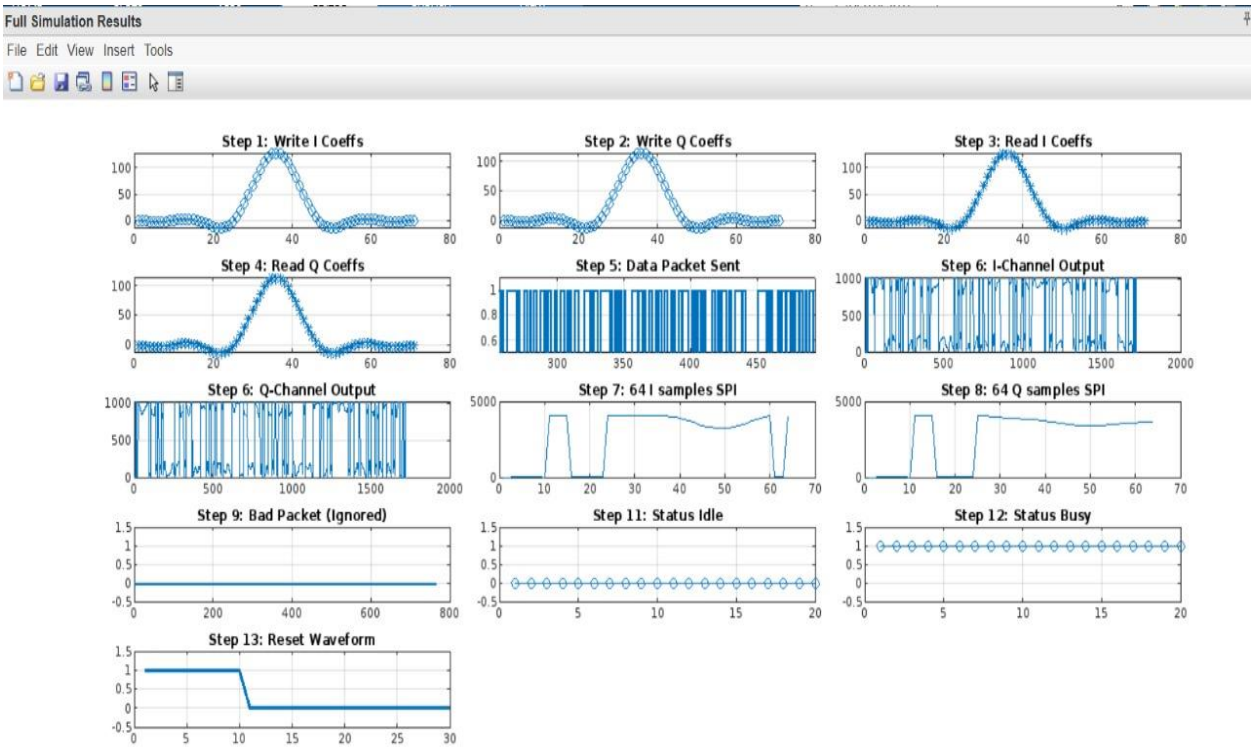
64QAM Modulator

- Developed a synthesizable RTL design for a 64-QAM modulator to process a 60 Mbps baseband stream into 10-bit I/Q data at 130 MHz, implementing a full DSP chain including symbol mapping, upsampling, and FIR filtering.
- Designed a fully custom SPI slave interface with a 34-bit control protocol, enabling dynamic runtime configuration of upsampling factors (×4–×13) and programmable filter coefficients.
- Implemented robust **clock domain crossing (CDC)** logic using FIFO-based synchronizers and multicycle paths to interface three asynchronous domains: SPI (100 kHz), baseband data (60 MHz), and DSP core (130 MHz).
- Verified proper **reset sequencing** across multiple clock domains and integrated RTL assertions to ensure safe enable/disable transitions.
- Designed register-mapped control architecture and validated functional coverage for SPI write/read back, modulation pipeline timing, and packet boundary detection using simulation waveforms.

The design using the MATLAB files to help generate test stimulus and response.



1. Write all 71 I-channel filter coefficients (graph step 1)
   The graph(step1) depicts the 71 I-channel filter taps that is being written.
   It exactly matches with the expected RRC shape, confirming proper tap generation.

2. Write all 71 Q-channel filter coefficients (graph step 2)
   The Q-channel coefficients are like the I-channel but slightly lower in amplitude, as required.
   Graph 2 confirms correct 0.9 scaling applied during filter creation.

3. Read back and check all 71 I-channel coefficients (graph step 3)
   This graph displays that all I-channel filter taps were correctly stored and retrieved without any mismatch.

4. Read back and check all 71 Q-channel coefficients (graph step 4)
   This confirms that Q-channel filter taps were being correctly written to and read back from memory.

5. Send data packet (graph step 5)
   The bitstream for the packet is completely visualized, displaying a correctly randomized 64-QAM data packet transmission.

6. Check all I/Q outputs to verify they are correct for the entire data packet (graph step 6)
   The graph(step6) represents the filtered modulator outputs (after pulse shaping and down-scaling).
   Values are within expected range, confirming correct I/Q processing.

7. Read back and check all 64 stored I-channels values via SPI interface (graph step 7)
   The reconstructed I samples are matching expected output samples correctly from SPI addresses 512–639.

8. Read back and check all 64 stored Q-channels values via SPI interface (graph step 8)
   Like above, the reconstructed Q samples match expected output samples correctly from SPI addresses 768–895.

9. Send data packet with incorrect header, demonstrate baseband data packet is ignored (graph step 9)
   The module ignores the invalid packet successfully, as no processing or storage occurs for the bad packet.

10. Disable mapping via SPI interface, demonstrate baseband data packet is ignored (graph step 10)
    After disabling the mapping, the valid packets are also ignored, demonstrating proper control via mapping enable.

11. Read back baseband mapping status address (register address x0) when no baseband data packed is being sent, check that status is correct (graph step 11)
    Mapping status reads 0 continuously, this confirms no active packet transmission and correct IDLE detection.

12. Read back baseband mapping status address (register address x0) when baseband data packed is being sent, check that status is correct (graph step 12)
    Mapping status reads 1 during transmission, showing the baseband is BUSY, confirming proper packet activity detection.

13. Reset waveforms showing proper synchronization and sequencing (graph step 13)
    The reset signal goes from high to low. This confirms the correct system reset behavior for initialization or error recovery.

What is the maximum SPI clock frequency?

For SPI Interface, given a slack of 7.329ns when simulating with a 10ns clock, the maximum minimum clock period is 10 – 7.329 = 2.671ns. Thus, the maximum clock frequency is 1/period = 374.391MHz.

What is the maximum data clock frequency?

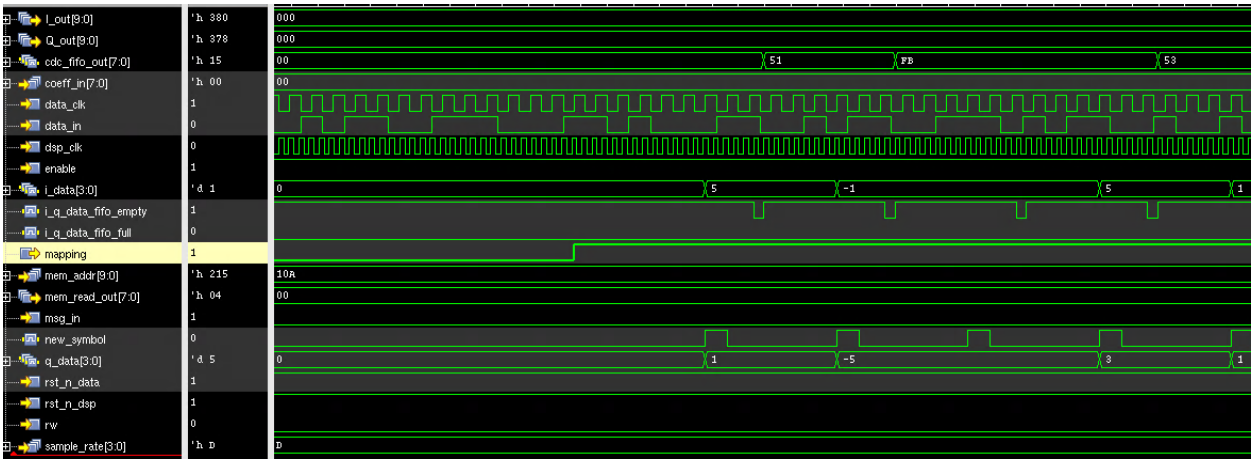
For SPI Interface, given a slack of 7.829ns when simulating with a 10ns clock, the maximum minimum clock period is 10 – 7.829 = 2.171ns. Thus, the maximum clock frequency is 1/period = 460.617MHz.

What is the maximum DSP clock frequency?

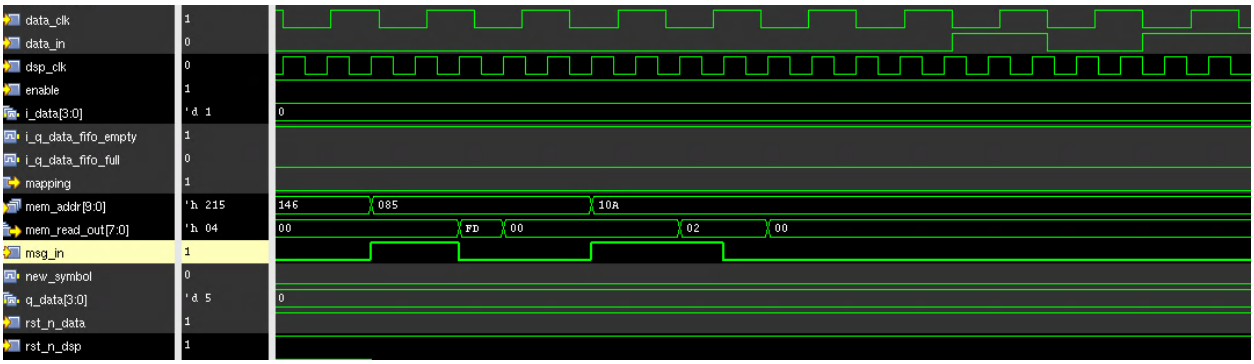
For SPI Interface, given a slack of 7.913ns when simulating with a 10ns clock, the maximum minimum clock period is 10 – 7.913= 2.087ns. Thus, the maximum clock frequency is 1/period = 479.156MHz.

What is the area?

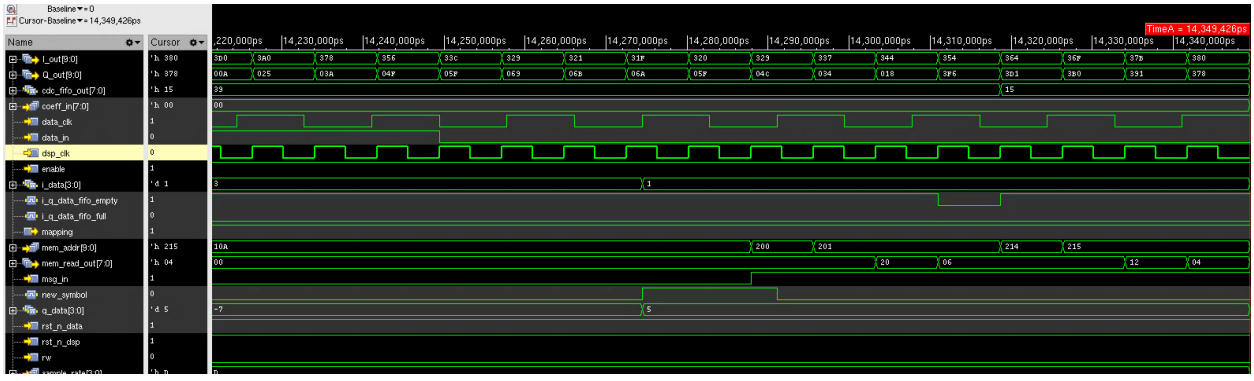
17905.920200



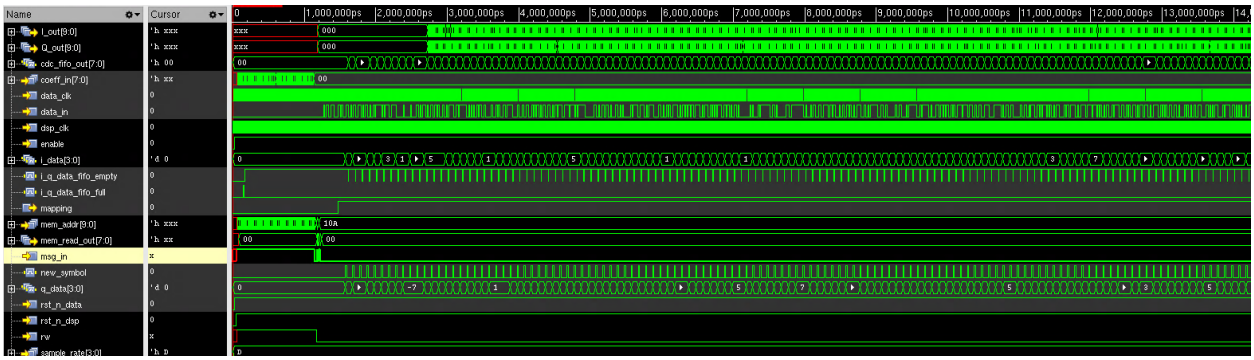
Symbol mapping functionality: header is read and mapping is enabled. Symbols are send to cdc



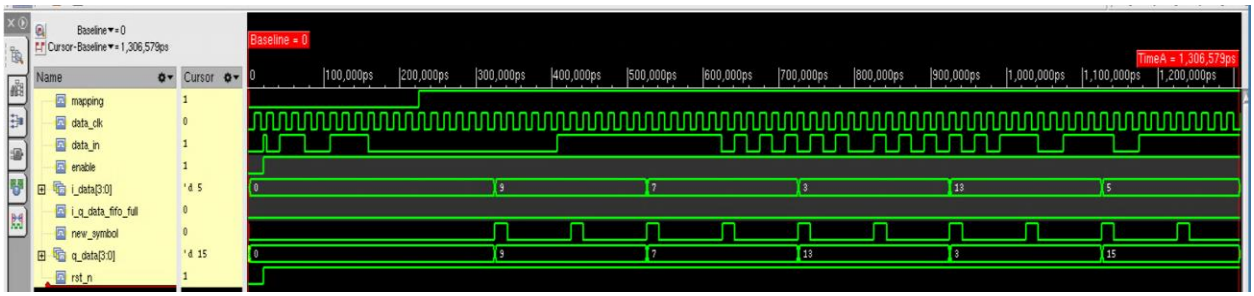
Coeff read memory functionality



Sample storage read functionality: first 8 msbs then last 4 lsbs



Full waveform



Symbol mapping &

constraints for clock domain crossing with 3 clock domains.

What is the maximum SPI clock frequency?

For SPI Interface, given a slack of 7.329ns when simulating with a 10ns clock, the maximum minimum clock period is 10 – 7.329 = 2.671ns. Thus, the maximum clock frequency is 1/period = 374.391MHz.

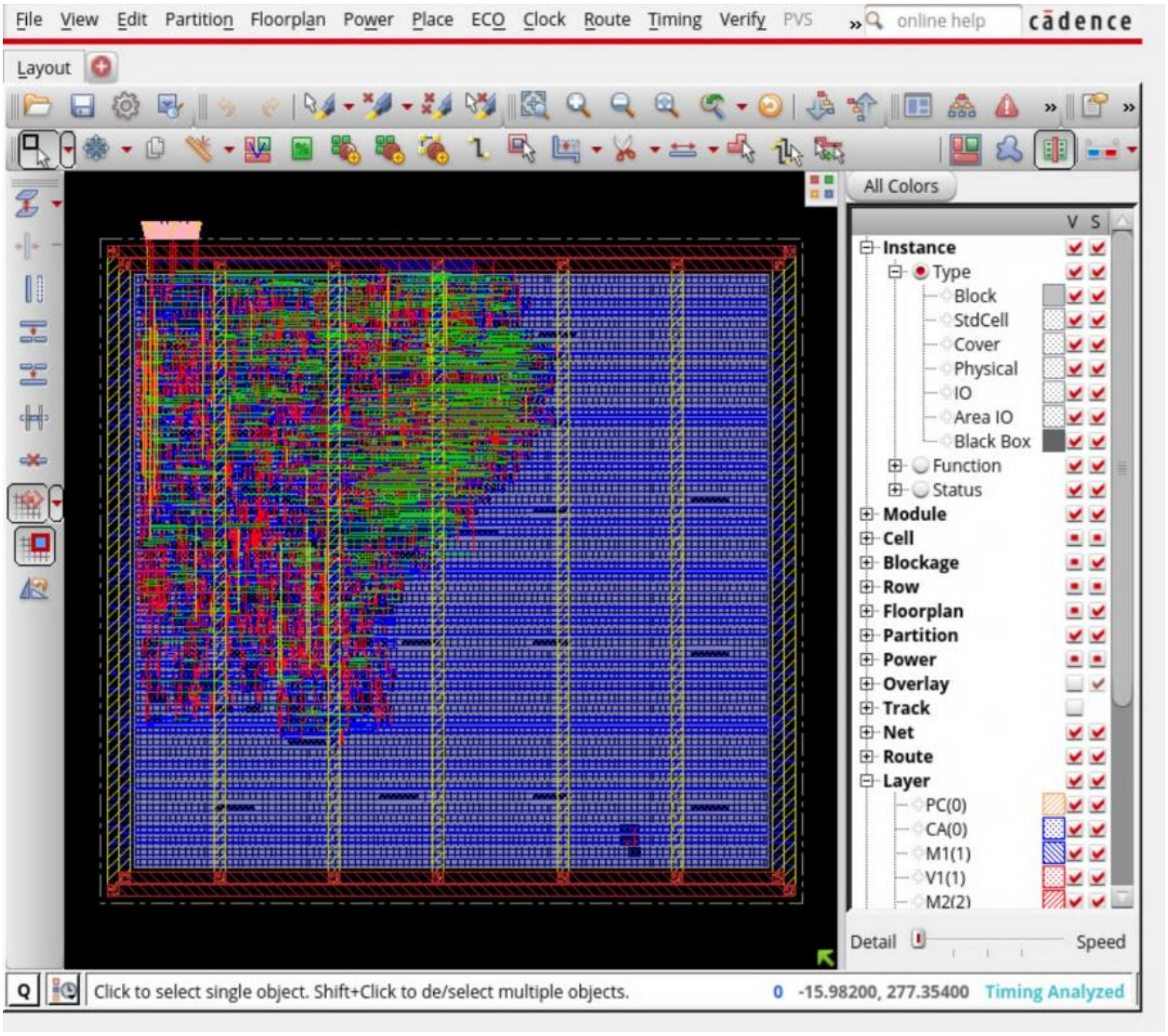
What is the maximum data clock frequency?

For SPI Interface, given a slack of 7.829ns when simulating with a 10ns clock, the maximum minimum clock period is 10 – 7.829 = 2.171ns. Thus, the maximum clock frequency is 1/period = 460.617MHz.

What is the maximum DSP clock frequency?

For SPI Interface, given a slack of 7.913ns when simulating with a 10ns clock, the maximum minimum clock period is 10 – 7.913= 2.087ns. Thus, the maximum clock frequency is 1/period = 479.156MHz.

3. Image of the layout


File View Edit Partition Floorplan Power Place ECO Clock Route Timing Verify PVS
online help
cadence
Layout
All Colors
V S
Instance
Type
Block
StdCell
Cover
Physical
IO
Area IO
Black Box
Function
Status
Module
Cell
Blockage
Row
Floorplan
Partition
Power
Overlay
Track
Net
Route
Layer
PC(0)
CA(0)
M1(1)
V1(1)
M2(2)
Detail
Speed
Click to select single object. Shift+Click to de/select multiple objects.
0 -15.98200, 277.35400
Timing Analyzed