The George Washington University
Depart of Electrical and Computer Engineering

64-QAM Modulator Datasheet

This document describes the functionality of a 64-QAM modulator for RF communications. This modulator processes a 60 Mbit/sec datastream into in-phase and quadrature components, upsamples the data, and then does pulse-shape filtering to optimize the output spectrum. This modulator is expected to supply 10-bit parallel data to two digital-to-analog converters (DACs) at 130 MHz. A serial peripheral interface (SPI) is implemented for control and configuration of the modulator.

Modulator Block Diagram

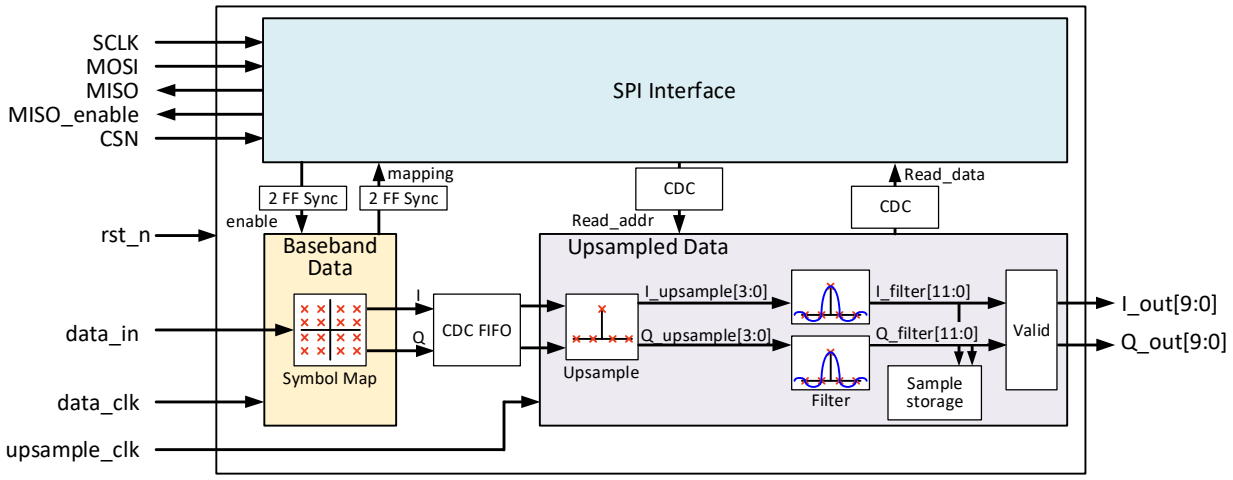

Figure 1. Modulator block diagram

IO Interface

Table 1. Input / Output Signals

Signal Name	Direction	Description
rst_n	Input	Active-low reset
SCLK	Input	Serial Clock – 100 kHz operating frequency
MOSI	Input	Main Out Sub In – input serial data stream
MISO	Output	Sub Out Main In – output serial data stream
MISO_enable	Output	Active-high enable to MISO output
CSN	Input	Active-low chip select
data_clk	Input	Datastream clock – 60 MHz operation
data_in	Input	60 Mbps datastream
upsample_clk	Input	Upsampled datastream clock – 130 MHz operation
I_out[9:0]	Output	In-phase output data to DAC
Q_out[9:0]	Output	Quadrature output data to DAC

Serial Data Interface

A SPI interface is implemented for control and configuration of the modulator. The 34-bit serial data message format is listed in table 2. The serial message and all address/data fields are shifted out MSB-first. Figure 2 shows the waveforms for the serial data message. The active-low CSN (chip select) goes active at the beginning of the serial message and the main shifts out the first bit of the serial data on MOSI, and all subsequent bits will be shifted out of the main on the SCLK falling edge. The sub should clock in the data with the rising edge of SCLK.

If the read/write bit indicates a serial write message, the 8-bit register data is shifted from the main on MOSI, and no activity should occur on MISO and MISO_enable is disabled. If the read/write bit indicated the serial message, no activity should occur on MOSI. The sub will send the 8-bit register data on MISO and MISO_enable will go active. The data on MISO will be shifted out on the SCLK falling edge and captured by the main on the SCLK rising edge.

Table 2. Serial Message Description

Message Bits	Description
33	Read/write bit, 0 = read, 1 = write
32:23	10-bit register address
22:14	9-bit dead time
13:6	8-bit register data
5:0	6-bit dead time

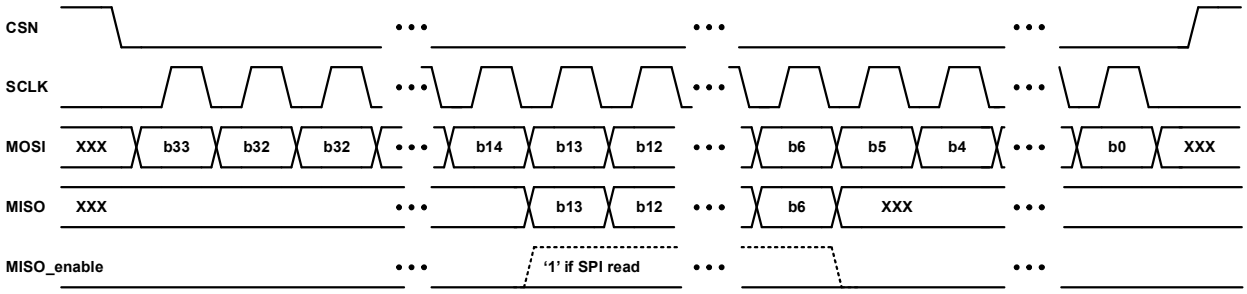

Figure 2. Serial data message

Baseband Data Interface

The baseband data interface receives data and does the 64-QAM symbol mapping. The baseband data interface and clock operate at 60 MHz. A baseband data packet is 3084 bits long. The first 12 bits serve as a packet header to signify the beginning of the message. A 12-bit header value of “101100 111000” will be at the beginning of all data packets, if a data packet has an incorrect header the remainder of the packet should be ignored. The 12-bit header should not be mapped into symbols.

The remaining 3072 bits contain the data that needs to be mapped into symbols. For every group of 6 bits in the message, the 6-bit value should be used to map the data into

I/Q-values in accordance with Figure 3. For example, a 6-bit input value of “111 000” will result in an output value of I = 7, Q = -7 (lower right constellation point). The I and Q outputs of the mapping should both be 4-bit two’s complement values.

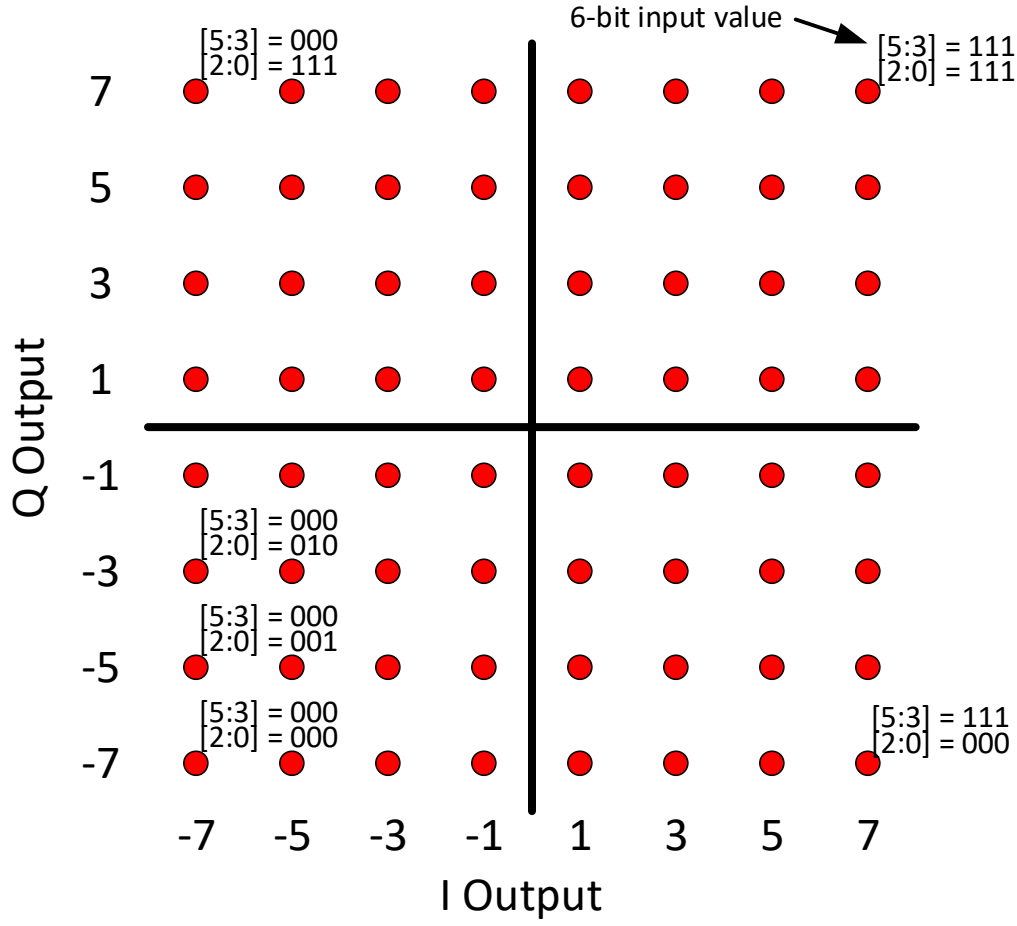

Figure 3. 64-QAM symbol mapping

As shown in Figure 4, the symbol mapping output should change every six clock cycles, and a “new_symbol” signal should go active the during the clock cycle where new values are available on the I and Q outputs. Once the entire data packet is mapped, the outputs of the mapping shall be set to all 0’s and the “new_symbol” output should stay inactive.

Note that the timing diagram shows the I, Q, and new_symbol changing one clock cycle after last bit of data_in symbol is received. Additional clock cycles of delay are allowed, as long as no data is lost during the mapping.

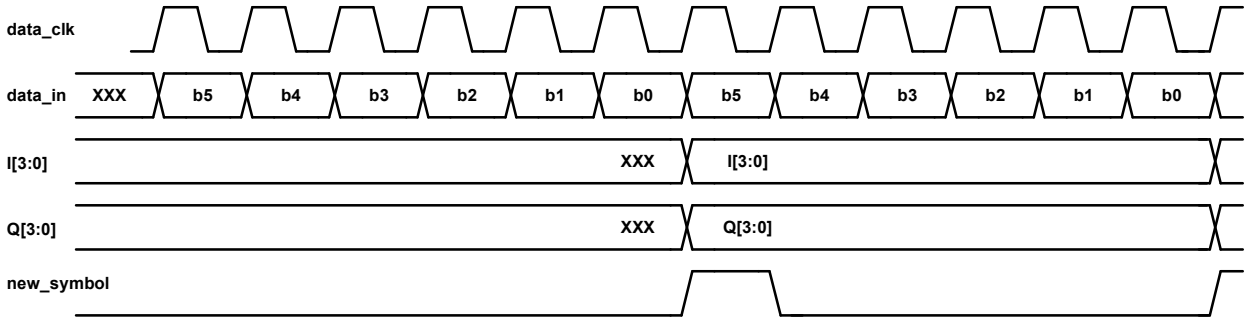

Figure 4. Symbol mapping timing

Upsampling

Once the 3072 data bits have been mapped into 512 symbols with 4-bit I and Q values, the I/Q values will cross clock domains via a clock-domain crossing (CDC) FIFO to the upsampled data clock domain. The I/Q data will then be upsampled with zero padding by a reconfigurable upsampler. This upsampler can be configured through the SPI interface to upsample by any factor from x4 upsampling to x13 upsampling. Both the I/Q data will be upsampled by the same factor.

Filtering

The upsampled I/Q values will next be filtered by independent 71-coefficient FIR filters. The filter coefficients will be 8-bit values that can written/read via the SPI interface, with separate filter coefficients being available for the I/Q channels. The width of the filter inputs/outputs are given by the below table.

Signal	Data width
x[n] width	4-bit
Filter Coefficients	8-bit
Filter output	12-bit

Filter Output Storage

After filtering, a portion the I/Q filter output values will be saved into an on-chip register file for readback via the SPI interface. Only the first 64 I/Q samples and last 64 I/Q samples of each serial data packets will be stored. See the register address map for specific details of storage formatting. Subsequent serial data packets will overwrite previous packets in the filter output storage.

Output Validation

The 10-bit I/Q outputs of the modulator will only be active when the filter output is valid, otherwise the outputs will be all zeros. The output is considered valid if it is the result of one of the 512 upsampled symbols being convolved with the filter. (Hint: this will be a function of the upsampling rate, filter latency, and number of filter taps) When valid, the I/Q outputs of the validation block will be the 10 MSBs of 12-bit I/Q filter outputs.

Register Address Map

Reg Address	Data Format (MSB -> LSB)	Description
0	xxxx xxME	Baseband data mapping control E = 0 – mapping disabled E = 1 – mapping enabled M = 0 – waiting for baseband data packet M = 1 – currently mapping baseband data packet
1	dddd dddd	SPI test data register Register to write/read data from to verify SPI functionality
2	xxxx UUUU	Upsampling control: U = upsample factor, valid values from x4 to x13 If invalid value is written, the upsampling factor will default to x13 upsampling
3-127		Unused
128-198	IIII IIII	8-bit Filter Coefficients for I-channel: Address 128 = coefficient 0 Address 198 = coefficient 70
199-255		Unused
256-326	QQQQ QQQQ	8-bit Filter Coefficients for Q-channel: Address 256 = coefficient 0 Address 326 = coefficient 70
512-639	IIII IIII (even addrs) xxxx IIII (odd addrs)	I-channel filter outputs, first 64 samples of serial data message (read only) Addr 512 = 8 MSBs of symbol 0 Addr 513 = 4 LSBs of symbol 0 Addr 514 = 8 MSBs of symbol 1 … Addr 639 = 4 LSBs of symbol 63
640-767	IIII IIII (even addrs) xxxx IIII (odd addrs)	I-channel filter outputs, last 64 samples of serial data message (read only) Addr 640 = 8 MSBs of symbol 448 Addr 641 = 4 LSBs of symbol 449 Addr 642 = 8 MSBs of symbol 450 … Addr 767 = 4 LSBs of symbol 511
768-895	QQQQ QQQQ (even addrs) 0000 QQQQ (odd addrs)	Q-channel filter outputs, first 64 samples of serial data message (read only) Addr 768 = 8 MSBs of symbol 0 Addr 769 = 4 LSBs of symbol 0 Addr 770 = 8 MSBs of symbol 1 … Addr 895 = 4 LSBs of symbol 63
896-1023	QQQQ QQQQ (even addrs)	Q-channel filter outputs, last 64 samples of serial data message (read only)

	0000 QQQQ (odd addrs)	Addr 896 = 8 MSBs of symbol 448 Addr 897 = 4 LSBs of symbol 449 Addr 898 = 8 MSBs of symbol 450 … Addr 1023 = 4 LSBs of symbol 511

Reset Synchronization and Sequencing

The modulator should use the reset synchronizer and reset sequencing shown in figure 5.

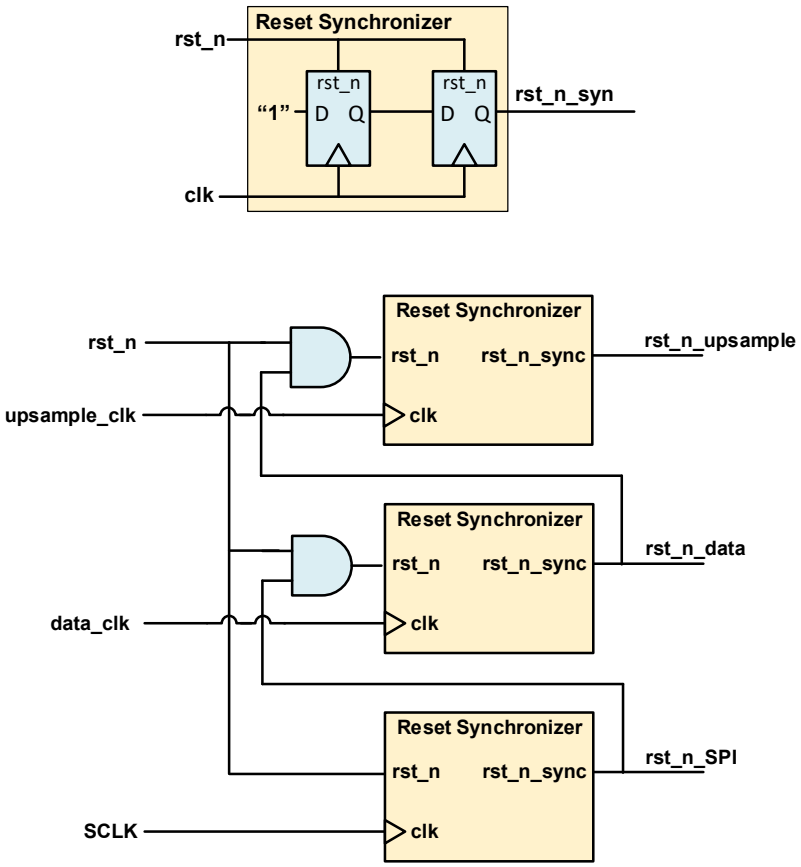

Figure 5. a) Reset synchronizer implementation and b) sequenced reset implementation