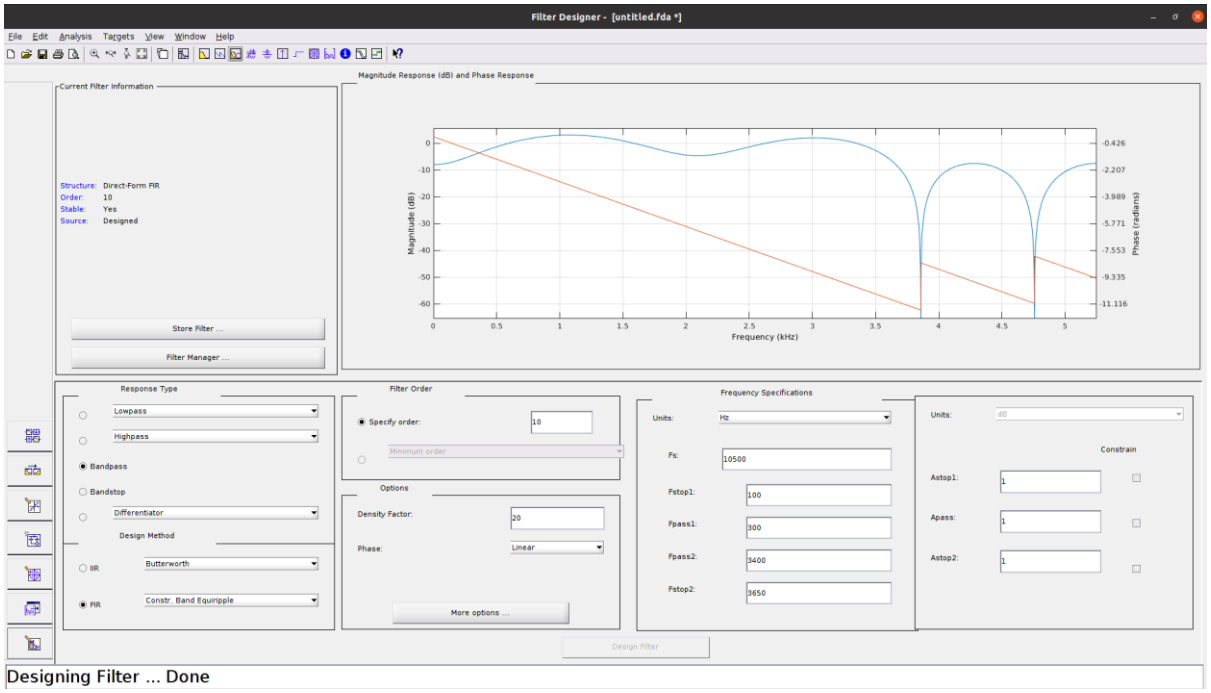

Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 10
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Magnitude Response (dB) and Phase Response
Magnitude (dB)
Phase (radians)
Frequency (kHz)
0
-10
-20
-30
-40
-50
-60
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
0.031
-1.854
-3.74
-5.625
-7.511
-9.396
-11.281
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR
Butterworth
FIR
Least-squares
Filter Order
Specify order:
10
Minimum order
Options
There are no optional parameters for this design method.
Frequency Specifications
Units:
Hz
Fs:
10500
Fstop1:
100
Fpass1:
300
Fpass2:
3400
Fstop2:
3650
Magnitude Specifications
Enter a weight value for each band below.
Wstop1:
1
Wpass:
1
Wstop2:
1
Design Filter
Designing Filter ... Done


Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 10
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Filter Information
Discrete-Time FIR Filter (real)
Filter Structure : Direct-Form FIR
Filter Length : 11
Stable : Yes
Linear Phase : No
Implementation Cost
Number of Multipliers : 11
Number of Adders : 10
Number of States : 10
Multiplications per Input Sample : 11
Additions per Input Sample : 10
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR
Butterworth
FIR
Least Pth-norm
Filter Order
Specify order:
10
Minimum order
Options
Density Factor:
20
Pth Norm:
128
Minimum Phase
More options ...
Frequency Specifications
Units:
Hz
Fs:
10500
Fstop1:
100
Fpass1:
300
Fpass2:
3400
Fstop2:
3650
Magnitude Specifications
Enter a weight value for each band below.
Wstop1:
1
Wpass:
1
Wstop2:
1
Design Filter
Designing Filter ... Done

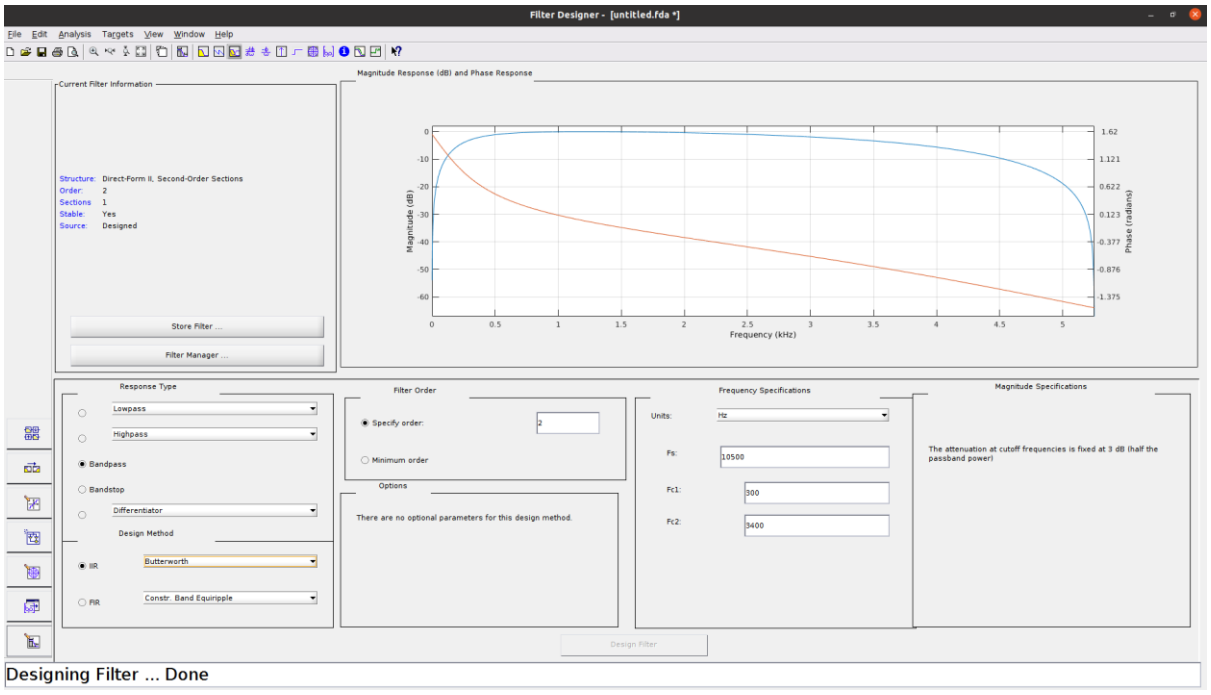

Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 10
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Magnitude Response (dB) and Phase Response
Magnitude (dB)
Phase (radians)
Frequency (kHz)
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR
Butterworth
FIR
Least Pth-norm
Filter Order
Specify order:
10
Minimum order
Options
Density Factor:
20
Pth Norm:
128
Minimum Phase
More options ...
Frequency Specifications
Units:
Hz
Fs:
10500
Fstop1:
100
Fpass1:
300
Fpass2:
3400
Fstop2:
3650
Magnitude Specifications
Enter a weight value for each band below.
Wstop1:
1
Wpass:
1
Wstop2:
1
Design Filter
Designing Filter ... Done

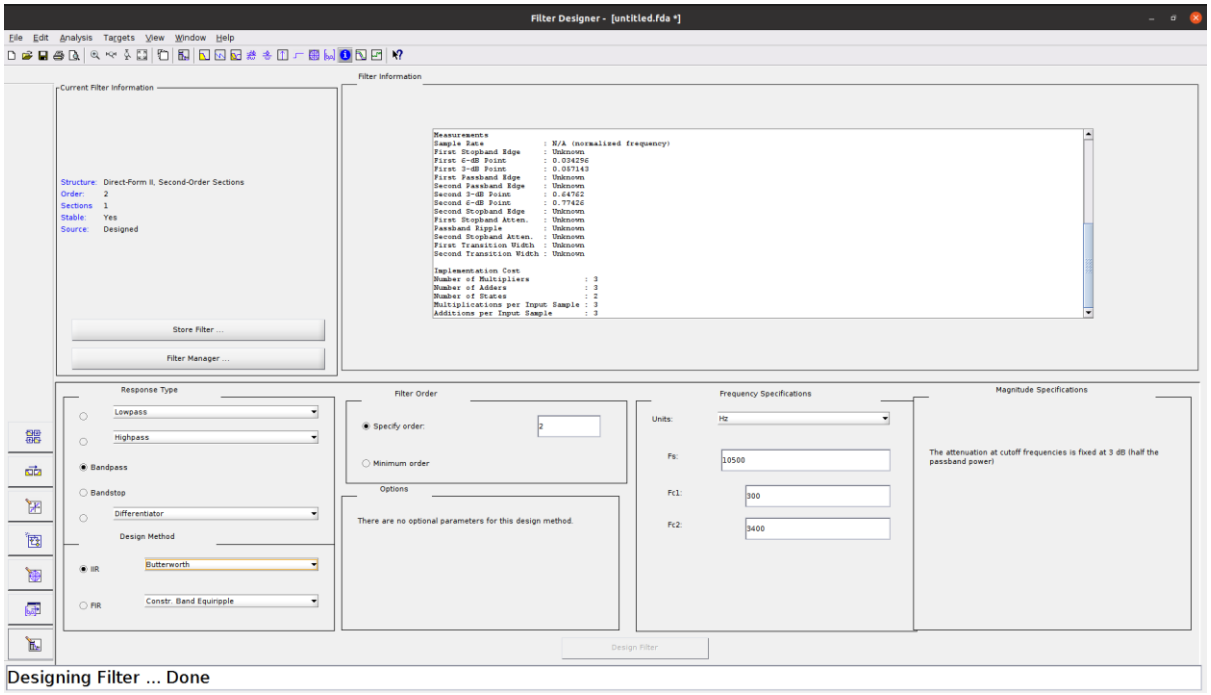

Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form II, Second-Order Sections
Order: 2
Sections: 1
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Magnitude Response (dB) and Phase Response
Magnitude (dB)
Phase (radians)
Frequency (kHz)
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR
Butterworth
FIR
Constr. Band Equiripple
Filter Order
Specify order:
2
Minimum order
Options
There are no optional parameters for this design method.
Frequency Specifications
Units:
Hz
Fs:
10500
Fc1:
300
Fc2:
3400
Magnitude Specifications
The attenuation at cutoff frequencies is fixed at 3 dB (half the passband power)
Design Filter
Designing Filter ... Done


Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 10
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Filter Information
Discrete-Time FIR Filter (real)
Filter Structure : Direct-Form FIR
Filter Length : 11
Stable : Yes
Linear Phase : Yes (Type 1)
Implementation Cost
Number of Multipliers : 11
Number of Adders : 10
Number of States : 10
Multiplications per Input Sample : 11
Additions per Input Sample : 10
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR
Butterworth
FIR
Constr. Band Equiripple
Filter Order
Specify order:
10
Minimum order
Options
Density Factor:
20
Phase:
Linear
More options ...
Frequency Specifications
Units:
Hz
Fs:
10500
Fstop1:
100
Fpass1:
300
Fpass2:
3400
Fstop2:
3650
Units:
dB
Constrain
Astop1:
1
Apass:
1
Astop2:
1
Design Filter
Designing Filter ... Done



By using an IIR:



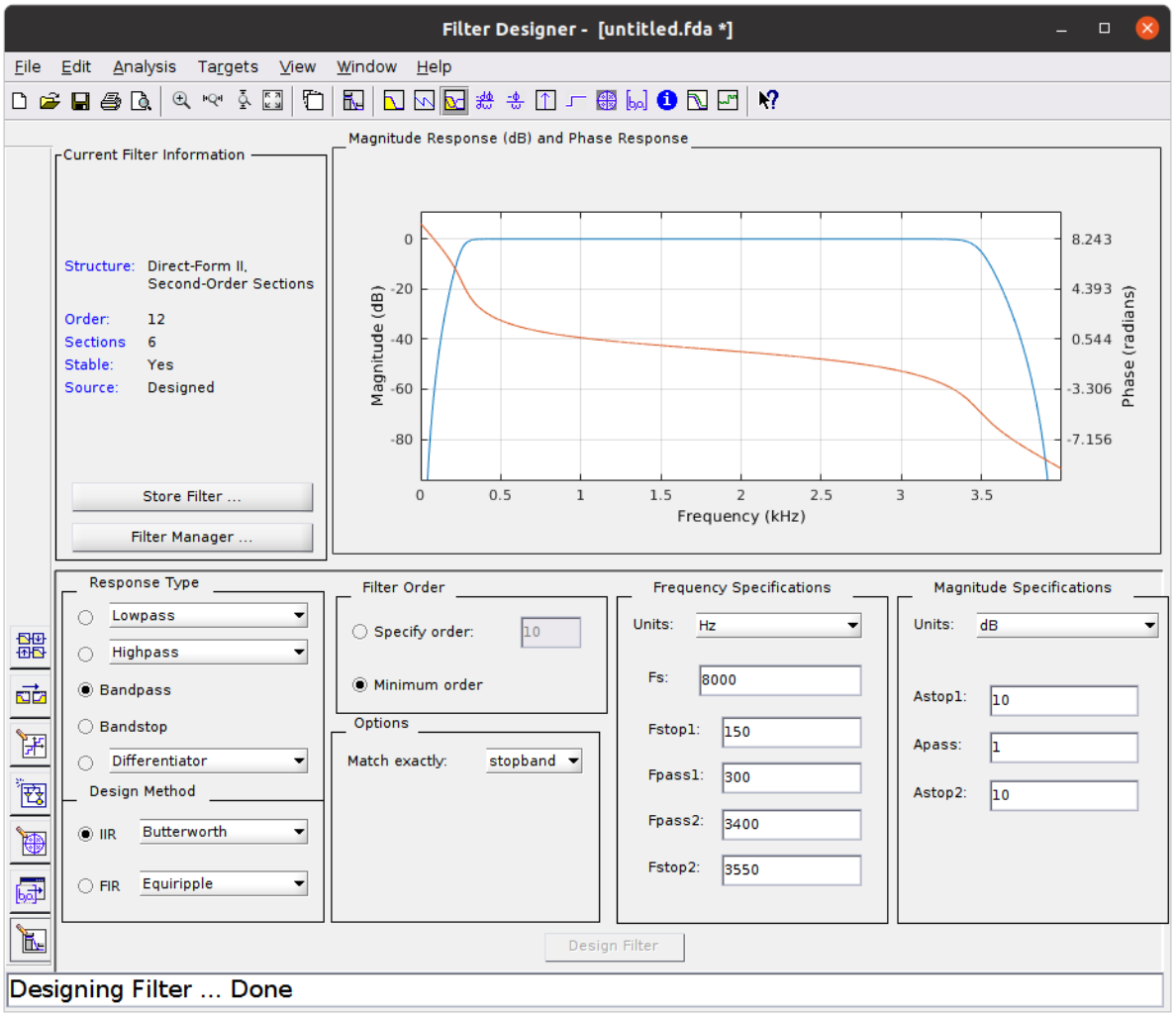


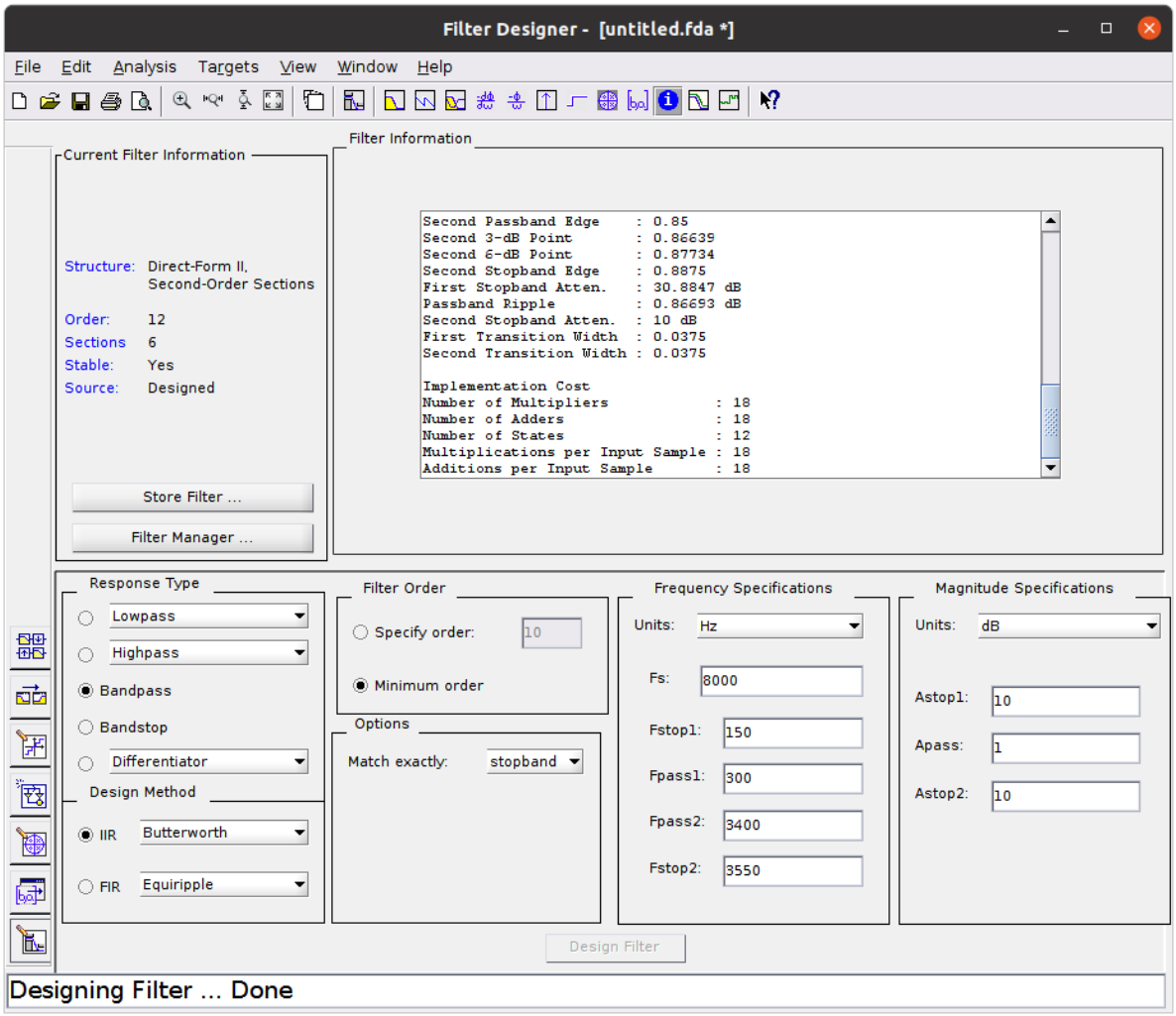
Designing a filter

Instructions are available at:

https://uk.mathworks.com/help/hdlfilter/basic-fir-filter.html

Let’s go for the following filter then:


Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form II, Second-Order Sections
Order: 12
Sections 6
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Magnitude Response (dB) and Phase Response
Magnitude (dB)
Phase (radians)
Frequency (kHz)
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR Butterworth
FIR Equiripple
Filter Order
Specify order: 10
Minimum order
Options
Match exactly: stopband
Frequency Specifications
Units: Hz
Fs: 8000
Fstop1: 150
Fpass1: 300
Fpass2: 3400
Fstop2: 3550
Magnitude Specifications
Units: dB
Astop1: 10
Apass: 1
Astop2: 10
Design Filter
Designing Filter ... Done



then under file -> generate MATLAB code -> filter design function, we can get the following script to replicate the filter:

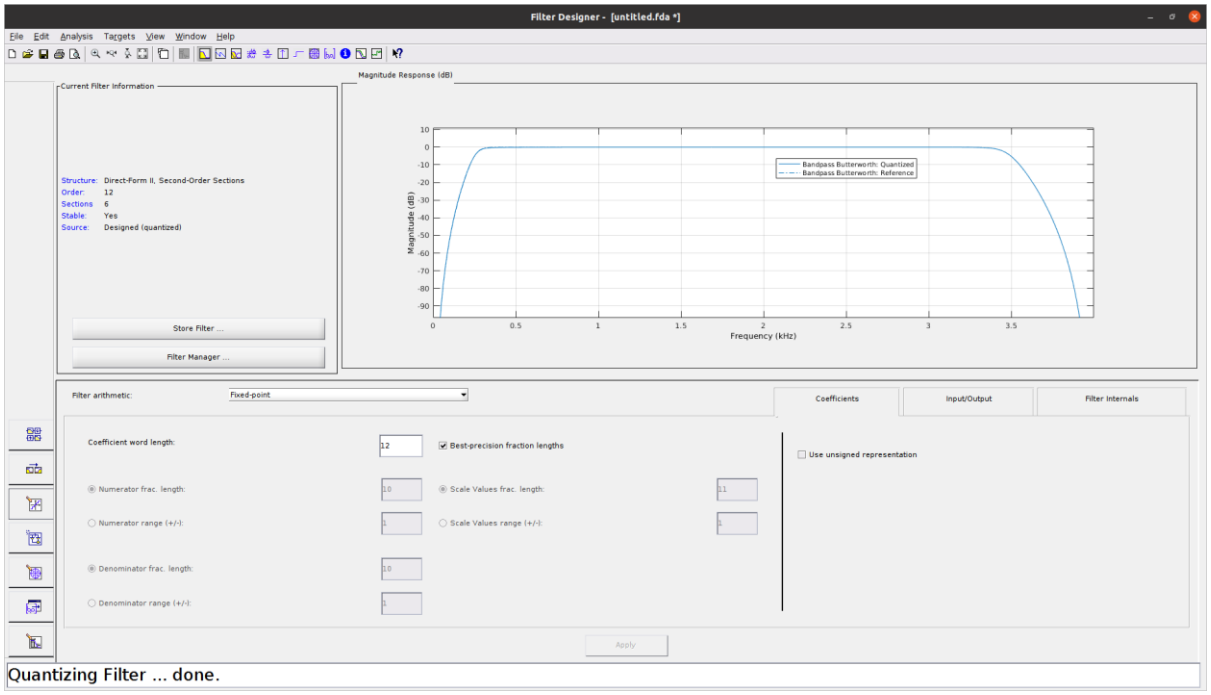
Editor - /home/caccolillo/Downloads/test_filter_colin/design_filter.m

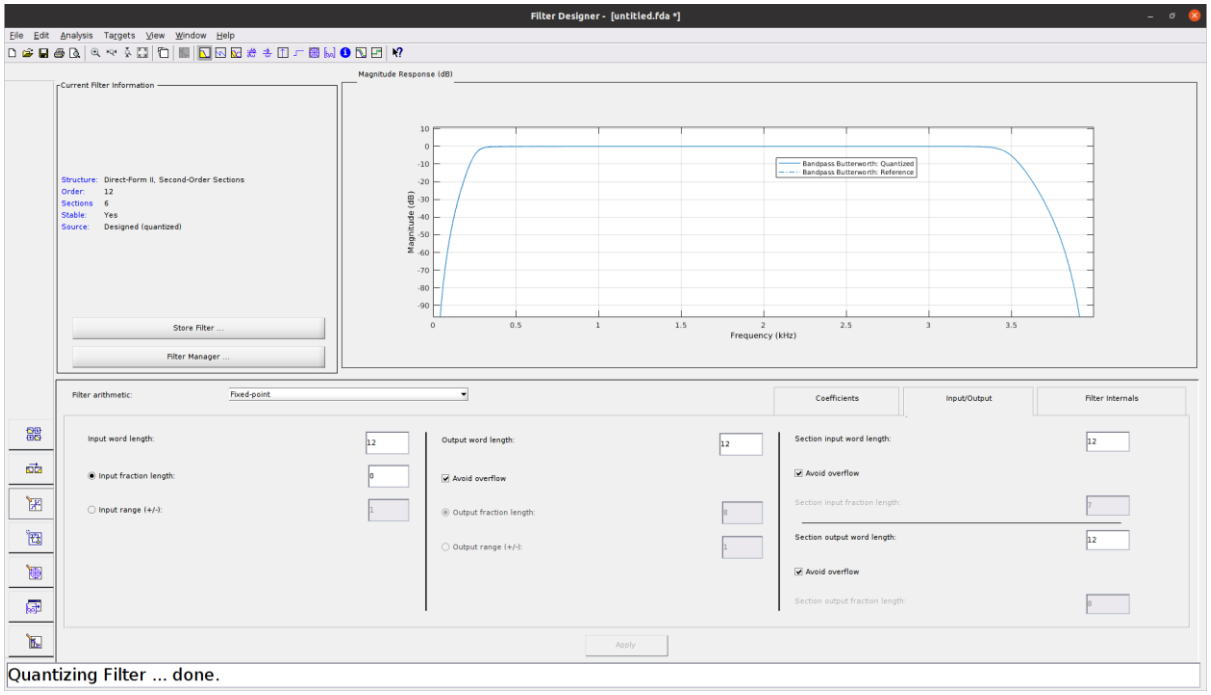
design_filter.m

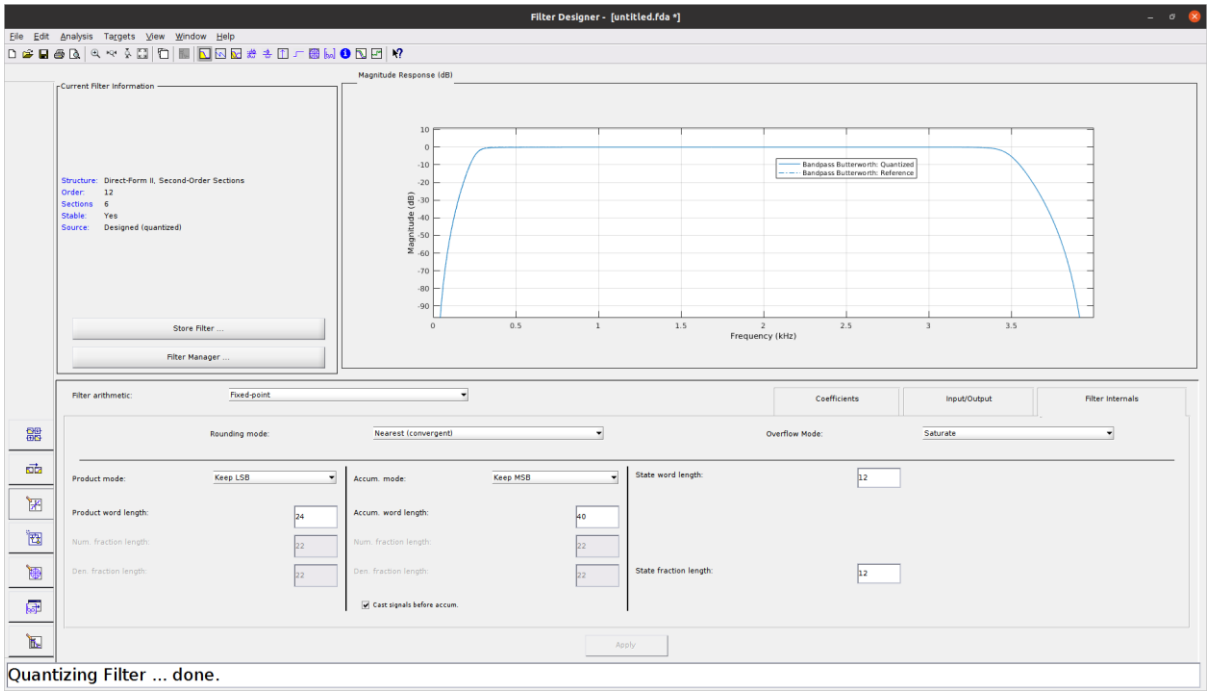
```
function Hd = design_filter
%DESIGN_FILTER Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 9.11 and DSP System Toolbox 9.13.
% Generated on: 09-May-2025 13:37:41

% Butterworth Bandpass filter designed using FDESIGN.BANDPASS.

% All frequency values are in Hz.
Fs = 8000;  % Sampling Frequency

Fstop1 = 150;          % First Stopband Frequency
Fpass1 = 300;          % First Passband Frequency
Fpass2 = 3400;         % Second Passband Frequency
Fstop2 = 3550;         % Second Stopband Frequency
Astop1 = 10;           % First Stopband Attenuation (dB)
Apass  = 1;            % Passband Ripple (dB)
Astop2 = 10;           % Second Stopband Attenuation (dB)
match  = 'stopband';   % Band to match exactly

% Construct an FDESIGN object and call its BUTTER method.
h  = fdesign.bandpass(Fstop1, Fpass1, Fpass2, Fstop2, Astop1, Apass, ...
                      Astop2, Fs);
Hd = design(h, 'butter', 'MatchExactly', match);

% [EOF]

```

Quantize the filter

Then we quantize the filter with the following settings:

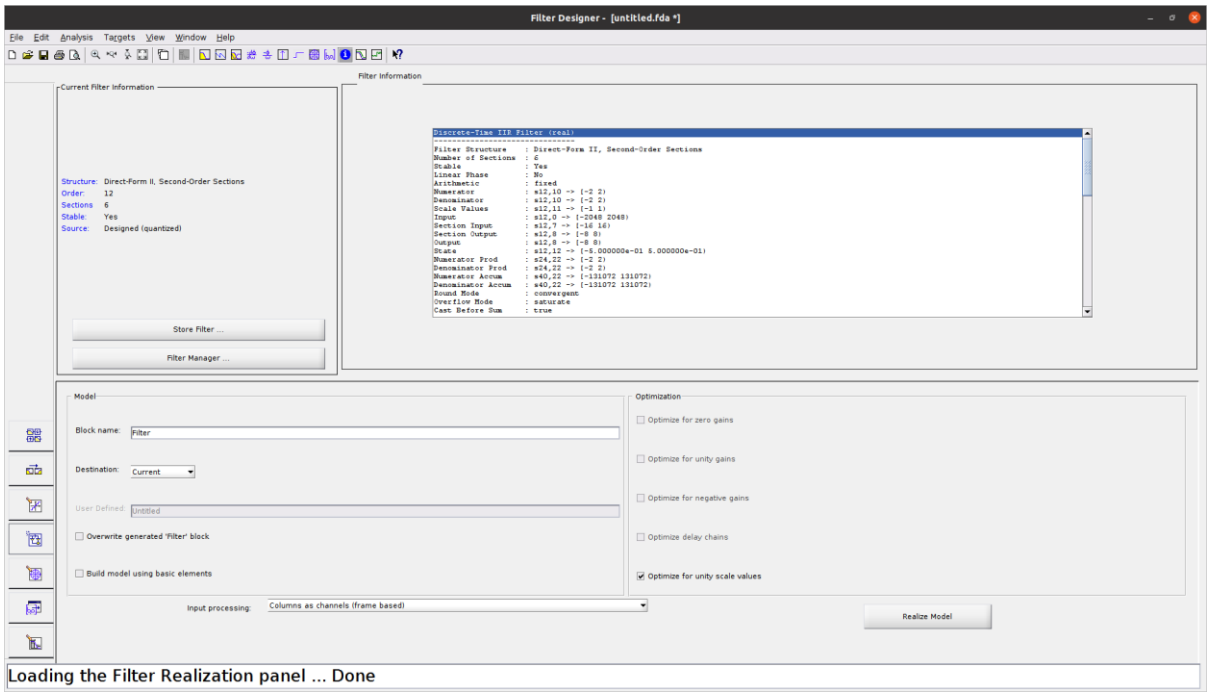

Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form II, Second-Order Sections
Order: 12
Sections: 6
Stable: Yes
Source: Designed (quantized)
Store Filter ...
Filter Manager ...
Magnitude Response (dB)
Bandpass Butterworth: Quantized
Bandpass Butterworth: Reference
Magnitude (dB)
Frequency (kHz)
Filter arithmetic:
Fixed-point
Coefficients
Input/Output
Filter Internals
Coefficient word length:
Best-precision fraction lengths
Use unsigned representation
Numerator frac. length:
Scale Values frac. length:
Numerator range (+/-):
Scale Values range (+/-):
Denominator frac. length:
Denominator range (+/-):
Apply
Quantizing Filter ... done.


Filter Designer - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form II, Second-Order Sections
Order: 12
Sections: 6
Stable: Yes
Source: Designed (quantized)
Store Filter ...
Filter Manager ...
Magnitude Response (dB)
Bandpass Butterworth: Quantized
Bandpass Butterworth: Reference
Magnitude (dB)
Frequency (kHz)
Filter arithmetic:
Fixed-point
Coefficients
Input/Output
Filter Internals
Input word length:
Input fraction length:
Input range (+/-):
Output word length:
Avoid overflow
Output fraction length:
Output range (+/-):
Section input word length:
Avoid overflow
Section input fraction length:
Section output word length:
Avoid overflow
Section output fraction length:
Apply
Quantizing Filter ... done.

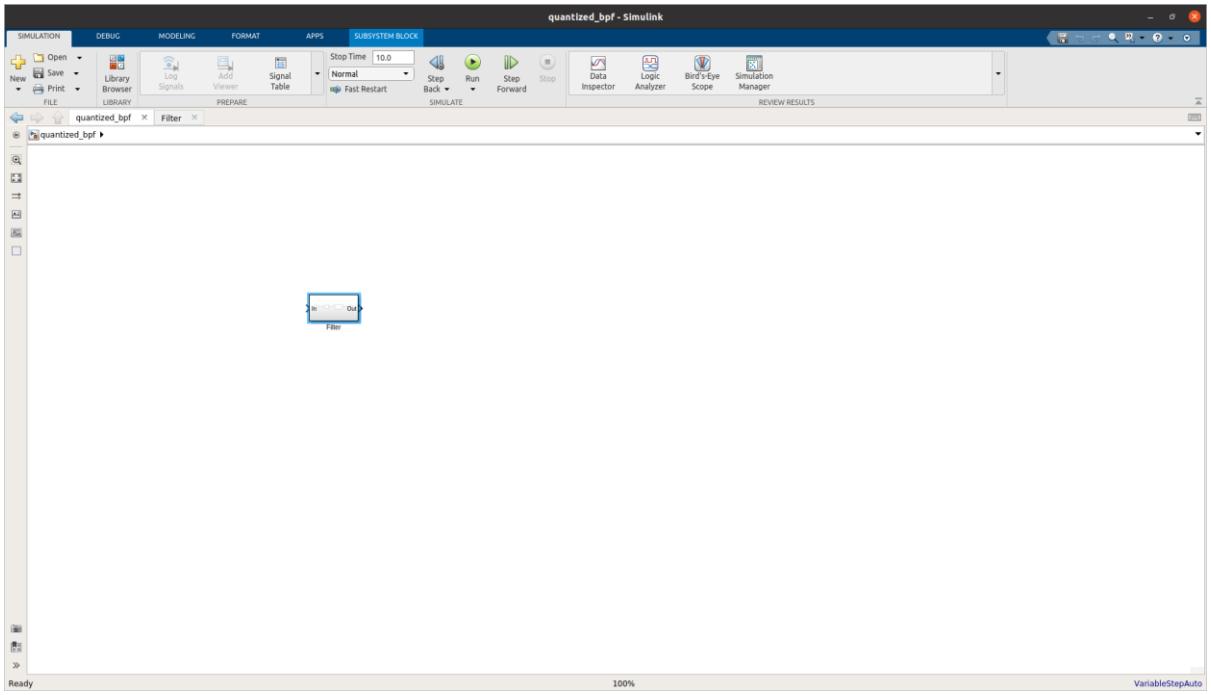


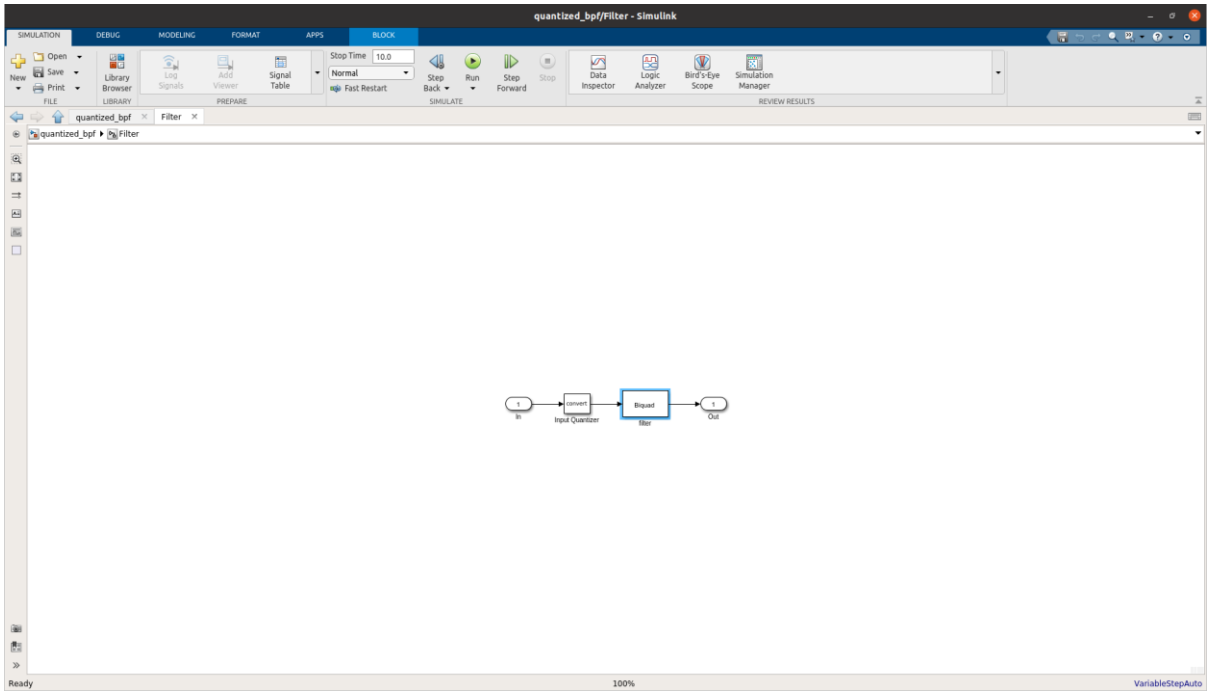
Export to Simulink

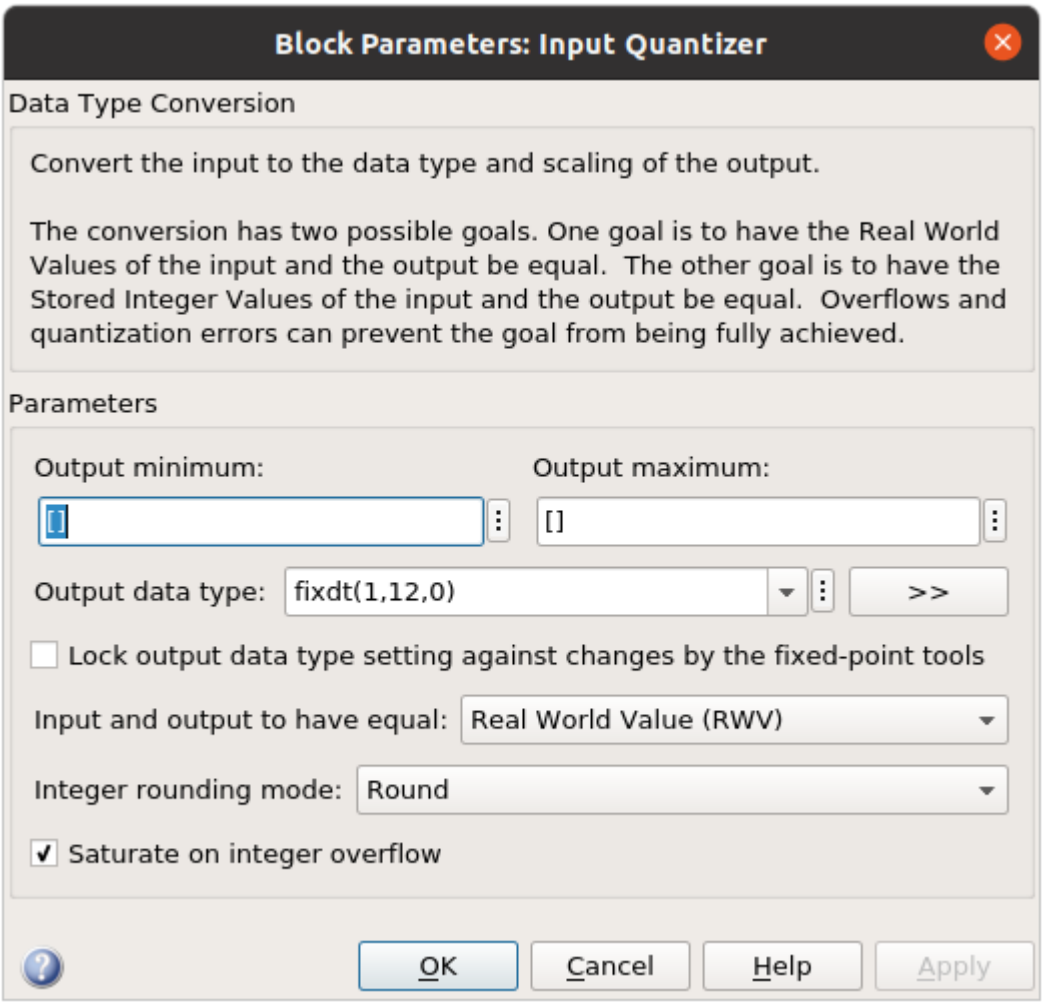
The quantized filter is exported in Simulink by going to file -> export to Simulink model:

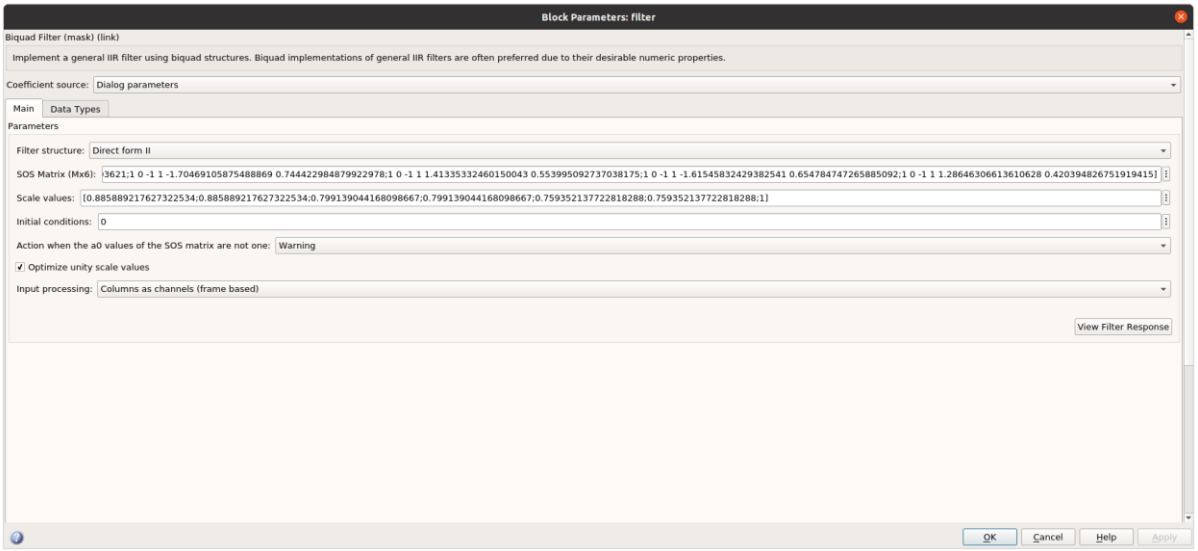


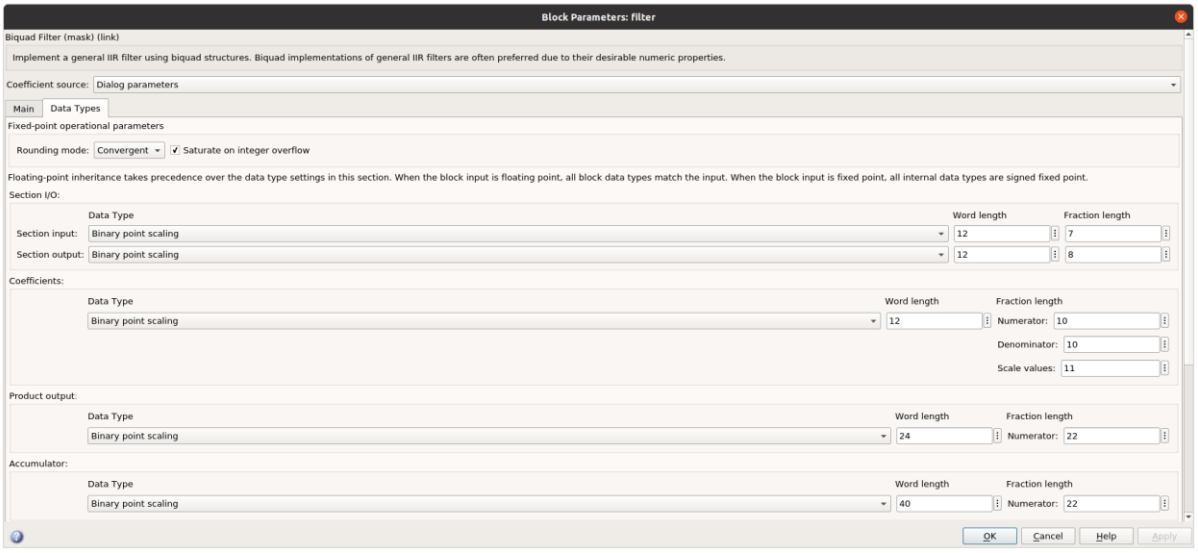
And pressing on realize model:


quantized_bpf - Simulink
SIMULATION
DEBUG
MODELING
FORMAT
APPS
SUBSYSTEM BLOCK
New
Open
Save
Print
Library Browser
Log Signals
Add Viewer
Signal Table
Stop Time
10.0
Normal
Fast Restart
Step Back
Run
Step Forward
Stop
Data Inspector
Logic Analyzer
Bird's-Eye Scope
Simulation Manager
FILE
LIBRARY
PREPARE
SIMULATE
REVIEW RESULTS
quantized_bpf
Filter
In
Out
Ready
100%
VariableStepAuto

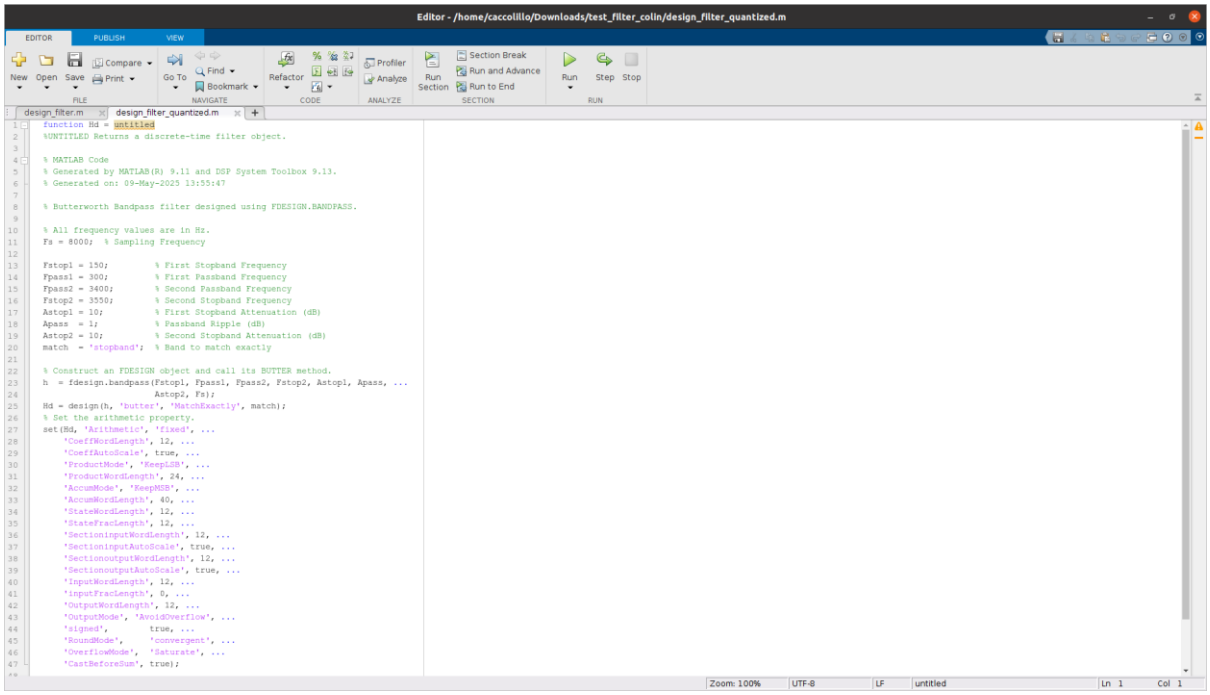

quantized_bpf/Filter - Simulink
SIMULATION
DEBUG
MODELING
FORMAT
APPS
BLOCK
New
Open
Save
Print
Library Browser
Log Signals
Add Viewer
Signal Table
Stop Time
10.0
Normal
Fast Restart
Step Back
Run
Step Forward
Stop
Data Inspector
Logic Analyzer
Bird's-Eye Scope
Simulation Manager
FILE
LIBRARY
PREPARE
SIMULATE
REVIEW RESULTS
quantized_bpf
Filter
1
In
convert
Input Quantizer
Biquad
filter
1
Out
Ready
100%
VariableStepAuto


Block Parameters: Input Quantizer
Data Type Conversion
Convert the input to the data type and scaling of the output.
The conversion has two possible goals. One goal is to have the Real World Values of the input and the output be equal. The other goal is to have the Stored Integer Values of the input and the output be equal. Overflows and quantization errors can prevent the goal from being fully achieved.
Parameters
Output minimum:
[]
Output maximum:
[]
Output data type:
fixdt(1,12,0)
>>
Lock output data type setting against changes by the fixed-point tools
Input and output to have equal:
Real World Value (RWV)
Integer rounding mode:
Round
Saturate on integer overflow
OK
Cancel
Help
Apply


Block Parameters: filter
Biquad Filter (mask) (link)
Implement a general IIR filter using biquad structures. Biquad implementations of general IIR filters are often preferred due to their desirable numeric properties.
Coefficient source:
Dialog parameters
Main
Data Types
Parameters
Filter structure:
Direct form II
SOS Matrix (Mx6):
Scale values:
Initial conditions:
0
Action when the a0 values of the SOS matrix are not one:
Warning
Optimize unity scale values
Input processing:
Columns as channels (frame based)
View Filter Response
OK
Cancel
Help
Apply



the filter can be replicated via a script, generated in the filter designer GUI:



Modifying filter parameters either via the GUI or configuration panels is easy.

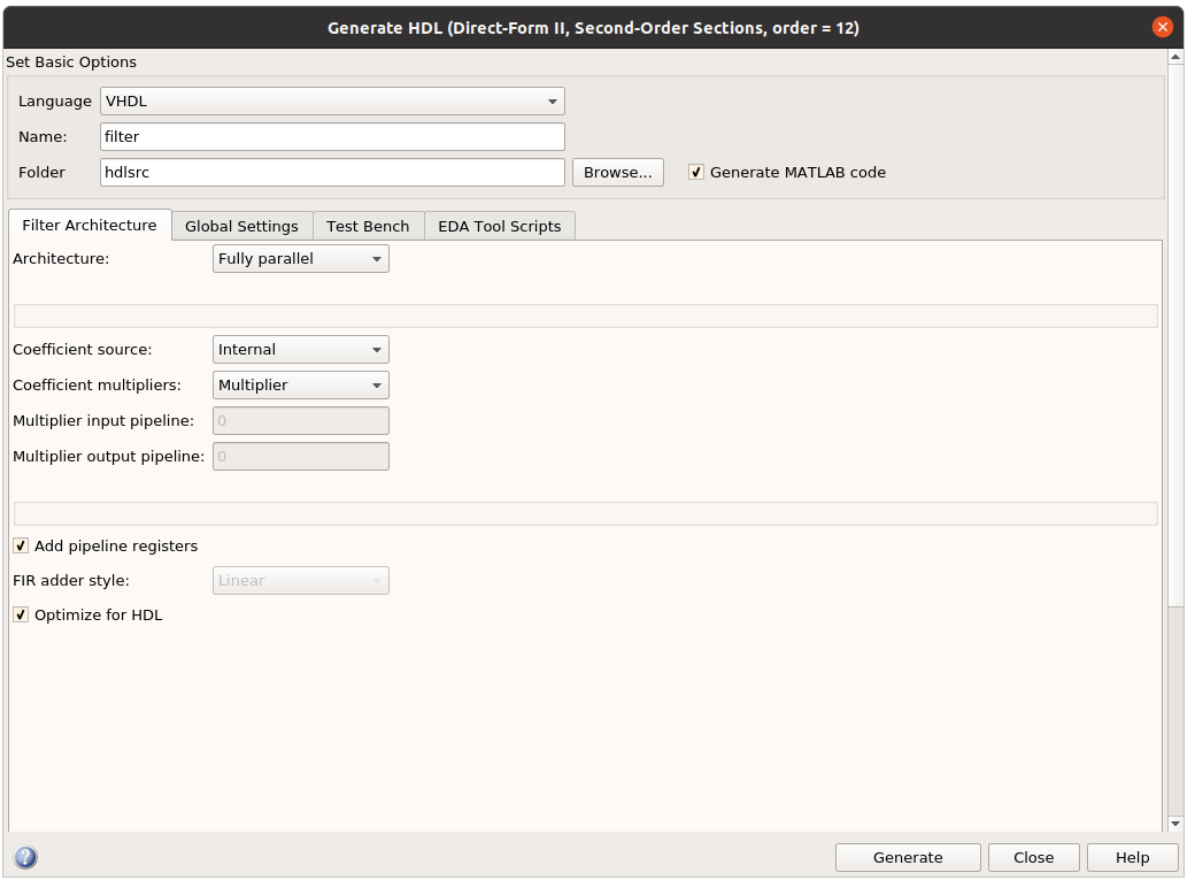
Generate VHDL code

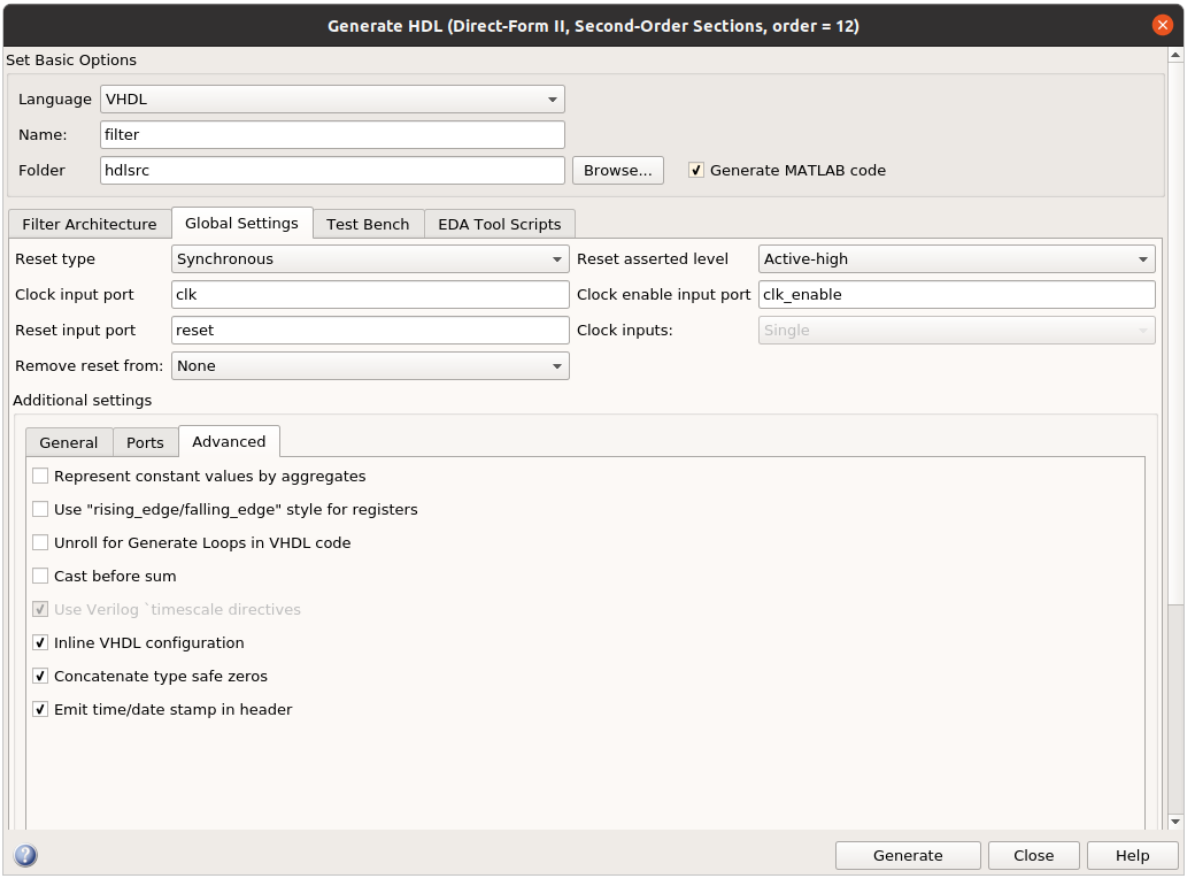
Instructions on:

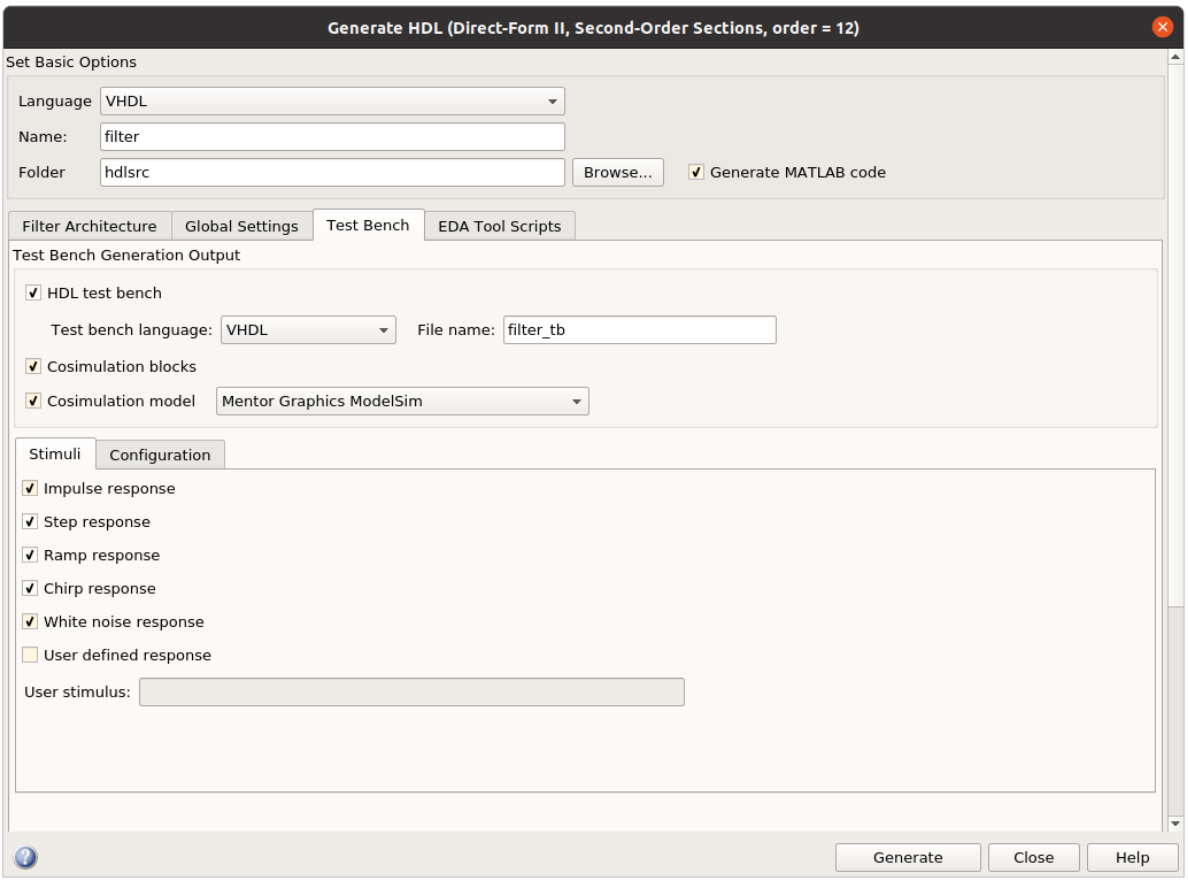
https://uk.mathworks.com/help/hdlfilter/basic-fir-filter.html

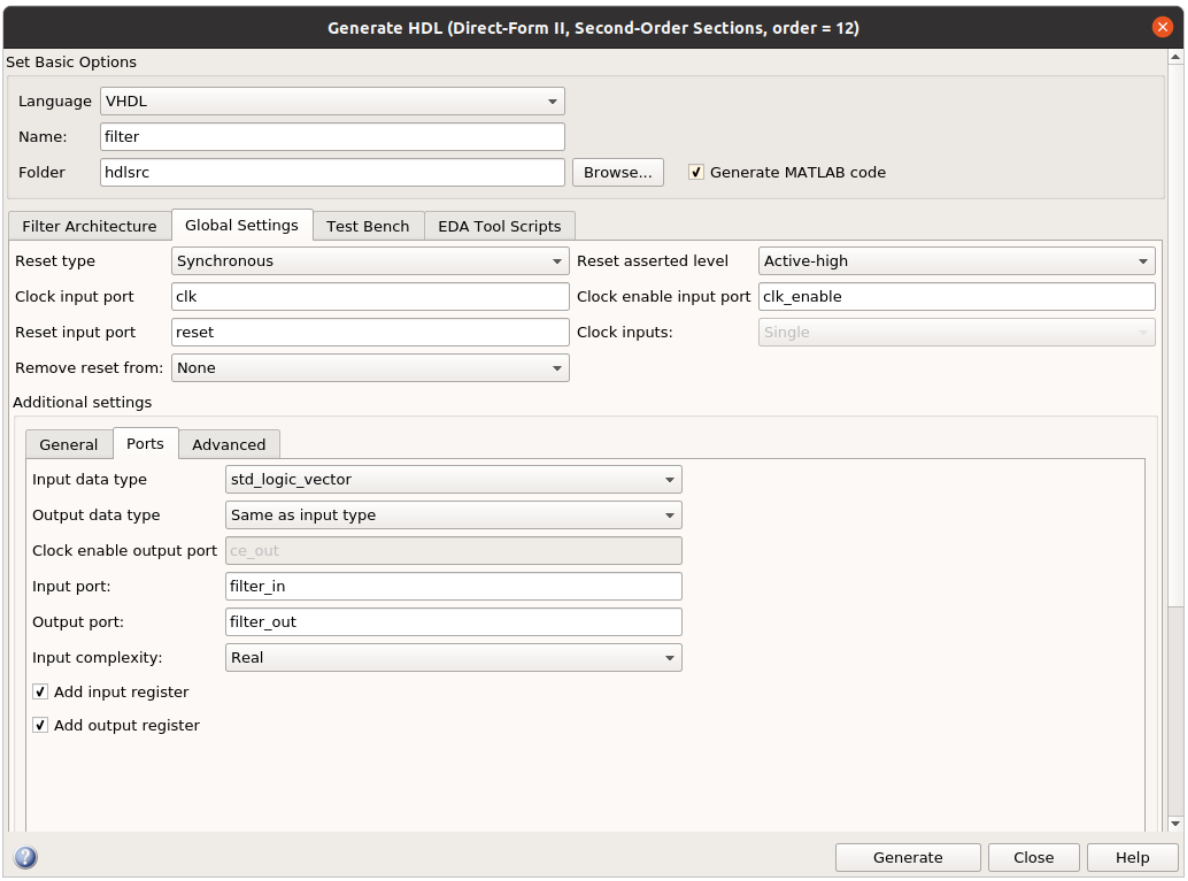
Are followed.

Going on targets->generate HDL, the generation process gets configured as follows:

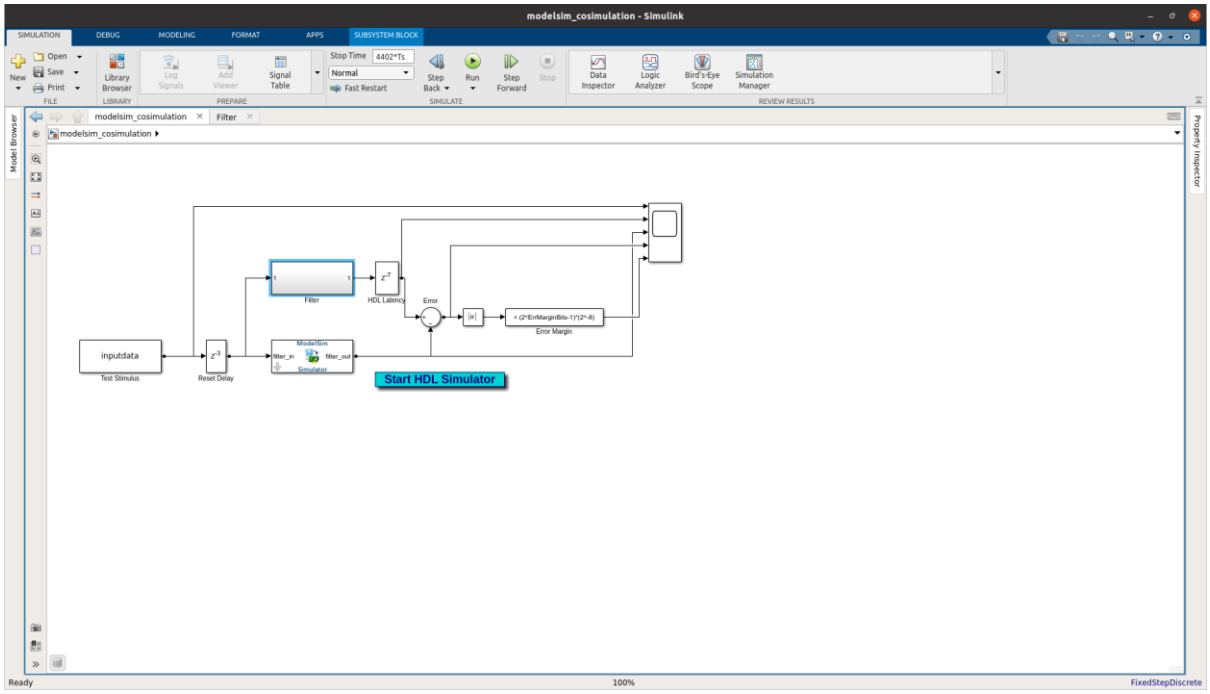

Generate HDL (Direct-Form II, Second-Order Sections, order = 12)
Set Basic Options
Language
VHDL
Name:
filter
Folder
hdlsrc
Browse...
Generate MATLAB code
Filter Architecture
Global Settings
Test Bench
EDA Tool Scripts
Architecture:
Fully parallel
Coefficient source:
Internal
Coefficient multipliers:
Multiplier
Multiplier input pipeline:
0
Multiplier output pipeline:
0
Add pipeline registers
FIR adder style:
Linear
Optimize for HDL
Generate
Close
Help

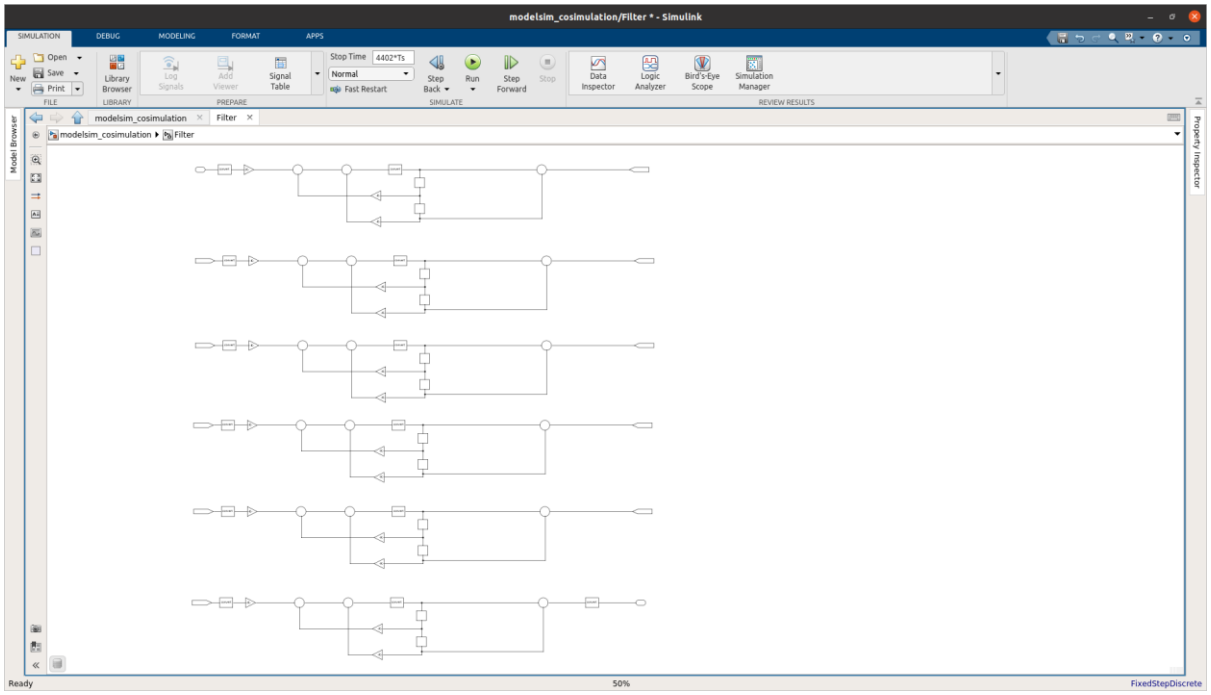

Generate HDL (Direct-Form II, Second-Order Sections, order = 12)
Set Basic Options
Language VHDL
Name: filter
Folder hdlsrc
Browse...
Generate MATLAB code
Filter Architecture
Global Settings
Test Bench
EDA Tool Scripts
Reset type Synchronous
Reset asserted level Active-high
Clock input port clk
Clock enable input port clk_enable
Reset input port reset
Clock inputs: Single
Remove reset from: None
Additional settings
General
Ports
Advanced
Represent constant values by aggregates
Use "rising_edge/falling_edge" style for registers
Unroll for Generate Loops in VHDL code
Cast before sum
Use Verilog `timescale directives
Inline VHDL configuration
Concatenate type safe zeros
Emit time/date stamp in header
Generate
Close
Help


Generate HDL (Direct-Form II, Second-Order Sections, order = 12)
Set Basic Options
Language VHDL
Name: filter
Folder hdlsrc
Browse...
Generate MATLAB code
Filter Architecture
Global Settings
Test Bench
EDA Tool Scripts
Test Bench Generation Output
HDL test bench
Test bench language: VHDL
File name: filter_tb
Cosimulation blocks
Cosimulation model Mentor Graphics ModelSim
Stimuli
Configuration
Impulse response
Step response
Ramp response
Chirp response
White noise response
User defined response
User stimulus:
Generate
Close
Help

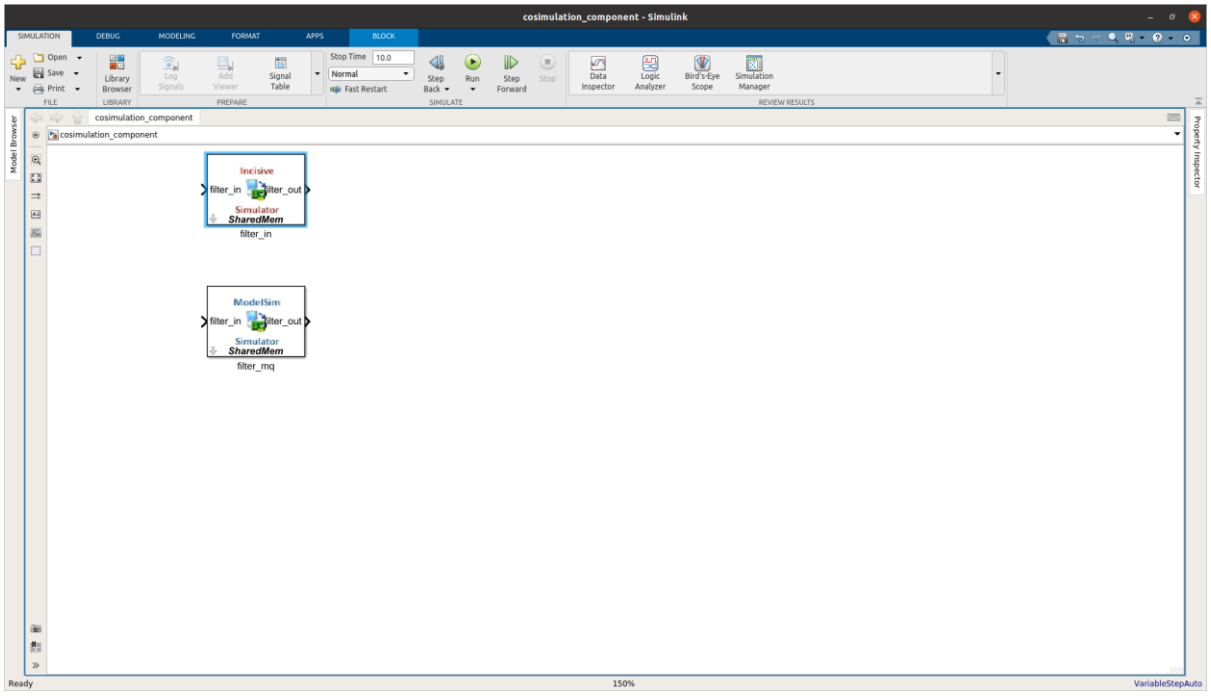


Then by pressing on generate, we get a model to be used with ”Modelsim” cosimulation:

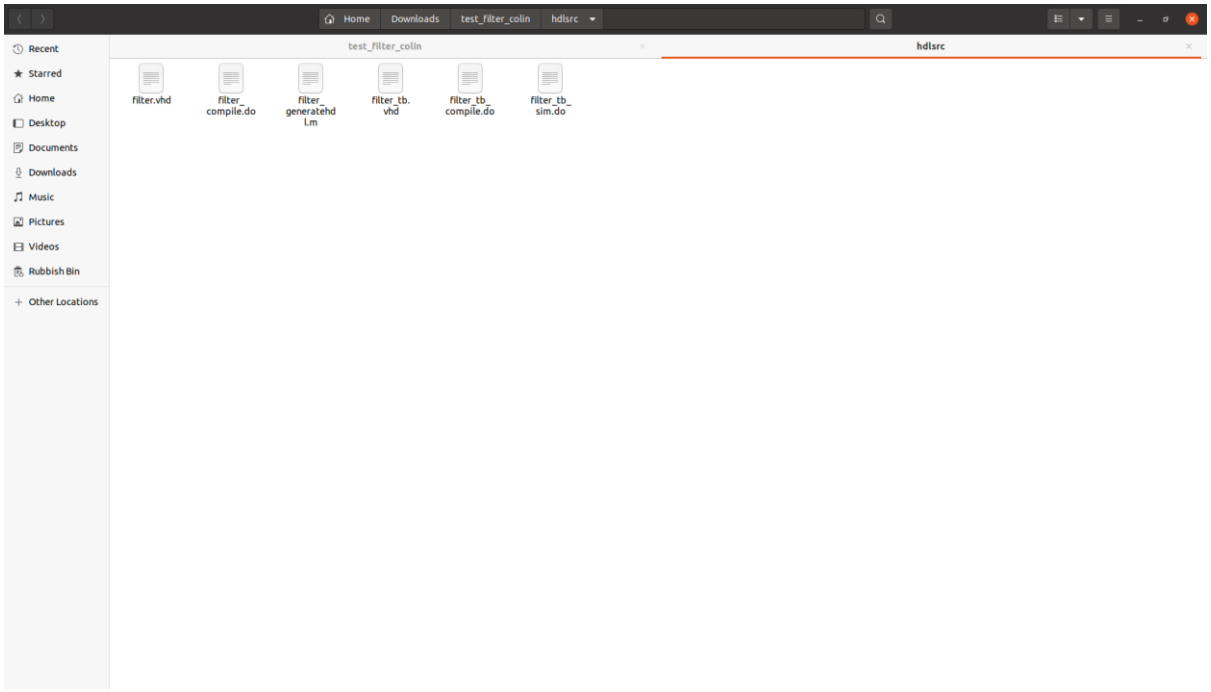


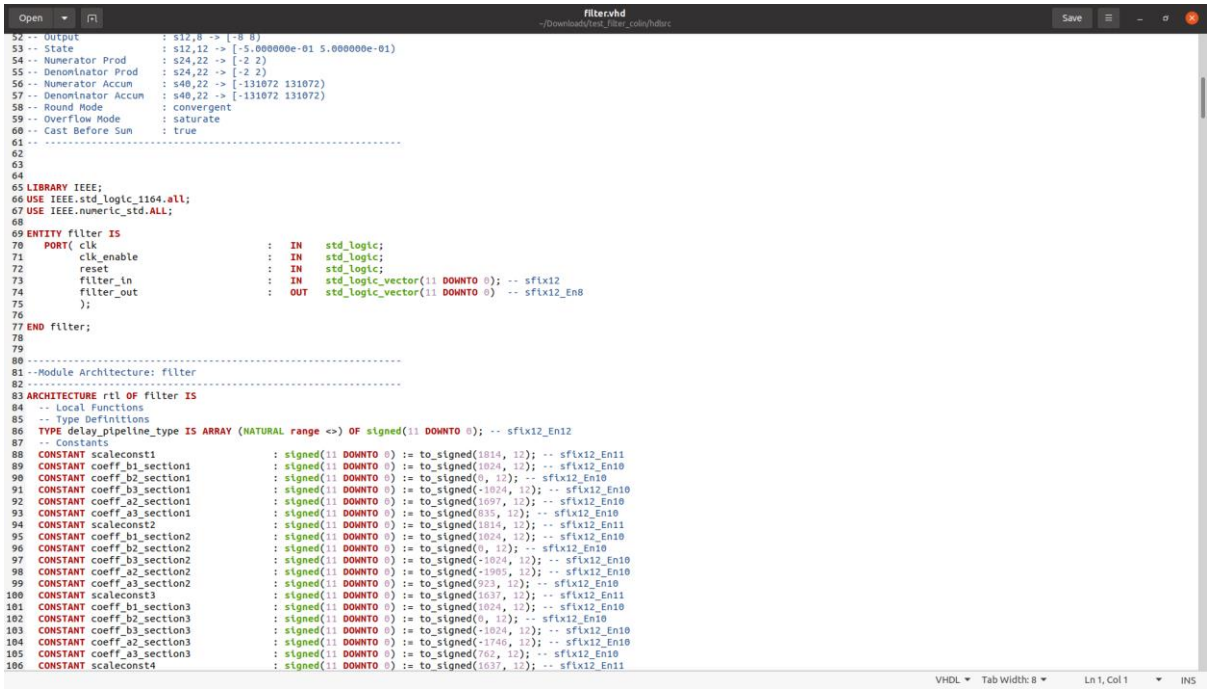


A model containing the cosimulation block only:



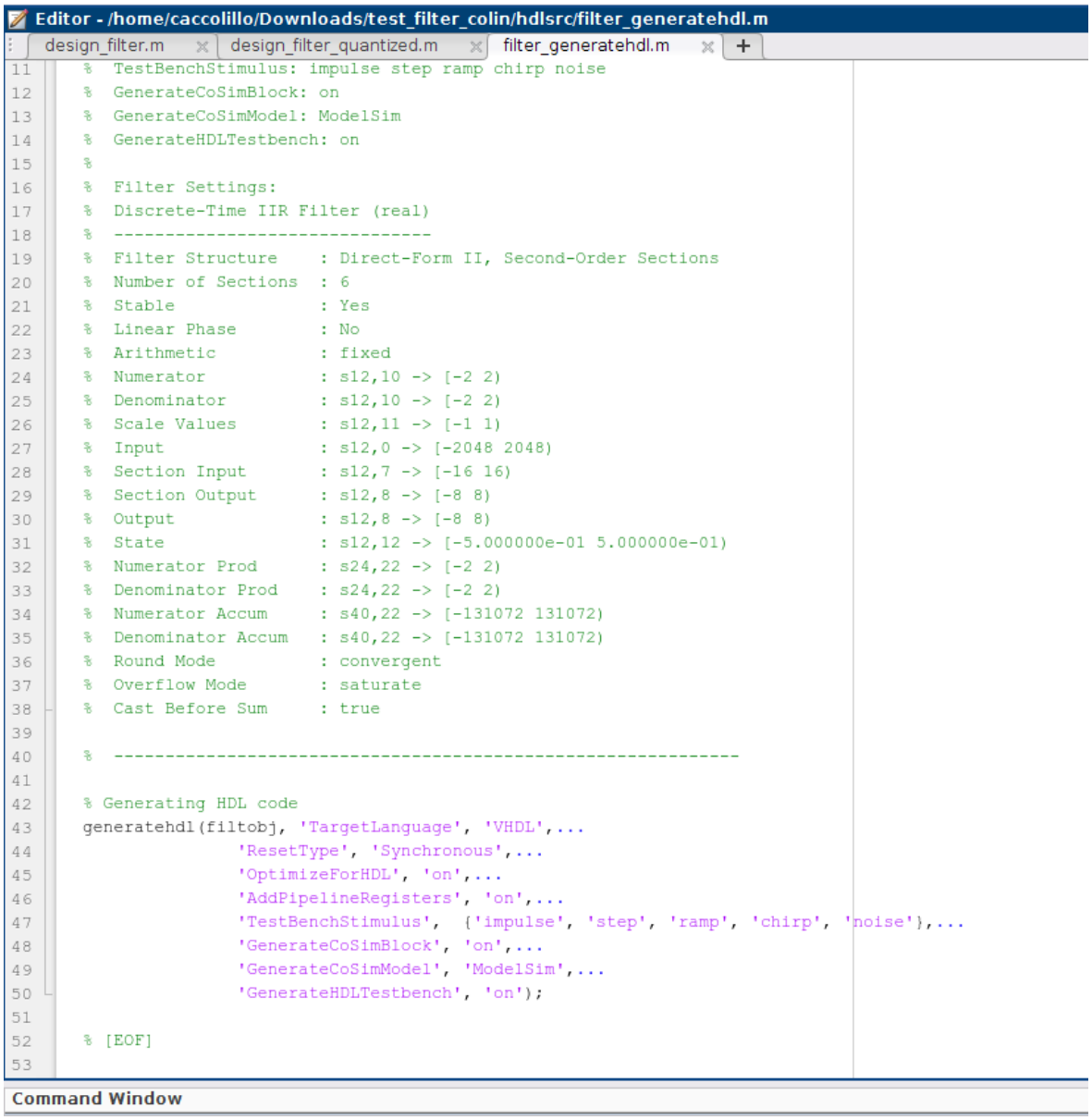
HDL code, HDL testbench and simulation scripts are generated in a folder local to the project:





The HDL code is HW agnostic.

A script with the code generation settings gets created:

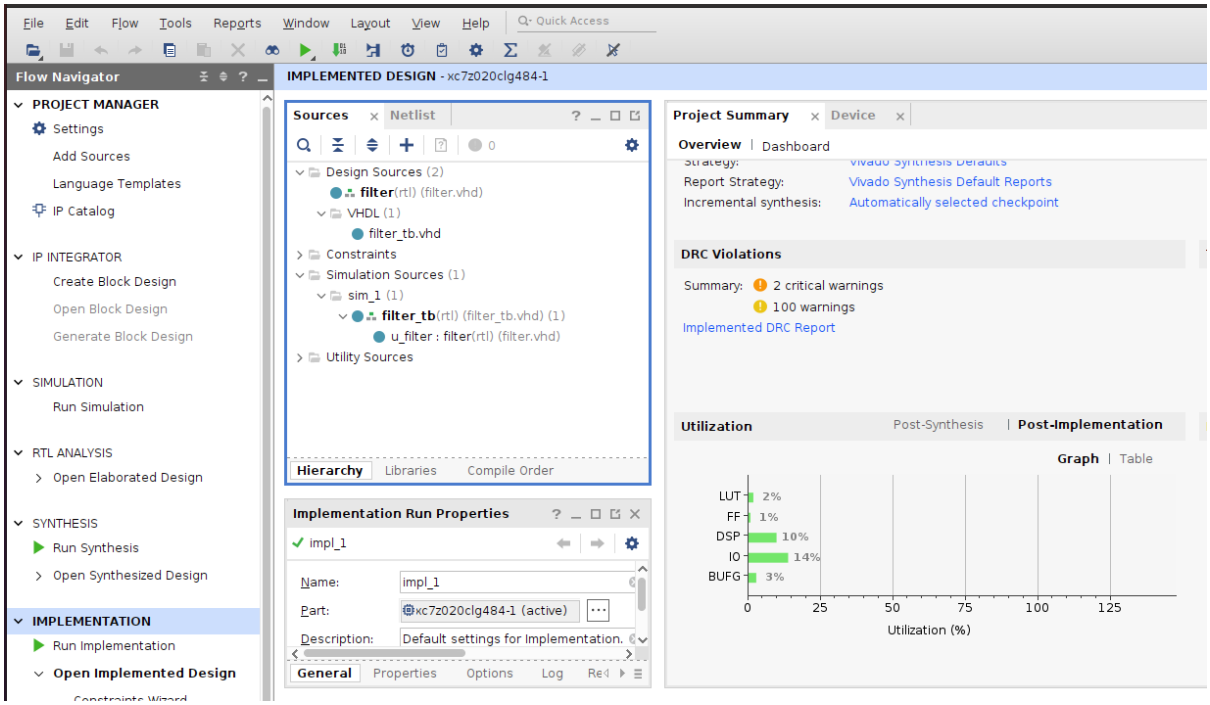


HW complexity of the generated HDL code can be reduced by using the “architecture” flag in the configuration settings: we can opt for fully parallel, fully serial or partially serial architectures, trading off speed versus HW complexity.

Creating a Vivado project

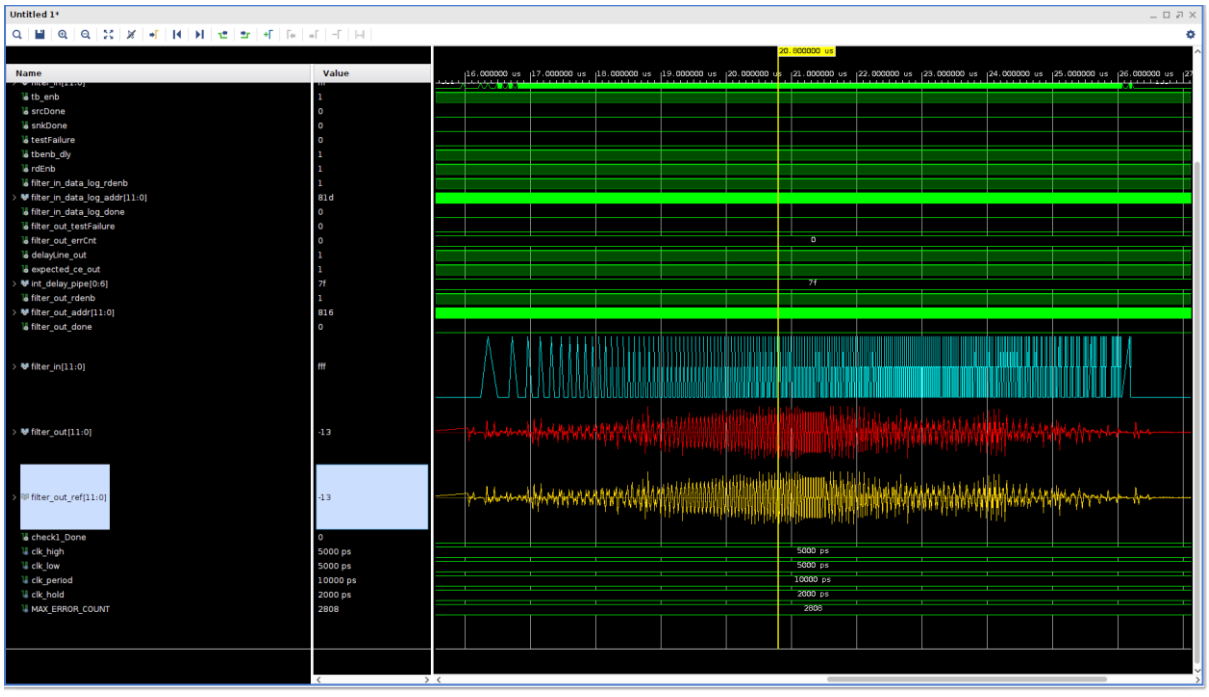
A vivado design is created to evaluate the generate code and to run the testbench.

The target device is a zynq 7020:

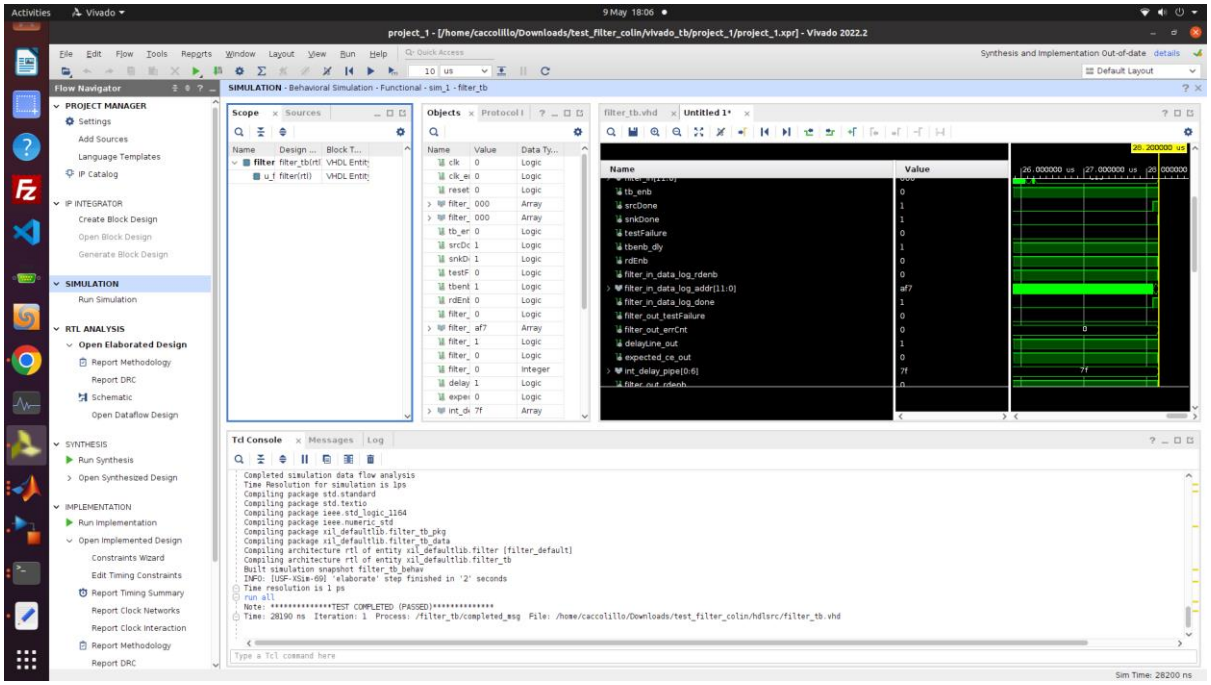


The fully parallel architecture is using 10% of the DSP slices. By going partially or fully serial, this number can be lowered.

By running the testbench:

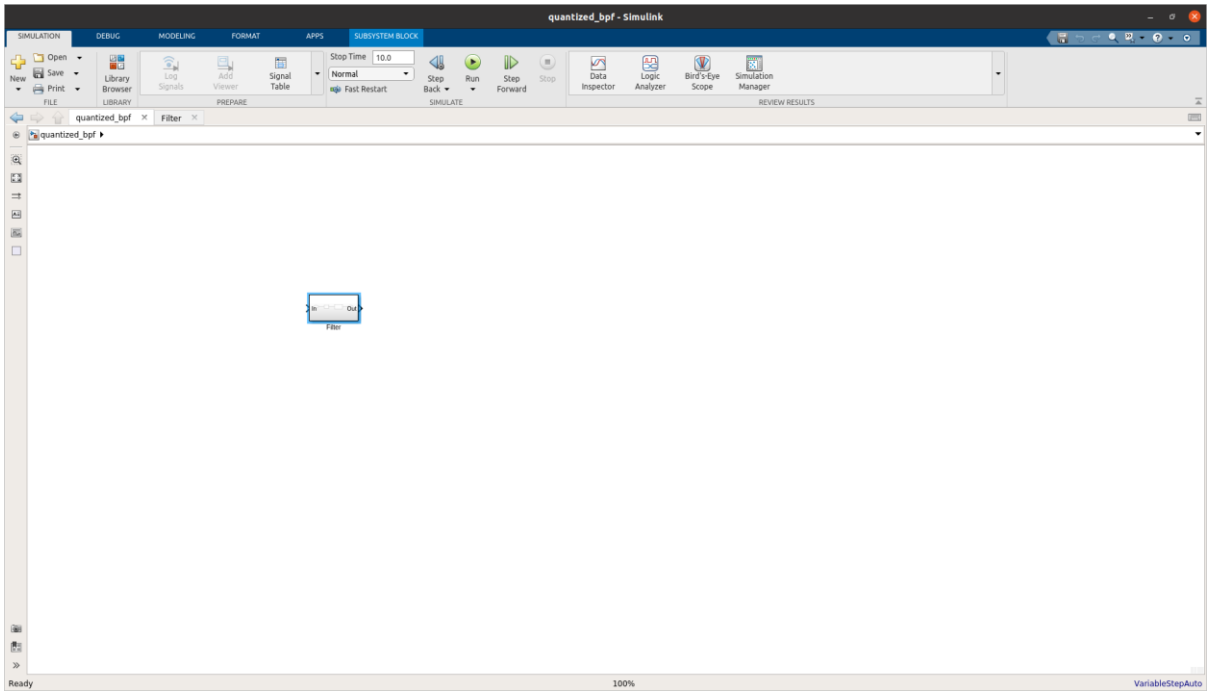


It is self checking and passes:

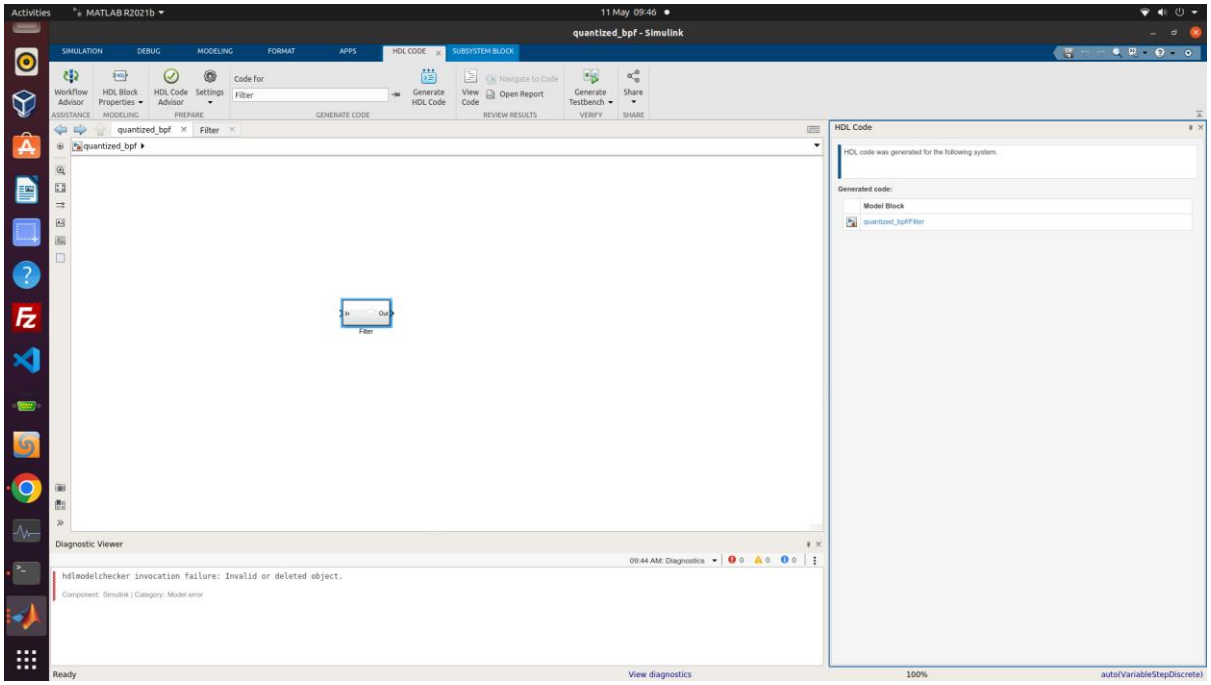


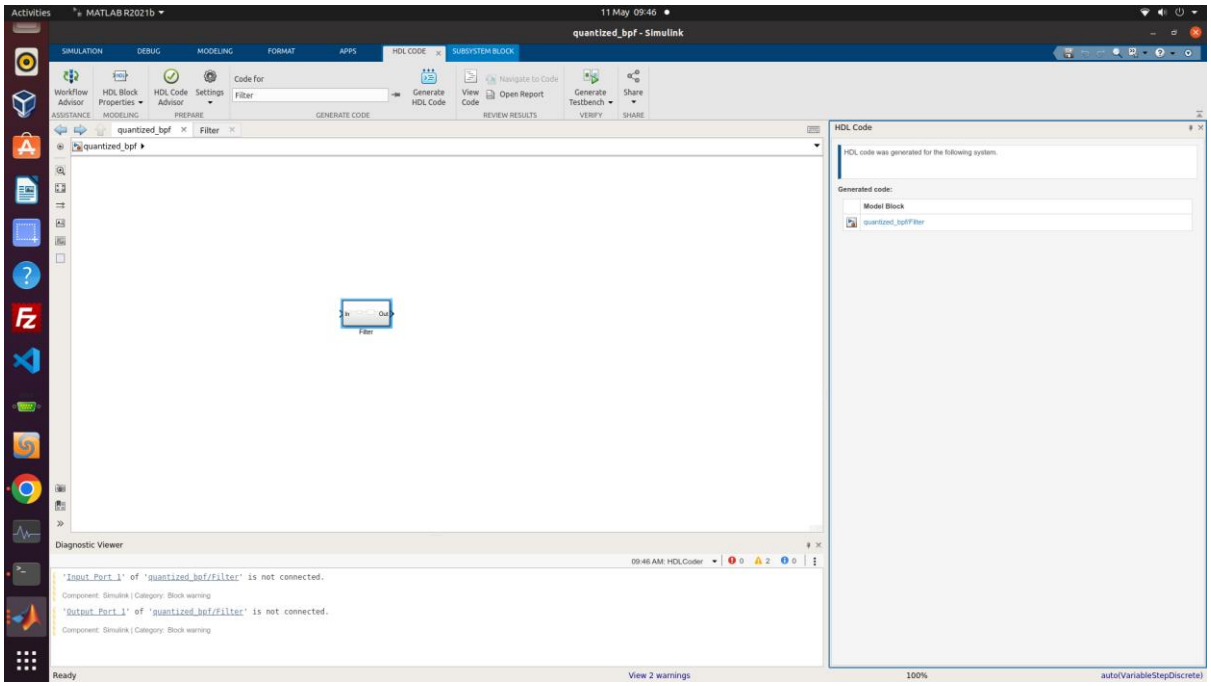
HDL coder flow

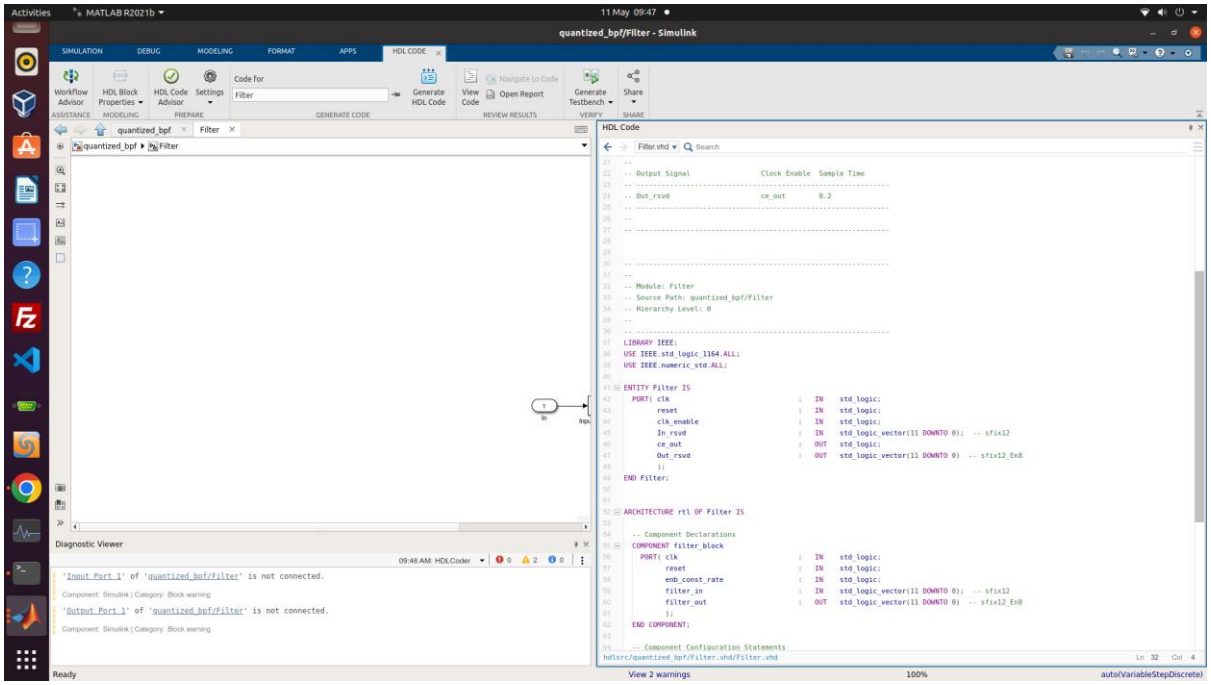
The model exported to Simulink seen earlier:

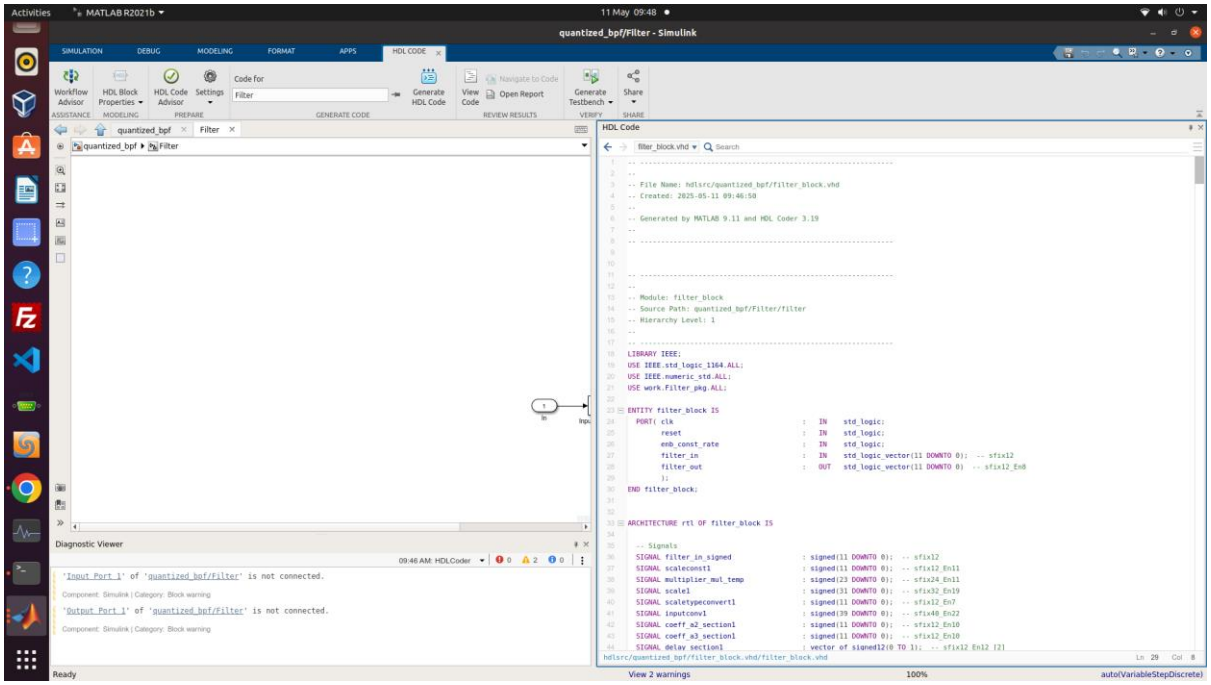


Can also be used with HDL coder:


Activities
MATLAB R2021b
11 May 09:46
quantized_bpf - Simulink
SIMULATION
DEBUG
MODELING
FORMAT
APPS
HDL CODE
SUBSYSTEM BLOCK
Workflow Advisor
HDL Block Properties
HDL Code Advisor
Settings
Code for
Filter
Generate HDL Code
View Code
Navigate to Code
Open Report
Generate Testbench
Share
ASSISTANCE
MODELING
PREPARE
GENERATE CODE
REVIEW RESULTS
VERIFY
SHARE
quantized_bpf
Filter
In
Out
HDL Code
HDL code was generated for the following system.
Generated code:
Model Block
quantized_bpf/Filter
Diagnostic Viewer
09:44 AM: Diagnostics
hdlmodelchecker invocation failure: Invalid or deleted object.
Component: Simulink | Category: Model error
Ready
View diagnostics
100%
auto(VariableStepDiscrete)


Activities
MATLAB R2021b
11 May 09:46
quantized_bpf - Simulink
SIMULATION
DEBUG
MODELING
FORMAT
APPS
HDL CODE
SUBSYSTEM BLOCK
Workflow Advisor
HDL Block Properties
HDL Code Advisor
Settings
Code for
Filter
Generate HDL Code
View Code
Navigate to Code
Open Report
Generate Testbench
Share
ASSISTANCE
MODELING
PREPARE
GENERATE CODE
REVIEW RESULTS
VERIFY
SHARE
quantized_bpf
Filter
In
Out
HDL Code
HDL code was generated for the following system.
Generated code:
Model Block
quantized_bpf/Filter
Diagnostic Viewer
09:46 AM: HDLCoder
'Input Port 1' of 'quantized_bpf/Filter' is not connected.
Component: Simulink | Category: Block warning
'Output Port 1' of 'quantized_bpf/Filter' is not connected.
Component: Simulink | Category: Block warning
Ready
View 2 warnings
100%
auto(VariableStepDiscrete)


Activities
MATLAB R2021b
11 May 09:47
quantized_bpf/Filter - Simulink
SIMULATION
DEBUG
MODELING
FORMAT
APPS
HDL CODE
Workflow Advisor
HDL Block Properties
HDL Code Advisor
Settings
Code for
Filter
Generate HDL Code
View Code
Navigate to Code
Open Report
Generate Testbench
Share
ASSISTANCE
MODELING
PREPARE
GENERATE CODE
REVIEW RESULTS
VERIFY
SHARE
quantized_bpf
Filter
quantized_bpf ▸ Filter
HDL Code
Filter.vhd
Search
-- Output Signal Clock Enable Sample Time
-- Out_rsvd ce_out 0.2
-- Module: Filter
-- Source Path: quantized_bpf/Filter
-- Hierarchy Level: 0
LIBRARY IEEE;
USE IEEE.std_logic_1164.ALL;
USE IEEE.numeric_std.ALL;
ENTITY Filter IS
PORT(clk : IN std_logic;
reset : IN std_logic;
clk_enable : IN std_logic;
In_rsvd : IN std_logic_vector(11 DOWNTO 0); -- sfix12
ce_out : OUT std_logic;
Out_rsvd : OUT std_logic_vector(11 DOWNTO 0) -- sfix12_En8
);
END Filter;
ARCHITECTURE rtl OF Filter IS
-- Component Declarations
COMPONENT filter_block
PORT(clk : IN std_logic;
reset : IN std_logic;
enb_const_rate : IN std_logic;
filter_in : IN std_logic_vector(11 DOWNTO 0); -- sfix12
filter_out : OUT std_logic_vector(11 DOWNTO 0) -- sfix12_En8
);
END COMPONENT;
-- Component Configuration Statements
hdlsrc/quantized_bpf/Filter.vhd/Filter.vhd
Ln 32 Col 4
Diagnostic Viewer
09:46 AM: HDLCoder
'Input Port 1' of 'quantized_bpf/Filter' is not connected.
Component: Simulink | Category: Block warning
'Output Port 1' of 'quantized_bpf/Filter' is not connected.
Component: Simulink | Category: Block warning
Ready
View 2 warnings
100%
auto(VariableStepDiscrete)


Activities
MATLAB R2021b
11 May 09:48
quantized_bpf/Filter - Simulink
SIMULATION
DEBUG
MODELING
FORMAT
APPS
HDL CODE
Workflow Advisor
HDL Block Properties
HDL Code Advisor
Settings
Code for
Filter
Generate HDL Code
View Code
Navigate to Code
Open Report
Generate Testbench
Share
ASSISTANCE
MODELING
PREPARE
GENERATE CODE
REVIEW RESULTS
VERIFY
SHARE
quantized_bpf
Filter
quantized_bpf ▸ Filter
HDL Code
filter_block.vhd
Search
-- File Name: hdlsrc/quantized_bpf/filter_block.vhd
-- Created: 2025-05-11 09:46:50
-- Generated by MATLAB 9.11 and HDL Coder 3.19
-- Module: filter_block
-- Source Path: quantized_bpf/Filter/filter
-- Hierarchy Level: 1
LIBRARY IEEE;
USE IEEE.std_logic_1164.ALL;
USE IEEE.numeric_std.ALL;
USE work.Filter_pkg.ALL;
ENTITY filter_block IS
PORT(clk : IN std_logic;
reset : IN std_logic;
enb_const_rate : IN std_logic;
filter_in : IN std_logic_vector(11 DOWNTO 0); -- sfix12
filter_out : OUT std_logic_vector(11 DOWNTO 0) -- sfix12_En8
);
END filter_block;
ARCHITECTURE rtl OF filter_block IS
-- Signals
SIGNAL filter_in_signed : signed(11 DOWNTO 0); -- sfix12
SIGNAL scaleconst1 : signed(11 DOWNTO 0); -- sfix12_En11
SIGNAL multiplier_mul_temp : signed(23 DOWNTO 0); -- sfix24_En11
SIGNAL scale1 : signed(31 DOWNTO 0); -- sfix32_En19
SIGNAL scaletypeconvert1 : signed(11 DOWNTO 0); -- sfix12_En7
SIGNAL inputconv1 : signed(39 DOWNTO 0); -- sfix40_En22
SIGNAL coeff_a2_section1 : signed(11 DOWNTO 0); -- sfix12_En10
SIGNAL coeff_a3_section1 : signed(11 DOWNTO 0); -- sfix12_En10
SIGNAL delay_section1 : vector of signed12(0 TO 1); -- sfix12_En12 [2]
hdlsrc/quantized_bpf/filter_block.vhd/filter_block.vhd
Ln 29 Col 8
Diagnostic Viewer
09:46 AM: HDLCoder
'Input Port 1' of 'quantized_bpf/Filter' is not connected.
Component: Simulink | Category: Block warning
'Output Port 1' of 'quantized_bpf/Filter' is not connected.
Component: Simulink | Category: Block warning
Ready
View 2 warnings
100%
auto(VariableStepDiscrete)