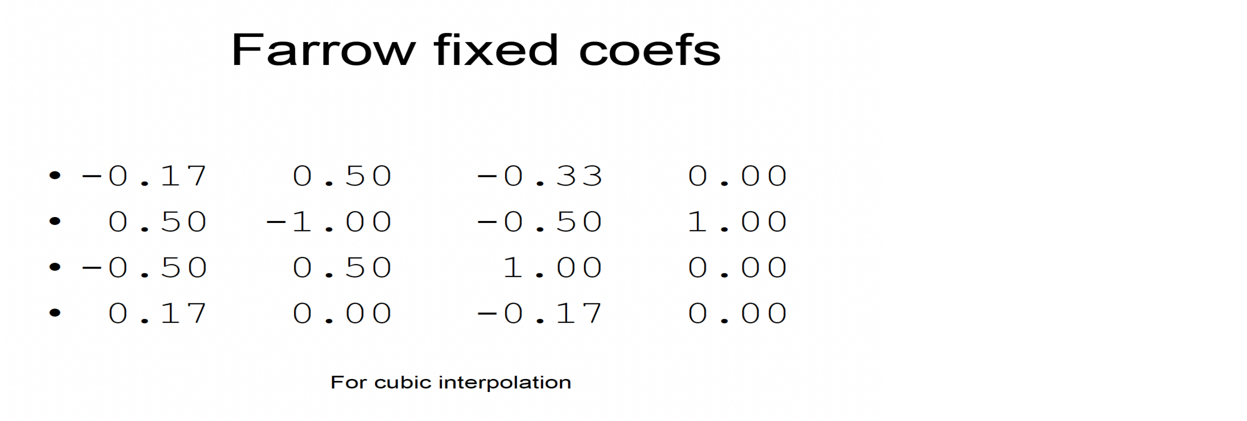
In this assignment, I designed a resampling fractional delay FIR filter using the farrow structure and emulated it on Arria || GX FPGA Platform (Quartus Prime) . The FIR fractional delay filter is a linear phase allpass interpolation filter. It generates a reconstructed signal shifted by some fractional displacement in time with respect to the original signal. Its impulse response is a time shifted discrete sinc function. Fractional delay FIR filters are typically found in synchronization of digital modems where the delay parameter varies over time. In order to compute the output of a fractional delay filter, we need to interpolate the signal in between the existing discrete-time samples. The Farrow structure is a hardware friendly implementation that allows us to interpolate the signal using polynomial coefficients and a bank of FIR filters. We use cubic interpolation to reconstruct the signal.



Farrow fixed coefs

- −0.17 0.50 −0.33 0.00
- 0.50 −1.00 −0.50 1.00
- −0.50 0.50 1.00 0.00
- 0.17 0.00 −0.17 0.00

For cubic interpolation

Fig 1 : Farrow filter coefficients for cubic interpolation

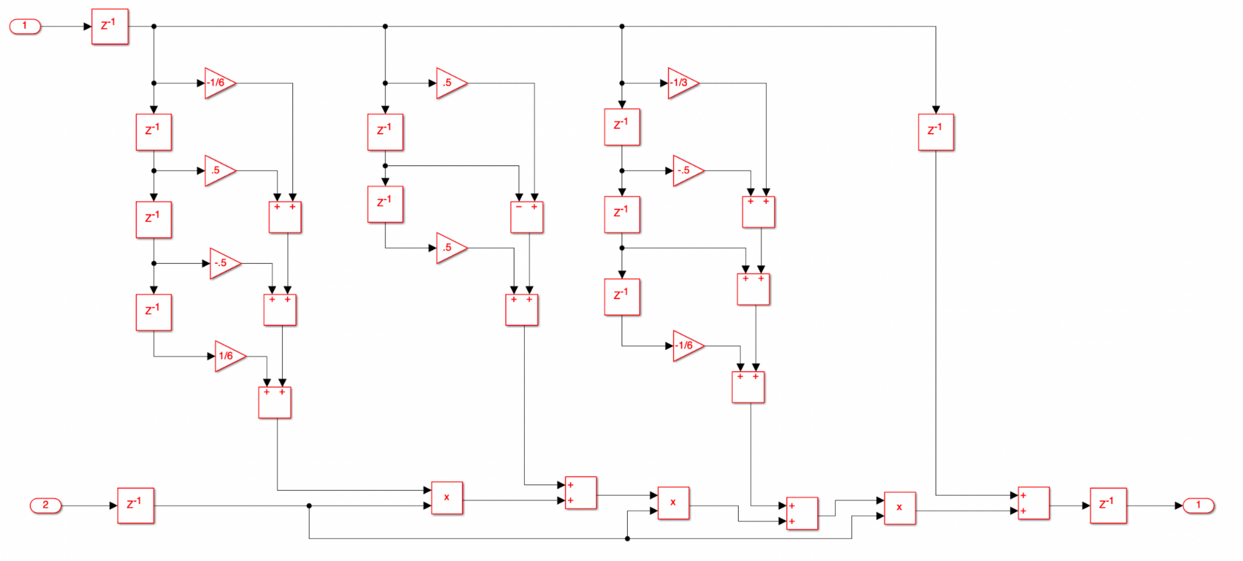


Fig 2. Canonical Farrow filter

I took the existing simulink model for the canonical Farrow filter, and modified the design to improve performance. The baseline design has a maximum frequency of 28 MHz at 900mV slow corner 100C. The critical path contains 4 multipliers and 6 adders (from the D flip flop just after the input port 1, to the D flip flop just before the output port 1 in figure 1). To reduce the critical path delay, I transformed the canonical FIR filter representation to a **transposed FIR filter** representation and introduced **pipeline stages by latching the outputs of each FIR filter**. I introduced further **pipeline stages in the bottom row where we compute the linear combination of the FIR filter outputs** to produce the final resampled output. As a consequence of the aggressive pipelining, the final output has an additional latency of 4 cycles. The new critical path includes a multiplier and an adder. I did not replace the fixed coefficient DSP multipliers with LUT based shift and add operations. While we can improve performance on the 0.5 multiplier using a simple right shift operation, I couldn't come up with a faster implementation for the 0.33 and 0.17 multipliers using shift operations. Hence, the critical path would always include a DSP multiplier in my design, and we wouldn't observe significant improvements if we only changed some of the multipliers selectively; changing all the multipliers using shift operations potentially could lead to a speed up. The input signal and fractional displacement signal have a precision of 16 bits (2's complement signed data) with 14 fractional bits. The maximum frequency for the modified Farrow filter is 101.56 MHz at 900 mV slow corner 100C.

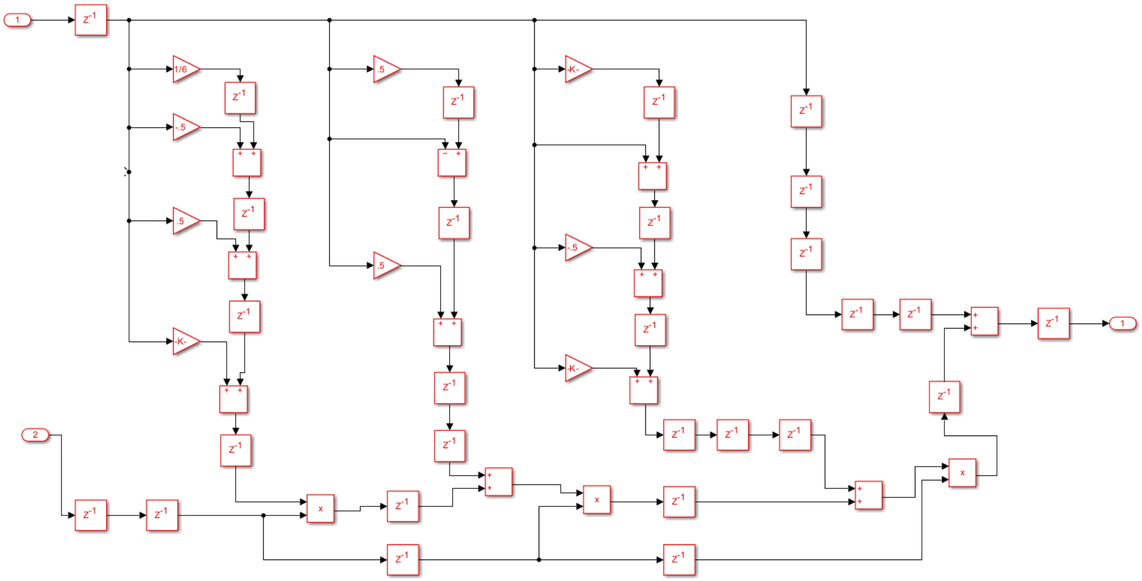


Fig 3: Farrow filter with transposed FIR filter and pipelining

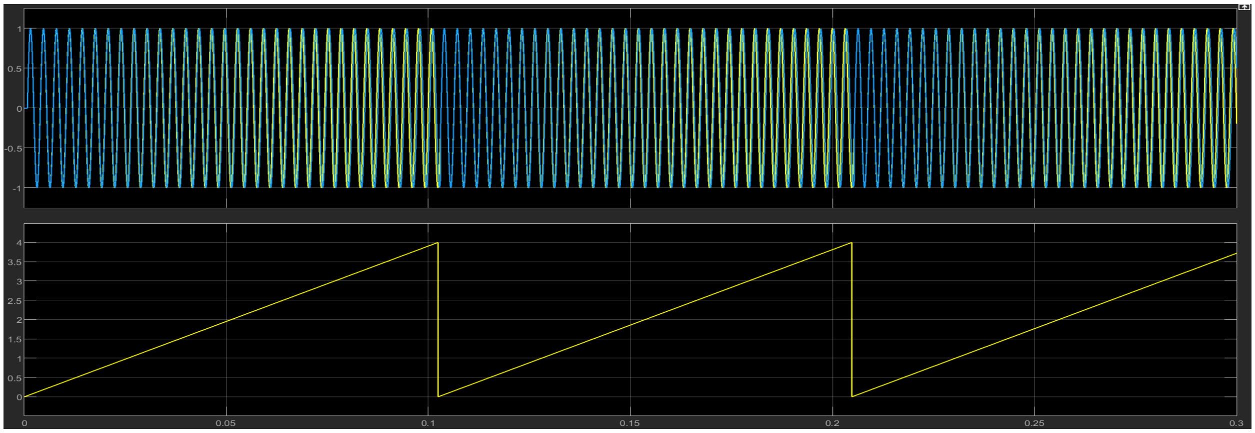
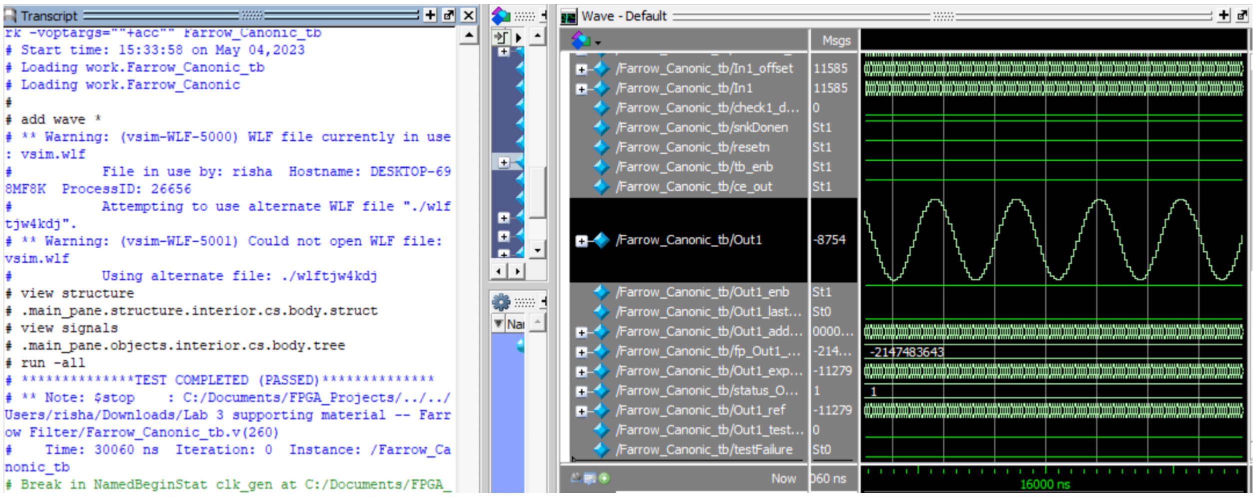


Fig 4 : Farrow filter output for a sinusoidal input signal and a ramp fractional displacement signal

Modelsim Simulation results:



```
rk -voptargs=""+acc"" Farrow_Canonic_tb
# Start time: 15:33:58 on May 04,2023
# Loading work.Farrow_Canonic_tb
# Loading work.Farrow_Canonic
#
# add wave *
# ** Warning: (vsim-WLF-5000) WLF file currently in use
: vsim.wlf
#           File in use by: risha  Hostname: DESKTOP-69
8MF8K  ProcessID: 26656
#           Attempting to use alternate WLF file "./wlf
tjw4kdj".
# ** Warning: (vsim-WLF-5001) Could not open WLF file:
vsim.wlf
#           Using alternate file: ./wlftjw4kdj
# view structure
# .main_pane.structure.interior.cs.body.struct
# view signals
# .main_pane.objects.interior.cs.body.tree
# run -all
# **************TEST COMPLETED (PASSED)**************
# ** Note: $stop    : C:/Documents/FPGA_Projects/../../
Users/risha/Downloads/Lab 3 supporting material -- Farr
ow Filter/Farrow_Canonic_tb.v(260)
#    Time: 30060 ns  Iteration: 0  Instance: /Farrow_Ca
nonic_tb
# Break in NamedBeginStat clk_gen at C:/Documents/FPGA_
```

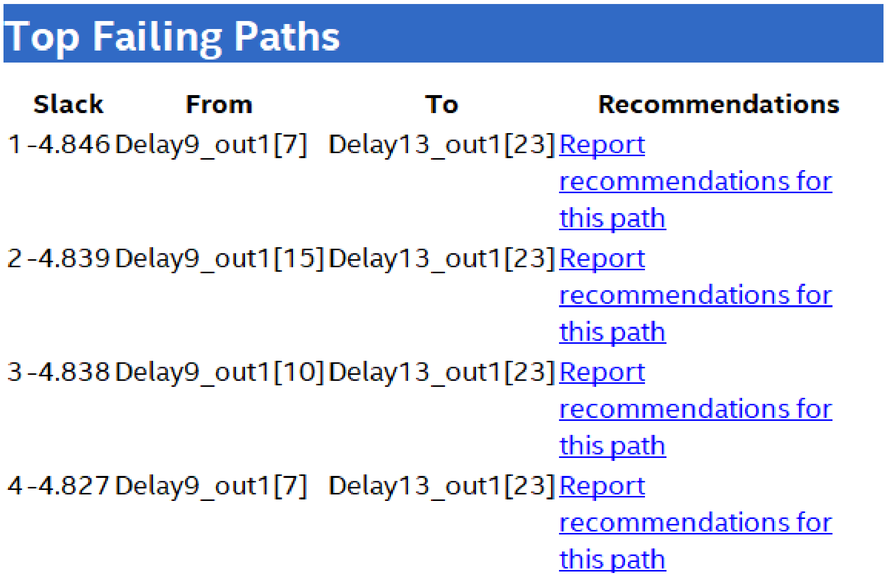
```
/Farrow_Canonic_tb/In1_offset   11585
/Farrow_Canonic_tb/In1          11585
/Farrow_Canonic_tb/check1_d...  0
/Farrow_Canonic_tb/snkDonen     St1
/Farrow_Canonic_tb/resetn       St1
/Farrow_Canonic_tb/tb_enb       St1
/Farrow_Canonic_tb/ce_out       St1
/Farrow_Canonic_tb/Out1         -8754
/Farrow_Canonic_tb/Out1_enb     St1
/Farrow_Canonic_tb/Out1_last... St0
/Farrow_Canonic_tb/Out1_add...  0000...
/Farrow_Canonic_tb/fp_Out1_...  -214...    -2147483643
/Farrow_Canonic_tb/Out1_exp...  -11279
/Farrow_Canonic_tb/status_O...  1          1
/Farrow_Canonic_tb/Out1_ref     -11279
/Farrow_Canonic_tb/Out1_test... 0
/Farrow_Canonic_tb/testFailure  St0
```

Synthesis Results:

```
 +-------------------------------------------------------------------------------------+
; Analysis & Synthesis Summary                                                          ;
+----------------------------------+---------------------------------------------------+
; Analysis & Synthesis Status       ; Successful - Mon May  8 17:41:29 2023           ;
; Quartus Prime Version             ; 22.1std.1 Build 917 02/14/2023 SC Lite Edition ;
; Revision Name                     ; Farrow_filters                                  ;
; Top-level Entity Name             ; Farrow_Canonic                                  ;
; Family                            ; Arria II GX                                     ;
; Logic utilization                 ; N/A                                             ;
;     Combinational ALUTs            ; 603                                  ;
;     Memory ALUTs                   ; 0                                    ;
;     Dedicated logic registers      ; 410                                  ;
; Total registers                    ; 410                                  ;
; Total pins                         ; 48                                   ;
; Total virtual pins                 ; 0                                    ;
; Total block memory bits            ; 152                                  ;
; DSP block 18-bit elements          ; 12                                   ;
; Total GXB Receiver Channel PCS     ; 0                                    ;
; Total GXB Receiver Channel PMA     ; 0                                    ;
; Total GXB Transmitter Channel PCS ; 0                                     ;
; Total GXB Transmitter Channel PMA ; 0                                     ;
; Total PLLs                         ; 0                                    ;
; Total DLLs                         ; 0                                    ;
+----------------------------------+----------------------------------------------+
```

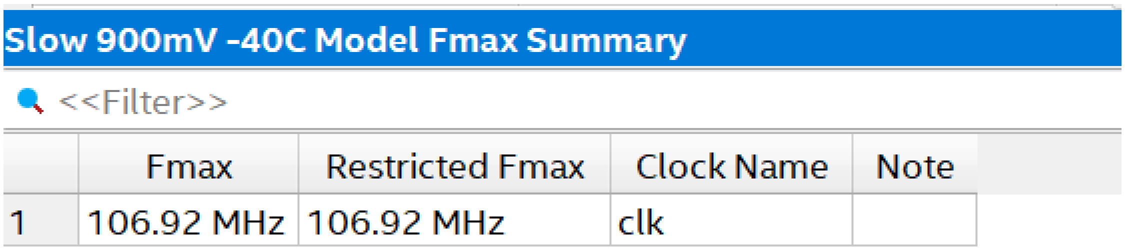
Farrow filter SDC file :

```
# Create clock constraints
create_clock -name clk -period 5 [get_ports {clk}]
# Create virtual clocks for input and output delay constraints
#create clock -name vclk -period 4.000

# derive clock uncertainty
derive_clock_uncertainty

# set input and output delays
set_input_delay -clock { clk } 0.1 [all_inputs]

set_output_delay -clock { clk } 0.1 [all_outputs]
```



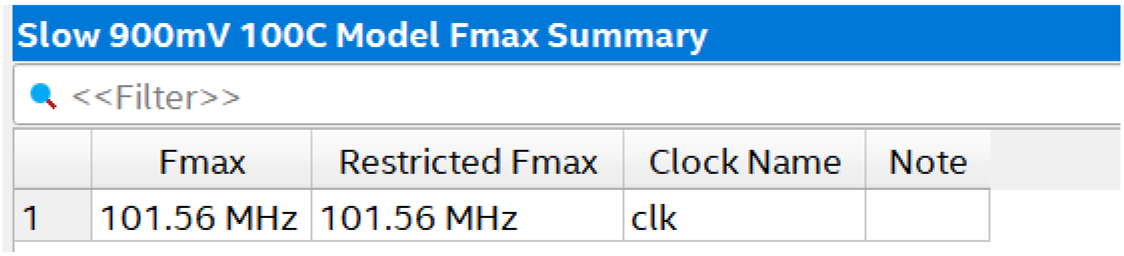
Top Failing Paths

	Slack	From	To	Recommendations
1	-4.846	Delay9_out1[7]	Delay13_out1[23]	Report recommendations for this path
2	-4.839	Delay9_out1[15]	Delay13_out1[23]	Report recommendations for this path
3	-4.838	Delay9_out1[10]	Delay13_out1[23]	Report recommendations for this path
4	-4.827	Delay9_out1[7]	Delay13_out1[23]	Report recommendations for this path



Slow 900mV -40C Model Fmax Summary

	Fmax	Restricted Fmax	Clock Name	Note
1	106.92 MHz	106.92 MHz	clk	



Slow 900mV 100C Model Fmax Summary

	Fmax	Restricted Fmax	Clock Name	Note
1	101.56 MHz	101.56 MHz	clk	

```
------------------------------------------------------------
Timing Analyzer Summary
------------------------------------------------------------

Type  : Slow 900mV 100C Model Setup 'clk'
Slack : -4.846
TNS   : -420.775

Type  : Slow 900mV 100C Model Hold 'clk'
```

```
Slack : 0.140
TNS   : 0.000

Type  : Slow 900mV 100C Model Recovery 'clk'
Slack : 3.508
TNS   : 0.000

Type  : Slow 900mV 100C Model Removal 'clk'
Slack : -0.112
TNS   : -25.851

Type  : Slow 900mV 100C Model Minimum Pulse Width 'clk'
Slack : 1.154
TNS   : 0.000

Type  : Slow 900mV -40C Model Setup 'clk'
Slack : -4.353
TNS   : -358.367

Type  : Slow 900mV -40C Model Hold 'clk'
Slack : 0.151
TNS   : 0.000

Type  : Slow 900mV -40C Model Recovery 'clk'
Slack : 3.575
TNS   : 0.000

Type  : Slow 900mV -40C Model Removal 'clk'
Slack : -0.091
TNS   : -20.092

Type  : Slow 900mV -40C Model Minimum Pulse Width 'clk'
Slack : 1.154
TNS   : 0.000

Type  : Fast 900mV -40C Model Setup 'clk'
Slack : 0.275
TNS   : 0.000

Type  : Fast 900mV -40C Model Hold 'clk'
Slack : 0.095
TNS   : 0.000

Type  : Fast 900mV -40C Model Recovery 'clk'
```

```
Slack : 4.144
TNS   : 0.000

Type  : Fast 900mV -40C Model Removal 'clk'
Slack : -0.016
TNS   : -1.195

Type  : Fast 900mV -40C Model Minimum Pulse Width 'clk'
Slack : 1.154
TNS   : 0.000

------------------------------------------------------------
```

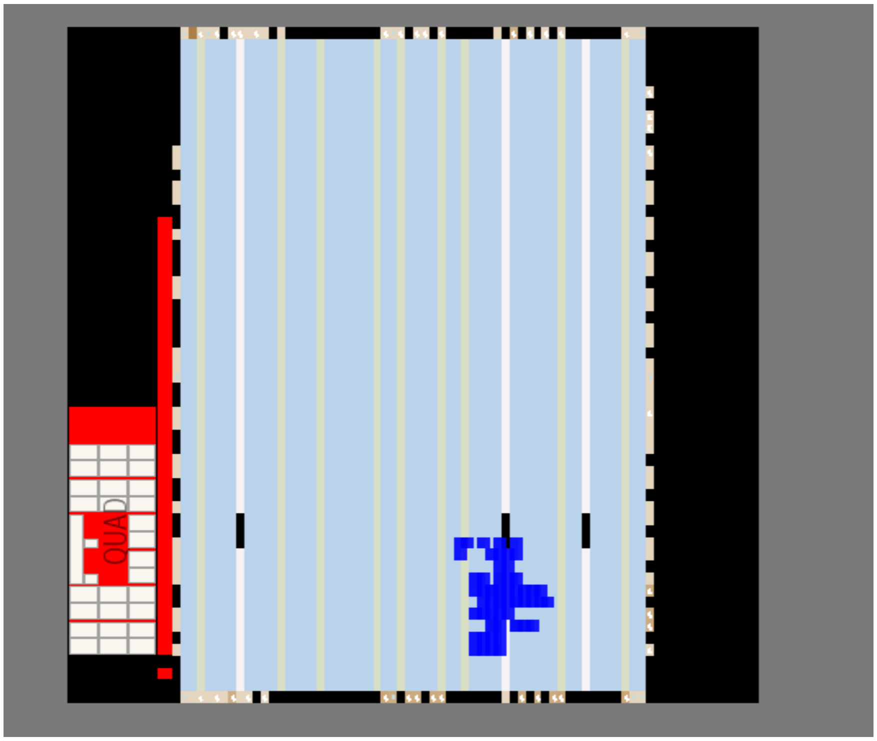


Fig 4 : FPGA resource usage highlighted in dark blue

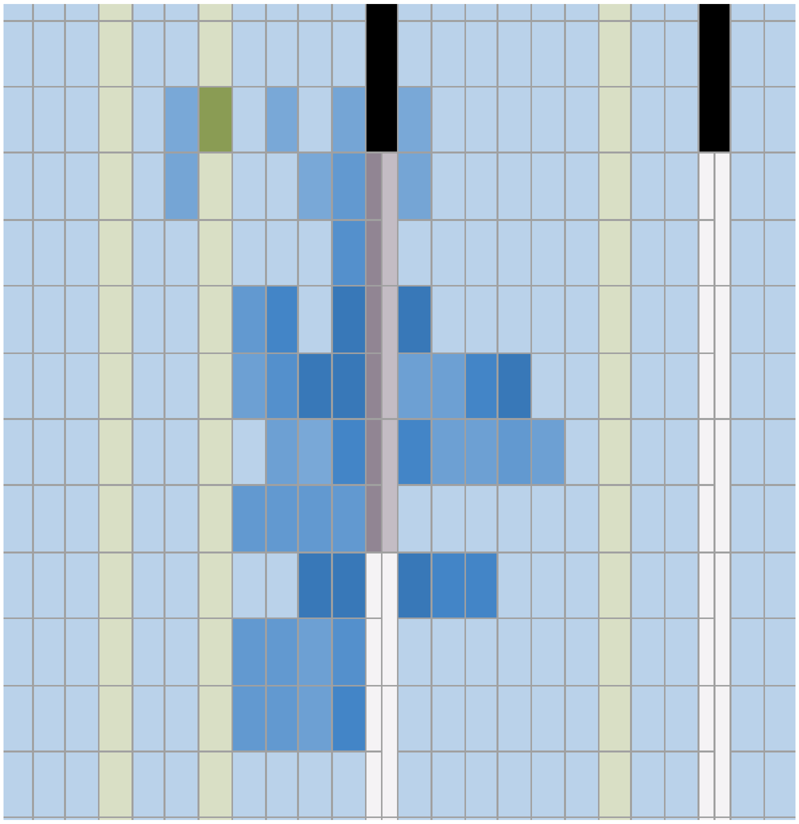


Fig 5 : FPGA resource usage - dark blue cells are configurable logic blocks, purple blocks are DSPs

Conclusion :

A resampling fractional delay FIR filter using Farrow structure was implemented in this assignment. The FIR filters are built using a transposed FIR filter architecture instead of the canonical / direct form 1 structure. FIR filter structures are also very amenable to pipelining. I further pipelined the second stage where a linear combination of the FIR filter outputs is computed, by adding 4 pipe cuts. The critical path only includes one multiplier (implemented on DSP) and an adder. The SDC file had a target clock frequency of 200 MHz, the maximum frequency of the farrow filter was 101.56 MHz (baseline Fmax was 28 MHz) at 900 mV 100C slow corner. 410 registers, 603 combinational ALUTs , 12 DSP 18-bit block elements, and 152 block memory bits were used in total on the Arria II GX FPGA platform.