AN4596

PolarFire® FPGA DSP FIR Filter

Introduction

PolarFire® FPGA devices integrate a fifth-generation flash-based FPGA fabric architecture that includes embedded MathBlocks optimized specifically for Digital Signal Processing (DSP) applications such as Finite Impulse Response (FIR) filters, Infinite Impulse Response (IIR) filters, and Fast Fourier Transform (FFT) functions.

This document describes how to run the DSP FIR filter demo on a PolarFire Evaluation board.

Table of Contents

1. DSP FIR Filter Demo

The DSP FIR filter demo is implemented using two Libero® SoC designs. The design files include the following Libero project folders:

- **CoreFIR**: Contains the DSP filter design implemented using Microchip's COREFIR_PF and CoreFFT IP cores.
- **FIR_RTL**: Contains the DSP filter design implemented using the FIR RTL inference and Microchip's CoreFFT IP core.

The demo design consists of:

- A FIR filter of tap 127 with re-loadable coefficients.
- A 256 point FFT on filter output to view spectrum.
- A GUI UART interface from host PC to generate filter coefficients and input signals (Pass-Band frequency and Stop-Band frequency). The GUI UART also plots the input/output waveforms and the required spectrum.

These demo designs can be programmed using either of the following options:

- **Using the .job file**: To program the device using the `.job` file, see 5. Appendix 1: Programming the Device Using FlashPro Express.
- **Using Libero SoC**: To program the device using Libero SoC, see 2. Libero Design Flow. Use this option when the demo design is modified.

1.1 Demo Requirements

The following table lists the hardware, software, and IP requirements for the demo.

Table 1-1. Design Requirements

Requirements	Version
Operating System	Windows® 10
Hardware	
PolarFire® Evaluation Kit (MPF300-EVAL-KIT)	Rev D or later
Software	
FlashPro® Express	**Note:** See the `readme.txt` file provided in the design files for the software versions used with this reference design.
Libero® SoC	

Important: Libero SmartDesign and configuration screenshots shown in this guide are for illustration purpose only. Open the Libero design to see the latest updates.

1.2 Prerequisites

Before you start, download the demo design files from the following location:

1. For demo design, download the files from: www.microchip.com/en-us/application-notes/AN4596
2. Download and install the Libero SoC software (as indicated on the website for this design) on the host PC from the following location:
   Libero SoC Documentation.
   Note: The latest versions of ModelSim® and Synplify Pro® are included in the Libero SoC installation package.

Important: A Libero Gold license is required to evaluate the designs using the PolarFire Evaluation Kit.

1.3 DSP FIR Design

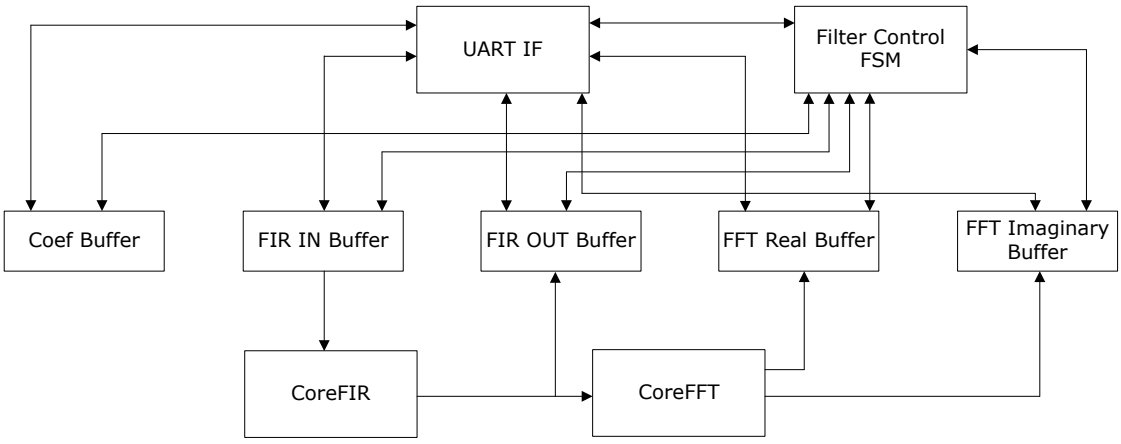
The PolarFire DSP FIR demo design is developed for demonstrating filtering applications using the DSP blocks such as FIR and FFT IPs.

The following steps describe the data flow in the design:

1. Upon UART handshaking (sending and receiving the known patterns over UART bus to pre-verify the serial channel before actual usage), GUI sends the filter coefficients followed by filter input data.
2. UART IF block creates 16-bit packets and stores the data in the corresponding input data buffers and coefficient buffers.
3. Filter control FSM controls the following operations:
   - Reading the data from buffers
   - Writing the data into CoreFIR IP
4. Once the CoreFIR generates the output response, the data is stored into FIR OUT buffer.
5. CoreFFT real and imaginary outputs are stored into corresponding buffers.
6. UART IF block reads the data from FIR and FFT output buffers and sends the data to GUI through UART.

The top-level block diagram of the DSP FIR filter demo design is illustrated in the following figure.

Figure 1-1. Block Diagram



1.3.1 Design Implementation

This section shows the DSP Filter design implemented using the CoreFIR and CoreFFT IP cores in Libero SoC.

In the DSP FIR filter demo design, the host interface and the FIR filter are implemented in the fabric for lowpass, bandpass, and bandstop filtering operations. The testbench provided for this demo uses pre-generated filter coefficients and input signals (Passband frequency and Stopband frequency) and passes the values to the demo design. The CoreFIR_PF IP is used to suppress unwanted frequency components, and the CoreFFT IP is used to generate the output spectrum to verify the filtering operation.

The following figure shows the top-level SmartDesign.

Figure 1-2. Top-level SmartDesign

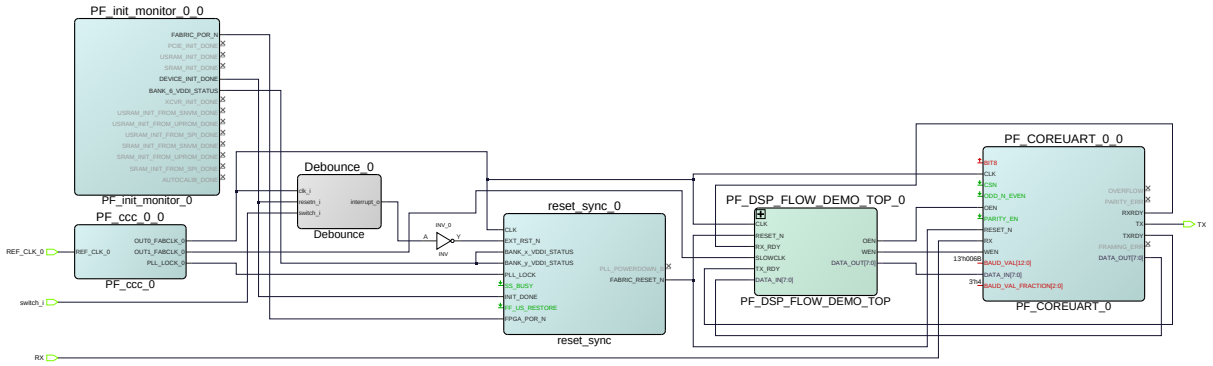
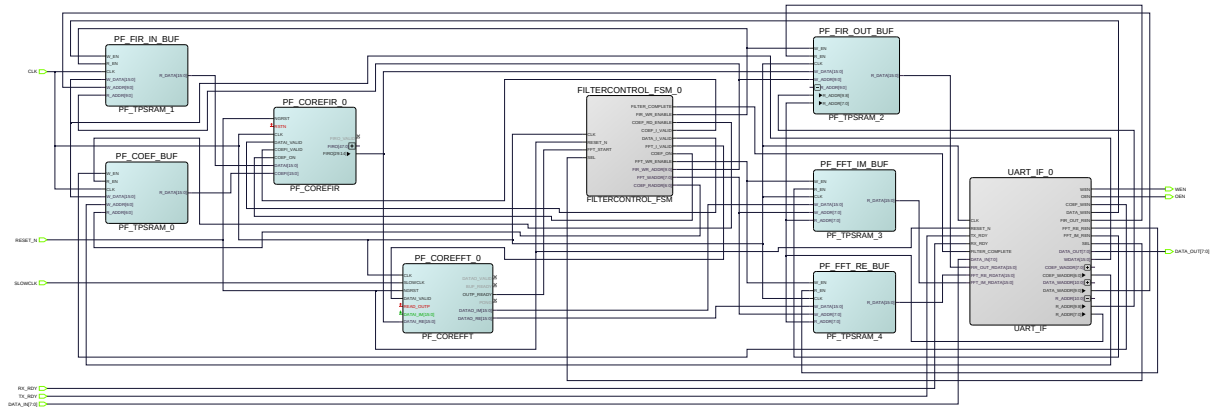


Table 1-2. I/O Signals

Signal	Description
Input Signals	
REF_CLK_0	Reference clock obtained from on-board 50 MHz oscillator.
RESET_N	This is the reset signal obtained from the SW push-button switch on the board.
RX	This is the UART receive data input.
Output Signals	
TX	This is the UART transmit data output.

The following figure shows the PF_DSP_FLOW_DEMO_TOP_0 SmartDesign.

Figure 1-3. DSP FIR Filter SmartDesign



Important: This design is similar to the design implemented using the FIR RTL inference and the CoreFFT IP core. The only difference is the use of the FIR RTL inference in the place of CoreFIR_PF.

1.3.2 Design Blocks and IP Configuration

The following IPs must be configured before simulating and implementing the demo design.

Important: For more information about IP blocks, see Figure 1-2 and Figure 1-3.

1.3.2.1 PF_TPSRAM IP

Five instances of PF_TPSRAM blocks in the design are described as follows:

- **Filter coefficient buffer (PF_COEF_BUF)**: Stores the coefficients received from GUI before sending it to the FIR
- **Input signal data buffer (PF_FIR_IN_BUF)**: Stores the input data received from GUI before sending it to the FIR
- **Output signal buffer (PF_FIR_OUT_BUF)**: Stores the FIR output data received from FIR IP before sending it to GUI
- **Output signal FFT real data buffer (PF_FFT_RE_BUF)**: Stores the output data (real part) received from FFT IP before sending it to GUI
- **Output signal FFT imaginary data buffer (PF_FFT_IM_BUF)**: Stores the output data (imaginary part) received from FFT IP before sending it to GUI

The PF_COEF_BUF, PF_FIR_IN_BUF, PF_FIR_OUT_BUF, PF_FFT_RE_BUF, and PF_FFT_IM_BUF blocks are configured for RAM size depth and width, as shown in the following figure and table.

Figure 1-4. Configuring the PF Two-Port Large SRAM

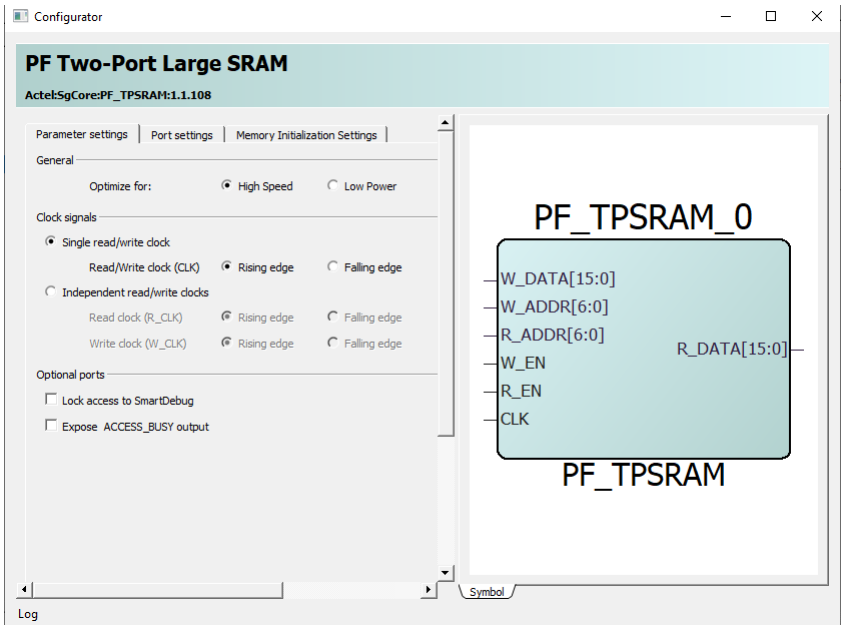


Table 1-3. PF_TPSRAM_0 Configuration for Data Buffers

Buffer	Write Port		Read Port	
	Depth	Width	Depth	Width
PF_COEF_BUF	128	16	128	16
PF_FIR_IN_BUF	1024	16	1024	16
PF_FIR_OUT_BUF	1024	16	1024	16
PF_FFT_RE_BUF	256	16	256	16
PF_FFT_IM_BUF	256	16	256	16

1.3.2.2 UART_IF_0

The UART_IF block (`UART_IF.v`) consists of a finite state machine handling control operations between UART and a fabric logic. Control operations include the loading of filter coefficients, filtering input data to the corresponding input data buffers and coefficient buffers, and sending and receiving data from UART.

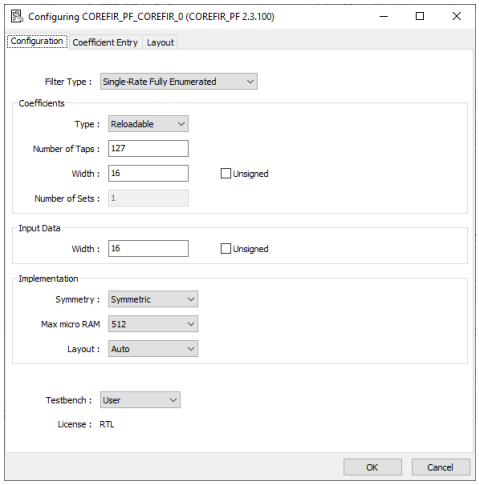
1.3.2.3 FILTERCONTROL_FSM_0

The filter control FSM block (`FILTER_CONTROL_FSM.v`) handles the data flow and controls signals of FIR filter and FFT. It loads the filtered data to the corresponding output buffer and moves the FFT output data to the corresponding FFT real and imaginary buffers.

1.3.2.4 PF_COREFIR_0

The PF_COREFIR_0 IP is used in reloadable coefficient mode to support lowpass, bandpass, and bandstop filters.

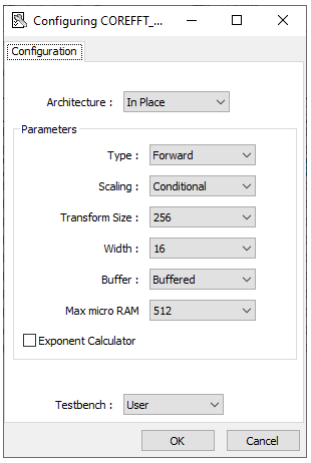
Figure 1-5. Configuring PF_COREFIR_0 IP



1.3.2.5 PF_COREFFT_0

The PF_COREFFT_0 IP generates the frequency spectrum of the filtered data, as shown in the following figure.

Figure 1-6. Configuring PF_COREFFT_0 IP

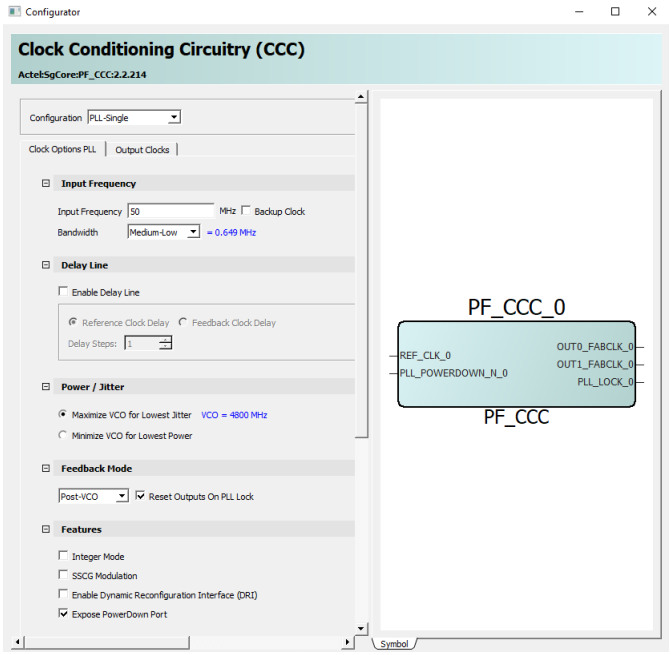


1.3.2.6 PF_ccc_0_0

The PF_ccc_0_0 IP is configured to take 50 MHz reference clock as an input and generates 25 MHz and 200 MHz output clocks, see the following figures.

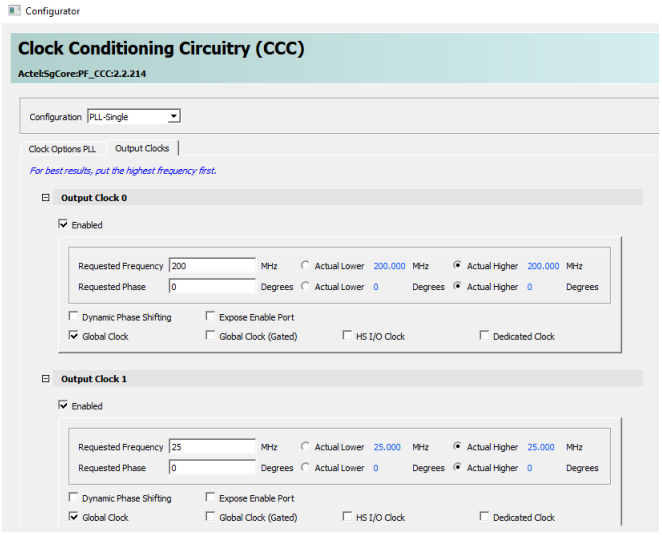
The following figure shows the input clock configuration:

Figure 1-7. Configuring PF_ccc_0_0 - Clock Options PLL



The following figure shows the output clock configuration.

Figure 1-8. Configuring PF_ccc_0_0 - Output Clocks

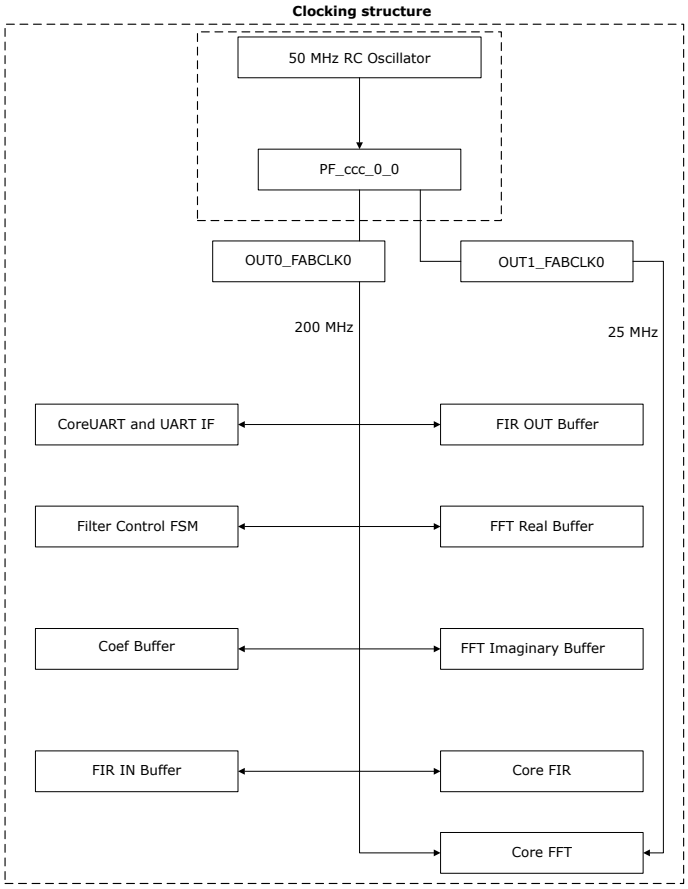


1.4 Clocking Structure

The demo design has only one clock domain. The on-board 50 MHz crystal oscillator connected to the PF_ccc_0_0 block generates 25 MHz and 200 MHz clocks that drive PF_COREUART_0_0 and PF_DSP_FLOW_DEMO_TOP_0 blocks.

The following figure shows the clocking structure in the demo design.

Figure 1-9. Clocking Structure

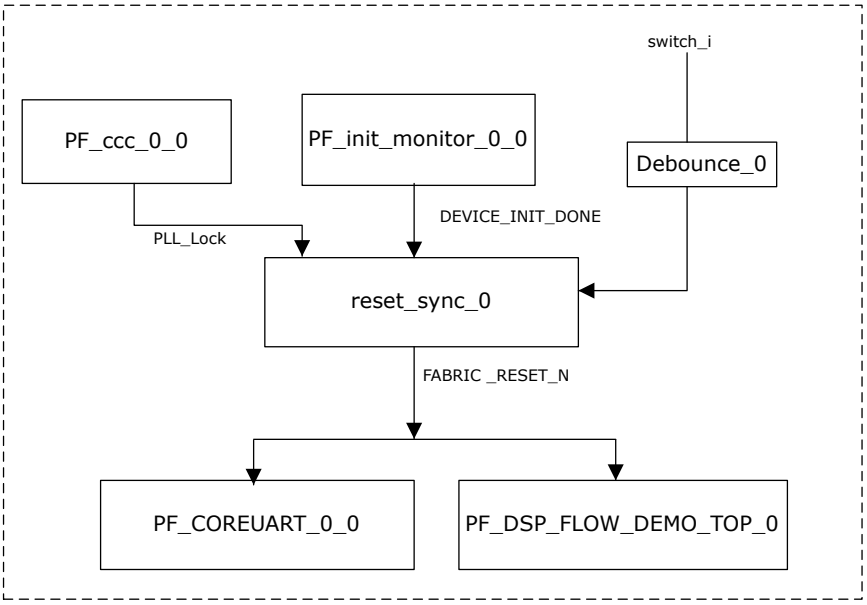


1.5 Reset Structure

The DEVICE_INIT_DONE, PLL Lock, and switch_i signals mapped to K22 (evaluation board) initiate the reset signal (FABRIC_RESET_N) from the reset_sync_0 block, which synchronizes with the Fabric Clock.

The following figure shows the reset structure in the demo design.

Figure 1-10. Reset Structure



1.6 Simulating the Design

Before you begin:

1. Start the Libero SoC software and from the menu bar, select **Project Tool Profiles**.
2. In the **Tool Profiles** window, select **Synthesis and Simulation** on the **Tools** pane and select the latest active installation directory paths for these two tools.
3. For the design to run, check-out the design file, execute the TCL scripts and to open the libero project, use the locations:
   - **Core:** `mpf_an4596_v2022p1_eval_df\TCL_scripts\CoreFIR\Libero_Project`
   - **RTL:** `mpf_an4596_v2022p1_eval_df\TCL_scripts\FIR_RTL\Libero_Project`
4. Click **Open**. The PolarFire DSP FLOW project opens in Libero SoC.
5. Open the **Design Hierarchy** tab and double-click the **TOP** component. The SmartDesign page opens in the right-side pane and displays the high-level design.
6. Double-click the **PF_DSP_FLOW_DEMO_TOP_0** component. The top-level Simulation design internal modules are displayed.
7. Download the following IP cores from **Libero SoC --> Catalog**:
   - PF_COREFIR_0
   - PF_COREFFT_0
   - PF_TPSRAM
   - PF_ccc_0_0
   - PF_COREUART_0_0

   A testbench is provided to simulate the design. The testbench simulates the filter pattern and waveform selection. It contains the test selection for the coefficient inputs (lowpass, bandpass, and bandstop) and data

input. It also monitors the UART_IF module status signals, output signals (DATAOUT), and FFT output status signals (DATA Valid and output ready) for the verification of filter output.

Figure 1-11. Testbench and DSP Demo Design Interaction

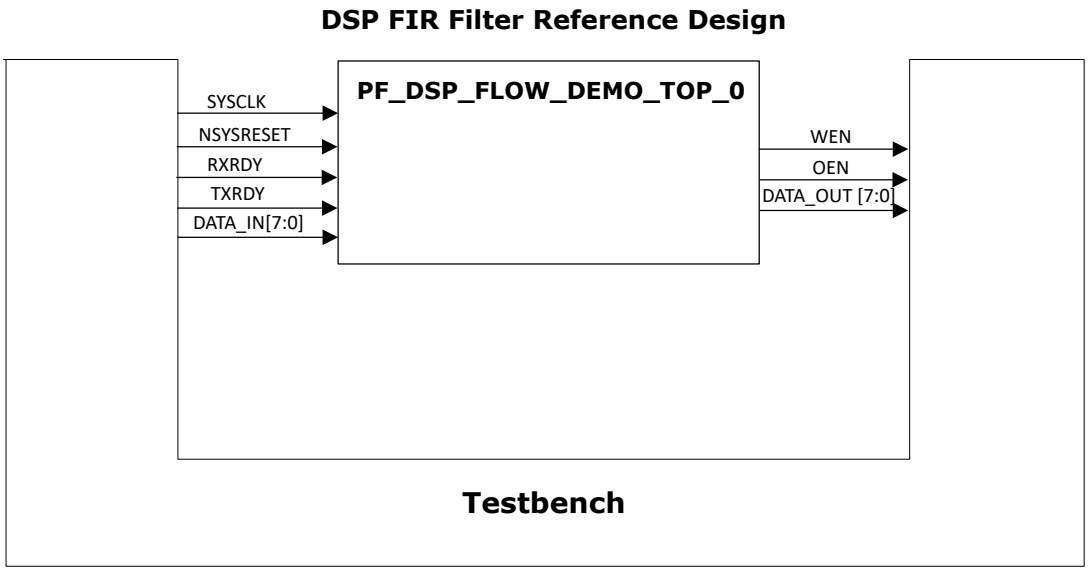


Table 1-4. Simulation Signals

Signals	Description
Input Signals	
SYSCLK	200 MHz generated clock
NSYSRESET	Active low reset signal
RXRDY	Receive ready
TXRDY	Transmit ready
DATA_IN[7:0]	8-bit input data (Handshake, Coefficient, and Data)
Output Signals	
WEN	Write enable
OEN	Output enable
DATA_OUT[7:0]	8-bit output data (Handshake, FIR Output, and FFT Output Data)

8. To simulate the low-pass, high-pass, band-pass, or the band-reject functionality, include the corresponding DAT file in the `testbench.v` file as shown in (Figure 1-12).
   The following DAT files are available for the respective filter functionality.
   - For low-pass, `coe_file_Low_Pass.dat`
   - For high-pass, `coe_file_High_pass.dat`
   - For band-pass, `coe_file_Band_pass.dat`
   - For band-stop, `coe_file_Band_reject.dat`

Figure 1-12. Pre-Synthesized Simulation

```
83  //Reading input data file
84  initial begin
85    data_file = $fopen("data_file.dat", "r");
86    if (data_file == `NULL) begin
87      $display("data_file handle was NULL");
88      $finish;
89    end
90  end
91  //COE File reading
92  initial begin
93    coe_data_file = $fopen("coe_file_Low_pass.dat", "r");
94    if (coe_data_file == `NULL) begin
95      $display("coe_file handle was NULL");
96      $finish;
97    end
98  end
99
```

Important: For more information about parameters used to generate coefficient files for all the filters, see `Readme.txt`.

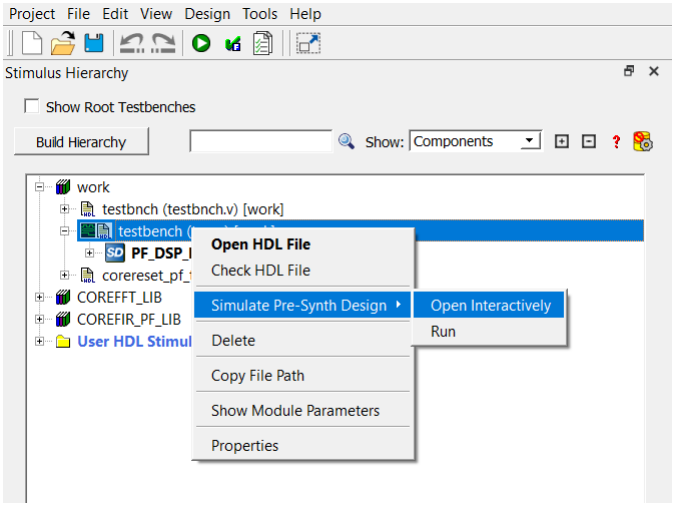
9. In the **Stimulus Hierarchy** tab, right click **testbench**, and select **Open Interactively** from **Simulate Pre-Synth Design**, as shown in the following figure.

The ModelSim tool completes the simulation in 3 minutes.

When the simulation is initiated, the ModelSim tool compiles all of the design source files, runs the simulation, and configures the waveform viewer to show the simulation signals.

When the simulation is successful, the success status is updated in the transcript tab in ModelSim.

Figure 1-13. Simulating the Pre-Synthesis Design



1.6.1 Simulation Flow

The following steps describe the testbench simulation flow:

At the start, the `NSYSREST` signal resets all of the components.

1. On initializing the UART_IF block, coefficient values are sent to the block when the receiver is high.

Important: UART communication channel is initialized by sending data “9” from test bench and checking whether the expected data “F” is sent from UART IF block, followed by “h” and “a” patterns.

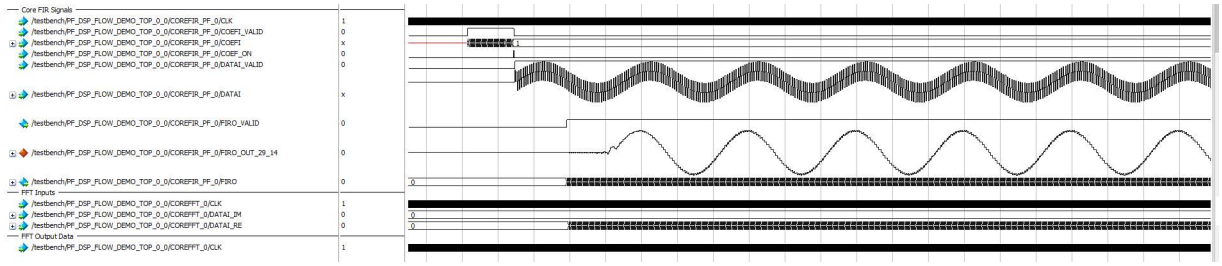
2. Filter input data values are sent to the design on handshaking pattern “r” received from the UART IF block, when the RX_READY signal is high.
3. When FFT output is ready, the mean value is calculated in the testbench with imaginary and real outputs.
4. When the FILTER_COMPLETE signal is received, the testbench issues the “**Test Completed successfully**” message indicating that the simulation was successful.

Filter coefficients are generated with the following parameters:

- Filter Type: Low Pass (Low pass/Band pass/Band stop filter)
- Filter Window: Blackman
- Low Cut-Off Frequency: Disabled for Low pass filter required
- High Cut-Off Frequency: 20 MHz (Band Pass/Reject Low Cut off frequency: 10 MHz, High Cut-off frequency: 20 MHz)
- Signal generation Input Frequency 1: 1 MHz
- Signal generation Input Frequency 1: 50 MHz
- Filter Taps: 31

Important: For more information about parameters used to generate coefficient files, see `Readme.txt`.

Figure 1-14. CoreFIR Input and Output Signals



Important: To get this waveform, perform the following steps:

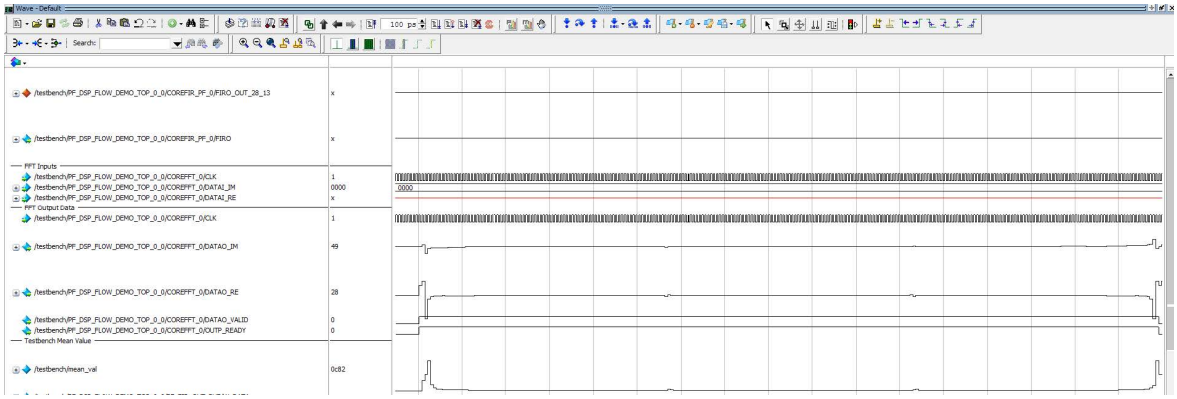
1. Change the radix of the input to decimal, right-click DATAI, go to radix option, and select decimal.
2. Right-click DATAI, select format option, and select Analog (automatic).

In the preceding figure:

- COEFFI represents Input coefficients
- DATAI represents FIR input data
- FIRO represents FIR output data

Important: Output data is valid when FIRO_VALID is high.

Figure 1-15. Low-Pass Filter Output



In the preceding figure:

- DATAO_IM represents the imaginary data output
- DATAO_RE represents the real part of FFT output

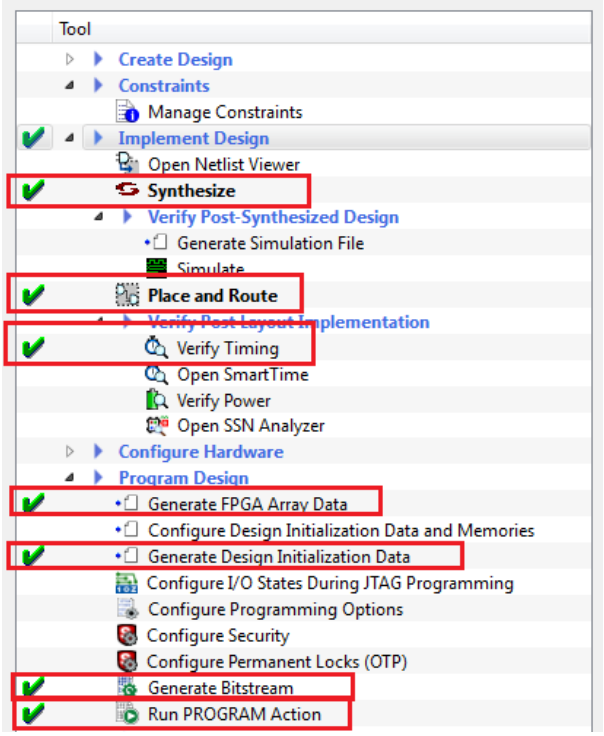
2. Libero Design Flow

The Libero design flow involves the following steps:

- 2.1. Synthesize
- 2.2. Place and Route
- 2.3. Verify Timing
- 2.4. Generate FPGA Array Data
- 2.5. Generate Bitstream
- 2.6. Run PROGRAM Action

The following figure shows these options in the **Design Flow** tab.

Figure 2-1. Libero Design Flow Options



2.1 Synthesize

To synthesize the design:

1. Double click **Synthesize** from the **Design Flow** tab. When the synthesis is successful, a green tick mark appears as shown in Figure 2-1.
2. Right click **Synthesize** and select **View Report** to view synthesis report and log files in the **Reports** tab.

2.2 Place and Route

To place and route the design:

From the **Design Flow** tab, double click **Place and Route**. When place and route is successful, a green tick mark appears as shown in Figure 2-1.

2.2.1 Resource Utilization

The DSP interface design is implemented on the PolarFire FPGA device (MPF300TS-1FCG1152I package). The following table lists the resource utilization report after Place and Route process.

Important: The resource utilization might vary slightly based on different runs.

Table 2-1. DSP FIR Filter Demo Resource Usage Summary for the Design with Core FIR

Type	Used	Total	Percentage
4LUT	5012	299544	1.67
DFF	6068	299544	2.03
I/O Register	0	1536	0.00
Logic Element	6789	299544	2.27
µSRAM	50	2772	1.66
LSRAM	5	952	0.53
MATH	68	924	7.36

Table 2-2. DSP FIR Filter Demo Resource Usage Summary for the Design with RTL Inference

Type	Used	Total	Percentage
4LUT	4942	299544	1.65
DFF	7697	299544	2.57
I/O Register	0	1536	0.00
Logic Element	8404	299544	2.81
µSRAM	42	2772	1.52
LSRAM	5	952	0.53
MATH	68	924	7.36

The following tables lists MathBlock usage summary.

Table 2-3. MathBlocks Usage Summary for the Design with Core FIR

CoreFIR	CoreFFT	Total
64	4	68

Table 2-4. MathBlocks Usage Summary for the Design with RTL Inference

FIR RTL	CoreFFT	Total
64	4	68

The following tables lists RAM block usage summary.

Table 2-5. RAM Blocks Usage Summary for the Design with Core FIR

RAM Type	CoreFIR	CoreFFT	Fabric Buffers	Total
µSRAM	4	42	0	46
LSRAM	0	0	5	5
Total	4	42	5	51

Table 2-6. RAM Blocks Usage Summary for the Design with RTL Inference

RAM Type	FIR RTL	CoreFFT	Fabric Buffers	Total
µSRAM	0	42	0	42
LSRAM	0	0	5	5
Total	0	42	5	47

2.3 Verify Timing

To verify timing:

1. Double click **Verify Timing** from the **Design Flow** tab. When the design successfully meets the timing requirements, a green tick mark appears as shown in Figure 2-1.
2. Right click **Verify Timing** and select **View Report** to view the verify timing report and log files in the **Reports** tab.

2.4 Generate FPGA Array Data

To generate FPGA array data, double-click **Generate FPGA Array Data** from the **Design Flow** tab. A green tick mark is displayed after the successful generation of the FPGA array data as shown in Figure 2-1.

2.5 Generate Bitstream

To generate the bitstream:

1. Double click **Generate Bitstream** from the **Design Flow** tab. When the bitstream is successfully generated, a green tick mark appears as shown in Figure 2-1.
2. Right click **Generate Bitstream** and select **View Report** to view the corresponding log file in the **Reports** tab.

2.6 Run PROGRAM Action

After generating bitstream, a PolarFire device must be programmed. Follow these steps to program the PolarFire device:

1. Ensure that the jumper settings on board are the same as those listed in the following table.

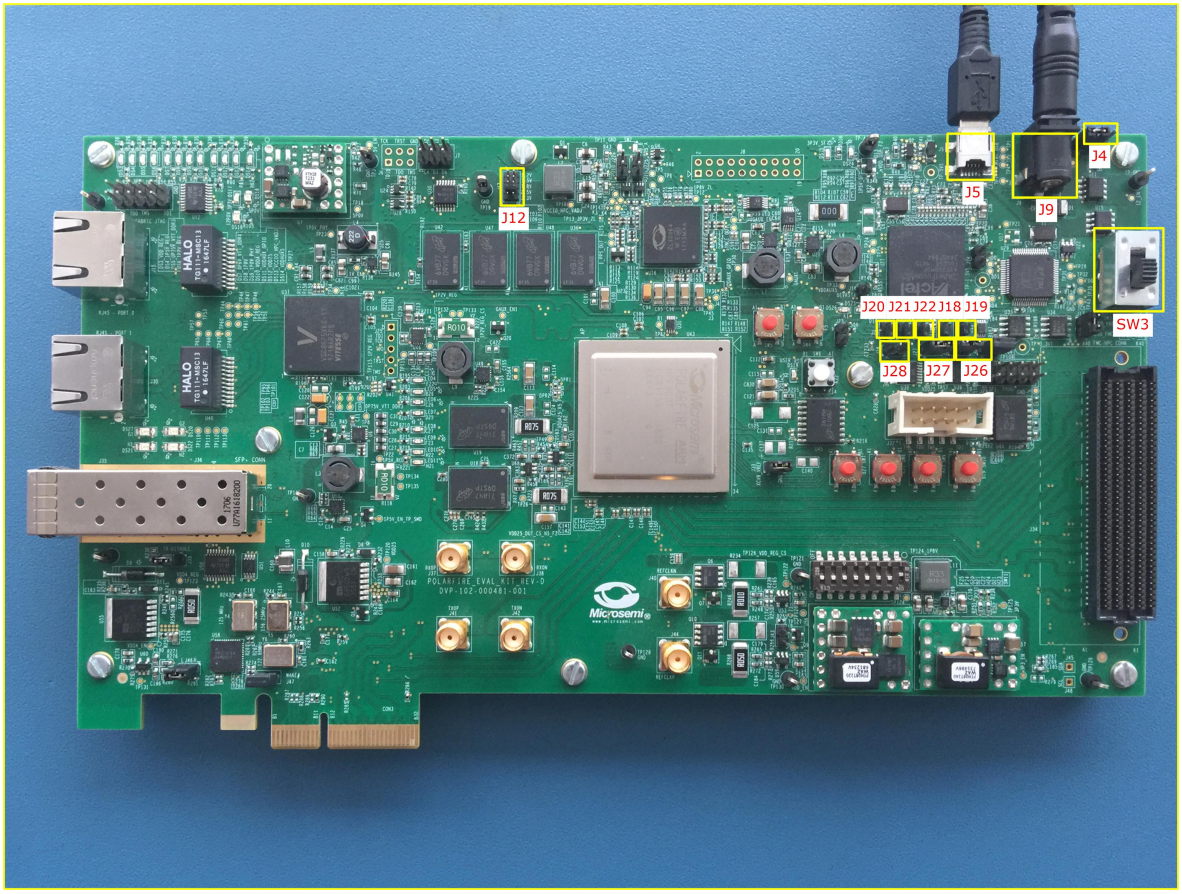
Table 2-7. Jumper Settings

Jumper	Description
J18, J19, J20, J21, and J22	Close pin 2 and 3 for programming the PolarFire FPGA through FTDI
J28	Close pin 2 and 3 for programming through the on-board FlashPro5
J26	Close pin 1 and 2 for programming through the FTDI SPI
J27	Close pin 1 and 2 for programming through the FTDI SPI

.....continued	
Jumper	**Description**
J4	Close pin 1 and 2 for manual power switching using SW3
J12	Close pin 3 and 4 for 2.5V

2. Connect the power supply cable to the **J9** connector on the board.
3. Connect the USB cable from the host PC to the **J5** (FTDI port) on the board.
4. Power on the board using the **SW3** slide switch.

Figure 2-2. Board Setup



5. Double-click **Run PROGRAM Action** from the **Libero** > **Design Flow** tab.
6. Right click **Run Program Action** and select **View Report** to view the corresponding log file in the **Reports** tab.

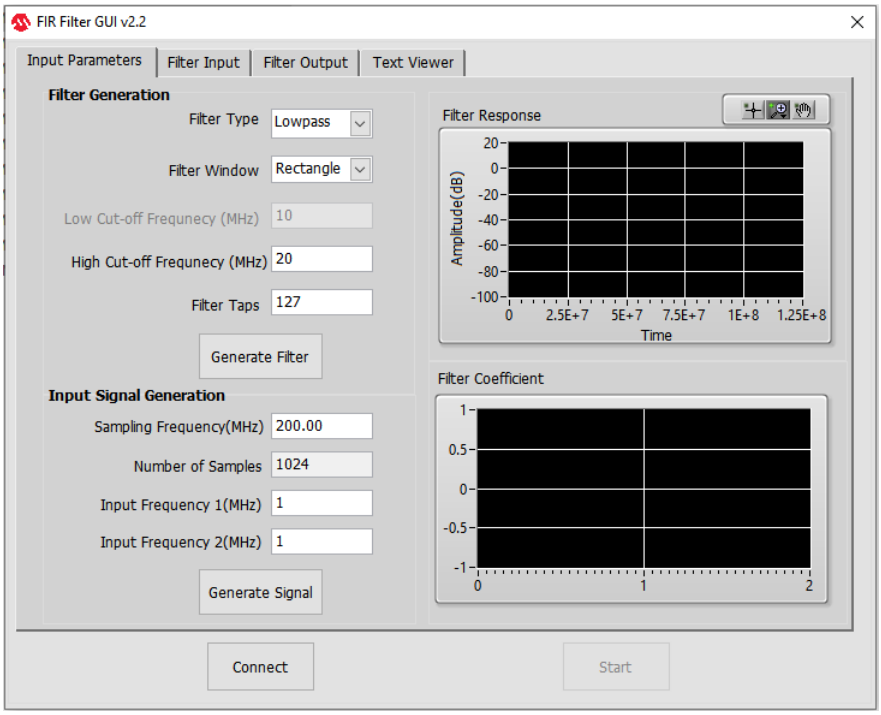
When the device is successfully programmed, a green tick mark appears as shown in Figure 2-1. See, 4. Running the Demo to run the DSP FIR filter demo.

3. DSP FIR Demo GUI

The FIR Filter GUI application runs on the host PC connected to the PolarFire Evaluation Kit. UART is used as the communication protocol between the host PC and the PolarFire Evaluation Kit.

The following figure shows the DSP FIR Filter GUI.

Figure 3-1. FIR Filter GUI



The DSP FIR demo window consists of the following tabs:

- **Input Parameters**: Configures the filter generation and signal generation
- **Filter Input**: Plots the input signal and its frequency spectrum
- **Filter Output**: Plots the output signal and its frequency spectrum
- **Text Viewer**: Shows the coefficients, input signal, output signal, and FFT data values

4. Running the Demo

Running the demo involves the following steps:

1. 4.1. Installing and Starting the GUI
2. 4.2. Generating the Filter Coefficients and Input Signal
3. 4.3. Generating the Filter Output

Important: The steps to run the demo are the same for the CoreFIR IP design and the FIR RTL inference design. The following sections take the CoreFIR IP design as an example.

4.1 Installing and Starting the GUI

To install and start the GUI, perform the following steps:

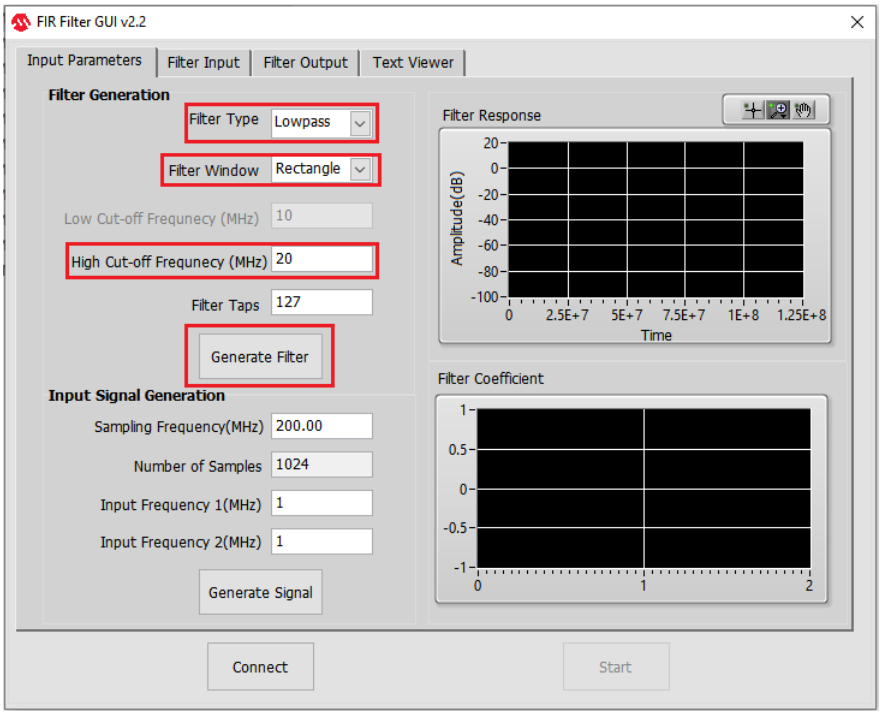
1. Double-click the DSP FIR Demo GUI application (`setup.exe`) from the following design files folder: `mpf_an4596_v2022p1_eval_df\GUI\setup.exe`
2. Follow the installation wizard to install the GUI application.
3. Double-click the `FIR_Filter_GUI.exe` application from the installation directory to start the GUI application. The FIR Filter GUI window is displayed as shown in Figure 3-1.

4.2 Generating the Filter Coefficients and Input Signal

Follow these steps:

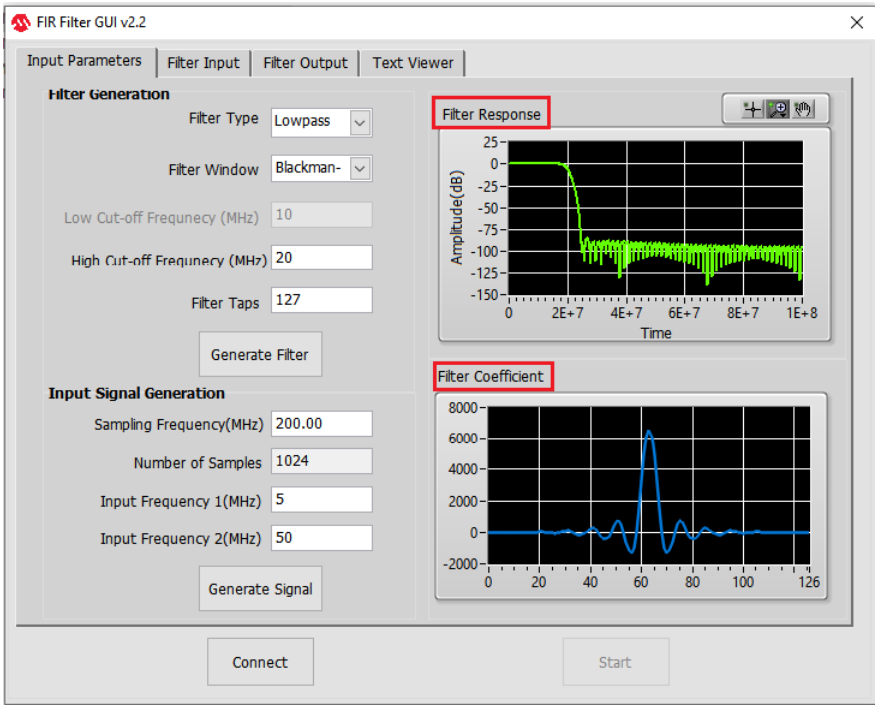
1. To generate the filter coefficients, set the following parameters in the GUI and click **Generate Filter**, as shown in Figure 4-1:
   - **Filter Type**: Low pass (Lowpass, Highpass, Bandpass, andd Bandstop filter)
   - **Filter Window**: Blackman-Harris (Blackman-Harris, Blackman, Hamming, Hanning, Rectangle, Flat Space, and Kaiser window)
   - **Low Cut-off Frequency (MHz)**: 10 (disabled for Low pass filter required)
   - **High Cut-off Frequency (MHz)**: 20
   - **Filter Taps**: 127 (Fixed)

Figure 4-1. Filter Generation



2. After generating the filter coefficients, the **Filter Response** and the Filter Coefficient plots are displayed as shown in the following figure.

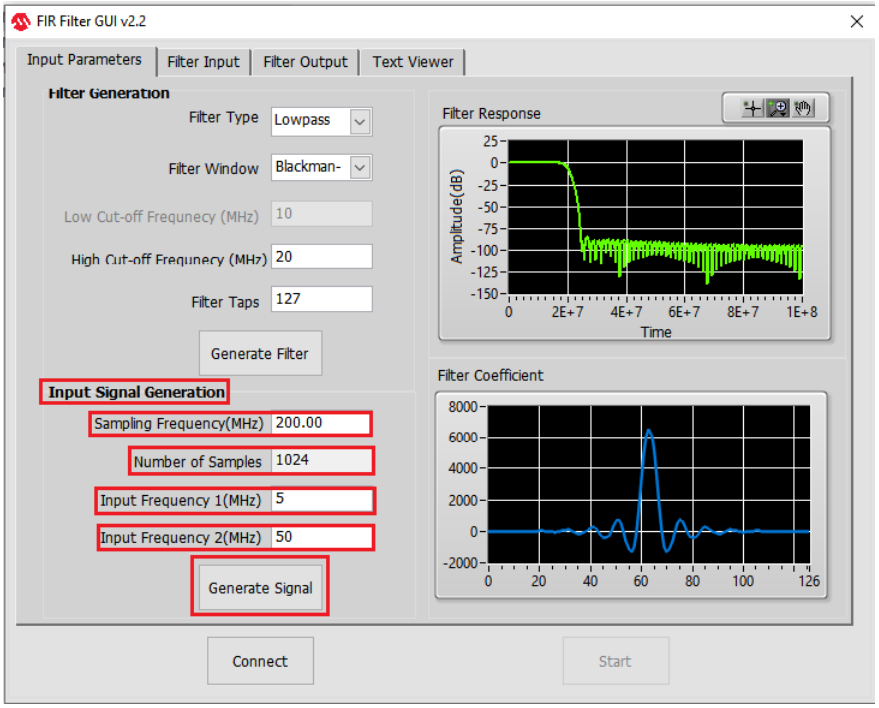
Figure 4-2. The Filter Response and Filter Coefficient Plot



3. To generate the inputs signals, set the following parameters in the GUI and click **Generate Signal** as shown in Figure 4-3:
   - **Sampling Frequency (MHz)**: 200 (Fixed)
   - **Number of Samples**: 1024 (Fixed)

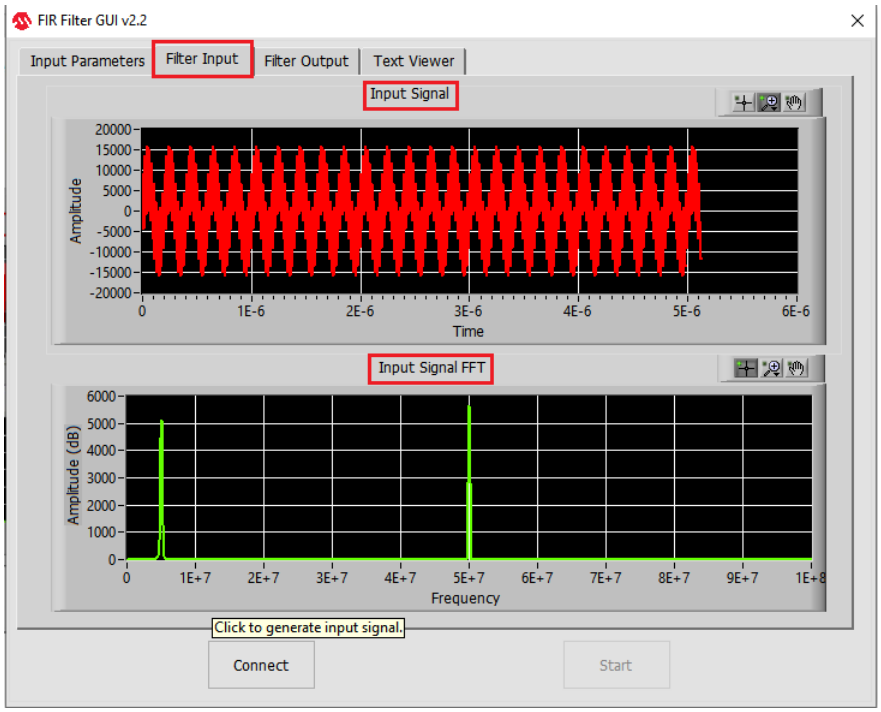
– **Input Frequency 1 (MHz)**: Enter the signal frequency in the passband region. For example, 5 MHz for high cut-off frequency
– **Input Frequency 2 (MHz)**: Enter the signal frequency in the stopband region. The stopband frequency is generally set to a value less than the half of the sampling frequency. For example, 50 MHz.

Figure 4-3. Input Signal Generation



4. The input signals and the frequency spectrum of the specified signals are displayed in the **Filter Input** tab, as shown in the following figure.

Figure 4-4. Input Signal and Input Signal FFT Plot

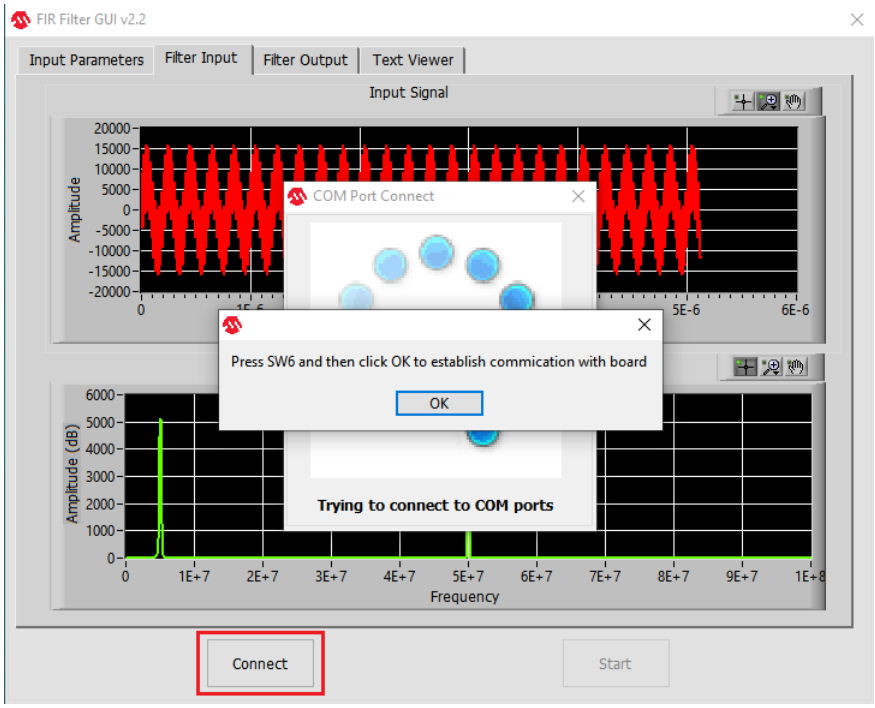


4.3 Generating the Filter Output

To generate the filter output, perform the following steps:

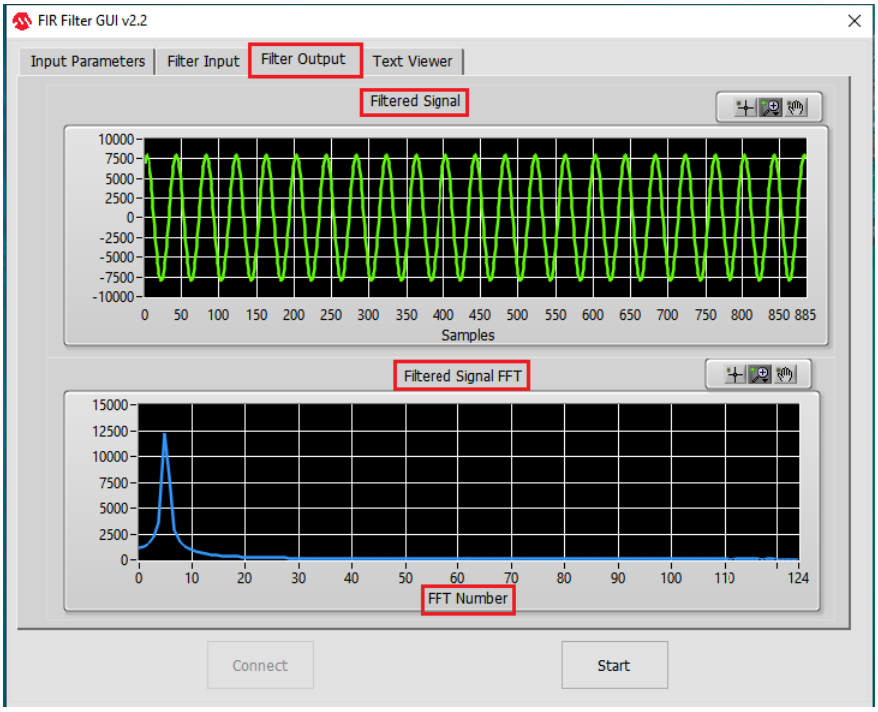
1. To configure the input frequencies and coefficients, click **Connect** as shown in following figure. After establishing the COM port connection click **Start**. The GUI prompts to press the on-board **SW6** and click **OK**, as shown in the following figure.
2. The GUI application sends the input data (1K samples) and the filter coefficients to the PolarFire device to process the filtering operation.

Figure 4-5. UART Connection



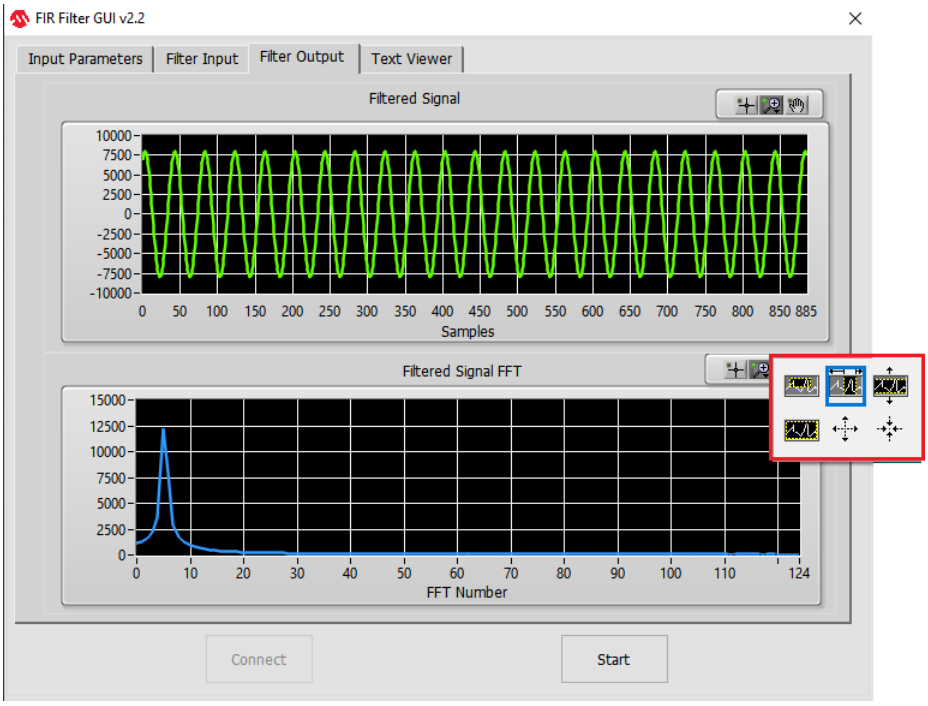
After completing the filter operation, the GUI displays the "**Operation Completed**" message and plots the filtered data and the FFT data on the **Filter Output** tab, as shown in the following figure. As the Low pass filter option was selected, the high-frequency component is suppressed while the low-frequency signal is preserved. This can be observed in the frequency spectrum of the output signal.

Figure 4-6. Filter Output



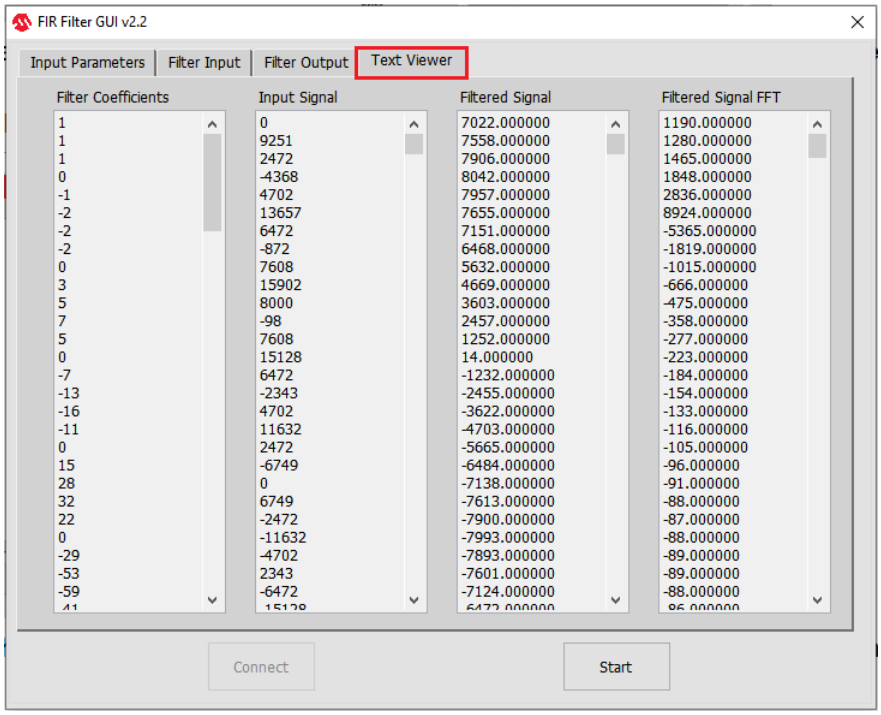
Waveform can be zoomed in or out by using these options, as shown in the following image.

Figure 4-7. Zooming the Filter Output



3. The filter coefficients, input signal, output signal, and FFT output data values can be viewed in the **Text Viewer**, as shown in the following figure.

Figure 4-8. The Text Viewer Tab



4. To save the coefficients, select the text in **Filter Coefficients** pane, copy and paste in the required location.
5. Close the **GUI**.

This concludes the DSP FIR Filter demo.

5. Appendix 1: Programming the Device Using FlashPro Express

This section describes how to program the PolarFire device with the job programming file using a FlashPro programmer. The default location of the job files is:

```
mpf_an4596_v2022p1_eval_df\Programming_Job\CoreFIR
```

and

```
mpf_an4596_v2022p1_eval_df\Programming_Job\FIR_RTL
```

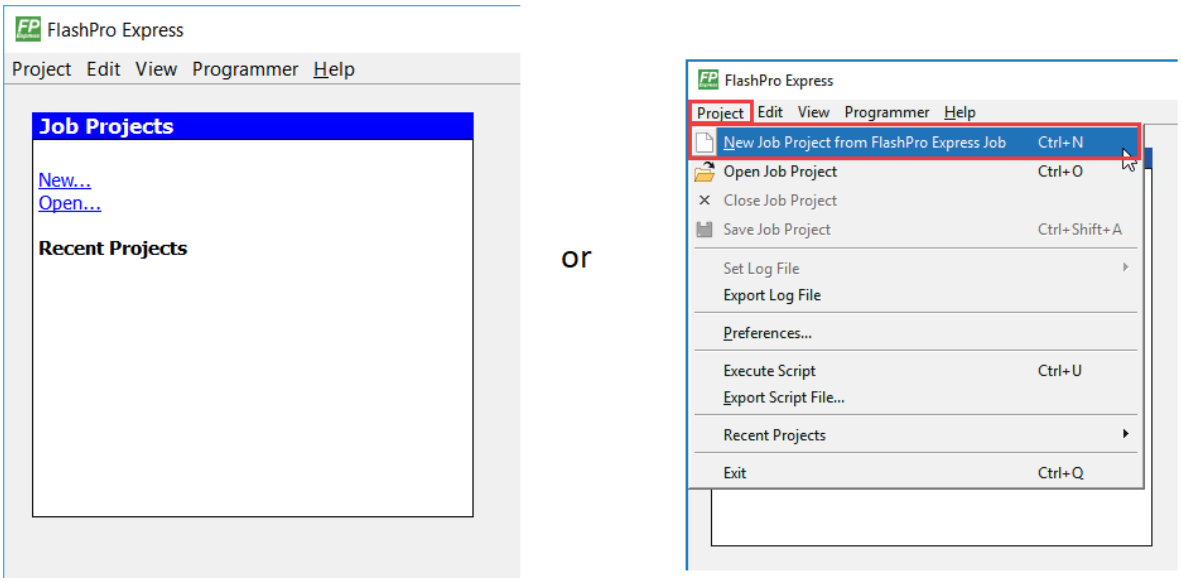
To program a PolarFire device using FlashPro Express, complete the following steps:

1. Ensure that the jumper settings on the board are the same as listed in Table 2-7.

Important: The power supply switch must be switched-off while making the jumper connections.

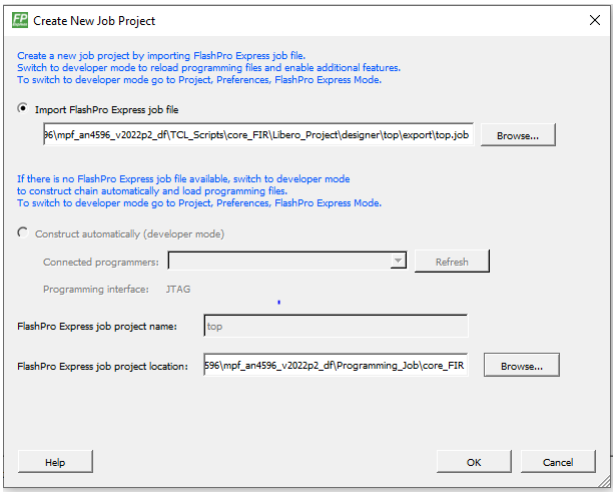
2. Connect the power supply cable to the **J9** connector on the board.
3. Connect the USB cable from the Host PC to **J5** (FTDI port) on the board.
4. Power on the board using the **SW3** slide switch.
5. On the host PC, launch the FlashPro Express software.
6. Click **New** or select **New Job Project** from FlashPro Express Job from **Project** menu to create a new job project, as shown in the following figure.

Figure 5-1. FlashPro Express Job Project



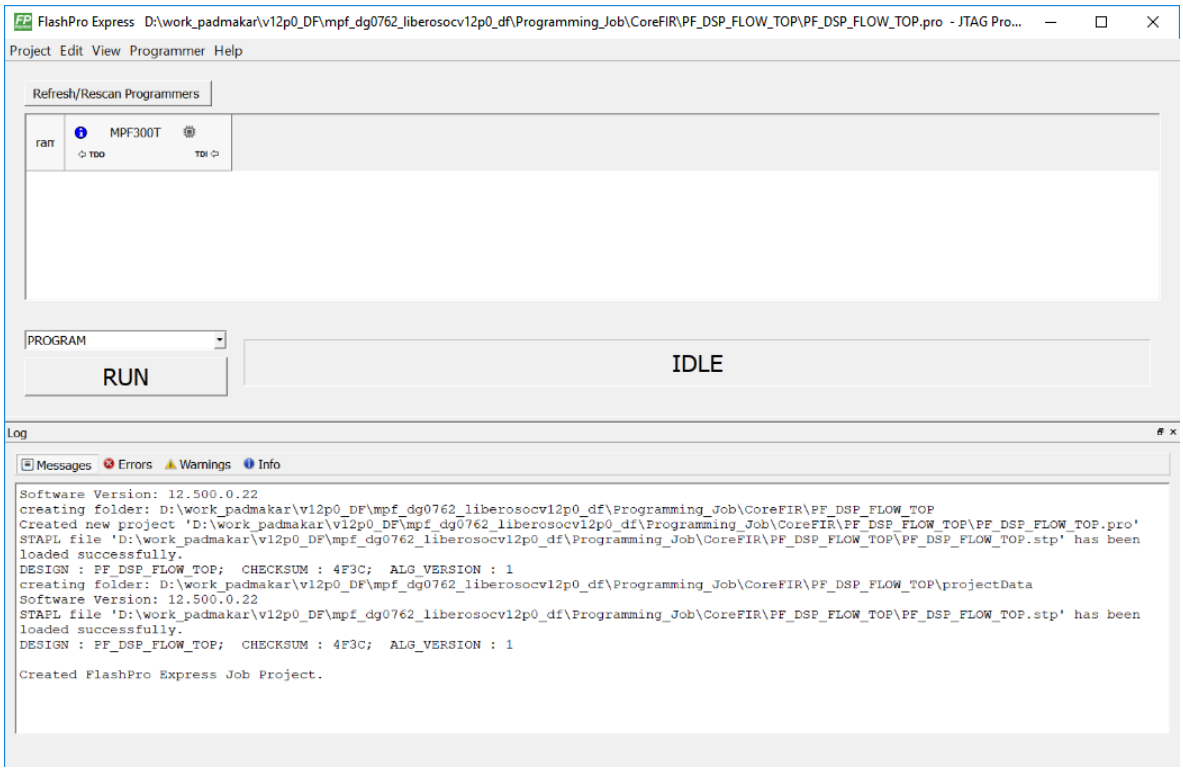
7. Enter the following in the New Job Project from FlashPro Express Job dialog box:
   - Programming job file: Click **Browse**, navigate to the location where the `.job` file is located, and select the file. The default location is: `<download_folder>\mpf_an4596_v2022p1_eval_df\Programming_Job`.
   - FlashPro Express job project location: Click **Browse** and navigate to the location where you want to save the project.

Figure 5-2. Create New Job Project



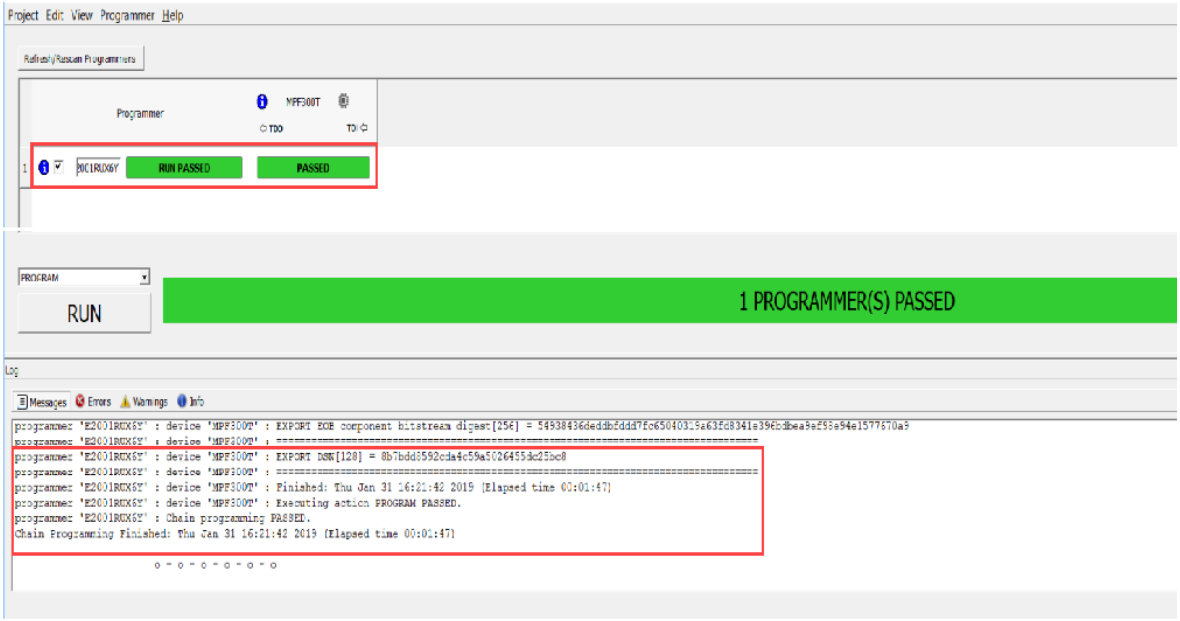
8. Click **OK**. The required programming file is selected and ready to be programmed in the device. The FlashProExpress window appears.
9. Verify that a programmer number appears in the **Programmer** box. If it does not show, verify the board connections and click **Refresh/Rescan Programmers**.

Figure 5-3. Programming the Device



10. Click **RUN** to program the device. When the device is programmed successfully, a RUN PASSED status is displayed as shown in the following figure.

Figure 5-4. FlashPro Express—RUN PASSED



11. Close **FlashPro Express** or in the **Project** tab, click **Exit**.

6. Appendix 2: Running the TCL Script

TCL scripts are provided in the folders of design files under directory TCL_Scripts. If required, the design flow can be reproduced from Design Implementation till generation of job file.

To run the TCL, follow the steps below:

1. Launch the Libero SoC software.
2. Select **Project** > **Execute Script**
3. Click **Browse** and select `script.tcl` from the downloaded `TCL_Scripts` directory.
4. Click **Run**.

After a successful execution of the TCL script, a Libero project is created within the `TCL_Scripts` directory.

For more information about TCL scripts, see:

- `mpf_an4596_v2022p1_eval_df/TCL_Scripts/core_FIR/readme.txt`
- `mpf_an4596_v2022p1_eval_df/TCL_Scripts/FIR_RTL/readme.txt`

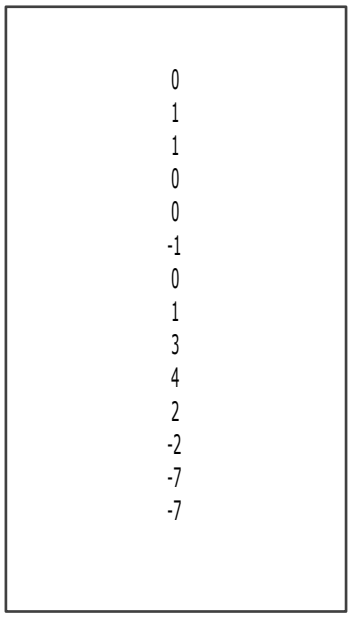
See Tcl Commands Reference Guide for more details on TCL commands. Contact Technical Support team for any queries encountered while running the TCL script.

7. Appendix 3: Coefficient Test File Format

Perform the following steps to create the coefficient file:

1. The FIR filter coefficients can be loaded from an ASCII text file (`*.txt`). Create the coefficient file using a text editor.
2. The format of the text file must be as shown in the following figure. The coefficient values are entered as integers. For asymmetric or anti-symmetric filter, only half of the coefficients must be listed in the file (this applies to the fully enumerated type only). Only one coefficient value per line is permitted. An empty line must be placed after the last coefficient of the last set.

Figure 7-1. Coefficient File Example - 9 Taps, Decimal Values



8. Appendix 4: References

This section lists documents that provide more information about the DSP filters and IP cores used in the design.

- For more information about PF_TPSRAM, see PolarFire FPGA and PolarFire SoC FPGA Fabric User Guide.
- For more information about PF_CCC, see PolarFire FPGA and PolarFire SoC FPGA Clocking Resources User Guide.
- For more information about CoreFIR, see CoreFIR User Guide.
- For more information about CoreFFT, see CoreFFT User Guide.
- For more information about CoreUART, see CoreUART User Guide.
- For more information about Libero, ModelSim, and Synplify Pro, see the Libero SoC Documentation.

9. Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the current publication.

Table 9-1. Revision History

Revision	Date	Description
A	11/2022	The following is the list of changes made in revision A of the document: • The document was migrated to the Microchip template. • The document number was updated to DS00004596 from 50200762. • The document ID was updated to AN4596 from DG0762. • Updated the design files, TCL script, and `.job` file throughout the document. • Updated Figure 1-2 and Figure 1-3. • Updated Figure 1-7, Figure 1-8, and Figure 1-9. • Updated Figure 3-1, Figure 4-1, Figure 4-2, Figure 4-3, and Figure 4-4. • Updated Figure 4-5, Figure 4-6, and Figure 4-7. • Added a new clock signal in 1.4. Clocking Structure.
10.0	—	Added 6. Appendix 2: Running the TCL Script.
9.0	—	The following is a summary of the changes made in this revision: • Updated the document for Libero® SoC v12.2. • Removed the references to Libero version numbers.
8.0	—	Updated the document for Libero® SoC v12.0.
7.0	—	Updated the document for Libero® SoC PolarFire® v2.3.
6.0	—	Updated the document for Libero® SoC PolarFire v2.2.
5.0	—	The following is a summary of the changes made in this revision. • The document was updated for Libero SoC PolarFire v2.1. • Updated 4. Running the Demo.
4.0	—	The following is a summary of the changes made in this revision: • The document was updated for Libero SoC PolarFire v2.0. • Updated the GUI screens in 4. Running the Demo.
3.0	—	The following is a summary of the changes made in this revision: • The GUI images were updated.
2.0	—	The following is a summary of the changes made in this revision: • The document was updated to include features and enhancements introduced in the Libero SoC PolarFire v1.1 SP1 release.
1.0	—	Revision 1.0 was the first publication of this document.

Microchip FPGA Support

Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select appropriate case category, and upload design files while creating a technical support case.

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

- From North America, call **800.262.1060**
- From the rest of the world, call **650.318.4460**
- Fax, from anywhere in the world, **650.318.8044**

Microchip Information

The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip design partner program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Product Change Notification Service

Microchip's product change notification service helps keep customers current on Microchip products. Subscribers will receive email notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip product is strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP "AS IS". MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Trademarks

The Microchip name and logo, the Microchip logo, Adaptec, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, CryptoMemory, CryptoRF, dsPIC, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

AgileSwitch, APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, Flashtec, Hyper Speed Control, HyperLight Load, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet- Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, TrueTime, and ZL are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, Augmented Switching, BlueSky, BodyCom, Clockstudio, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, Espresso T1S, EtherGREEN, GridTime, IdealBridge, In-Circuit Serial Programming, ICSP, INICnet, Intelligent Paralleling, IntelliMOS, Inter-Chip Connectivity, JitterBlocker, Knob-on-Display, KoD, maxCrypto, maxView, memBrain, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, RTAX, RTG4, SAM-

ISBN: 978-1-6683-1488-3

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: www.microchip.com/support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Australia - Sydney** Tel: 61-2-9868-6733 **China - Beijing** Tel: 86-10-8569-7000 **China - Chengdu** Tel: 86-28-8665-5511 **China - Chongqing** Tel: 86-23-8980-9588 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 **China - Hong Kong SAR** Tel: 852-2943-5100 **China - Nanjing** Tel: 86-25-8473-2460 **China - Qingdao** Tel: 86-532-8502-7355 **China - Shanghai** Tel: 86-21-3326-8000 **China - Shenyang** Tel: 86-24-2334-2829 **China - Shenzhen** Tel: 86-755-8864-2200 **China - Suzhou** Tel: 86-186-6233-1526 **China - Wuhan** Tel: 86-27-5980-5300 **China - Xian** Tel: 86-29-8833-7252 **China - Xiamen** Tel: 86-592-2388138 **China - Zhuhai** Tel: 86-756-3210040	**India - Bangalore** Tel: 91-80-3090-4444 **India - New Delhi** Tel: 91-11-4160-8631 **India - Pune** Tel: 91-20-4121-0141 **Japan - Osaka** Tel: 81-6-6152-7160 **Japan - Tokyo** Tel: 81-3-6880- 3770 **Korea - Daegu** Tel: 82-53-744-4301 **Korea - Seoul** Tel: 82-2-554-7200 **Malaysia - Kuala Lumpur** Tel: 60-3-7651-7906 **Malaysia - Penang** Tel: 60-4-227-8870 **Philippines - Manila** Tel: 63-2-634-9065 **Singapore** Tel: 65-6334-8870 **Taiwan - Hsin Chu** Tel: 886-3-577-8366 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 **Thailand - Bangkok** Tel: 66-2-694-1351 **Vietnam - Ho Chi Minh** Tel: 84-28-5448-2100	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4485-5910 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-72400 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-72884388 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820