Updated 1/18/24

The module will output pulses with defined rise and fall edge shapes and with a programmable duration for the flat top of the pulse.
Pulse rise and fall edges will be symmetrical. I.e. The falling edge is the reverse of the rising edge.
Data points (16-bit) describing the edge will be stored in Wavetable RAM.

A second RAM, the Pulse Definition RAM, contains a table describing each output pulse's parameters. The Pulse Definition RAM can hold more than 16 possible entries.
Each entry consists of:

- A 24-bit time value, indicating when to start a pulse, in 10 nano-sec increments. The time counter itself is in the block containing all 32 of these channels. Max duration 2^24 x 10 nsec = ~168 msec.

- 16 bits for the pulse edge time scale factor (stretch & compress concern).
- 16 bits for the pulse edge amplitude scale factor (*).
- 12 bits for the start address of the pulse edge data in the Waveform RAM, the bit width should increase with the amount of address the wavetable has.
- 10 bits for the number of points/addresses used by the pulse edge, the bit width should increase with the amount of addresses the wavetable has.
- 17 bits of wait times (flat_top), managed by an internal counter process sm_top_counter in state S_WAVE_TOP. Max duration 2^17 - 1 * 10 nsec = 131 usec

Each pulse takes four memory addresses of the pulse definition RAM, with the following assignments for each address of an entry to four different registers:

1. Start time 24 bits. [23:0] (1 24 bit reg)
2. Wave start address 12 bit at [11:0] + Wavelength 10-bit at [25:16] (Should we do 12-bit [27:16])? (
3. Scale factors, stored as 16-bit fixed point values with 8 fractional bits. [31:16] [15:0]
    a. Gain scale would always be <=1
        i. 1 bit int + 15 fractional bit
    b. Time factor always >= 1 (8 bit int + fraction)
4. Flat-top 17-bit. [16:0]

The above allocation methods are based on the following assumptions/constraints, which will be checked in software:

- This module runs at a 100MHz clock frequency. .i.e. 10 nsec steps.
- Minimum pulse duration is 3 clock cycles i.e. 30 nsec.
- Flat top duration cannot exceed 2^17 * 10 nsec.
- The wavetable contains at least one wave edge.

- The number of points in a pulse rising/falling edge edge for one entry has to be less than the number of addresses available for the wavetable ram.
- The numbers of points in the pulse edges can be between 1 and maximum size of the waveform RAM (4096). A value of zero means that there is a single point in the waveform rise. (i.e. a step function). A value of 0x3FF means 4096 points.

The time scale factor controls the rate at which the waveform points are read from the Waveform table. The amplitude scale factor is used to multiply each waveform point, which is then shifted to maintain a 16-bit range.

Total wavetable ram storage is 4096 points, and it can be divided into as many different waveform shapes that are needed. If there are two different pulses stored in the RAM, each of them can have a maximum of 2048 points, and if a third pulse is added, each pulse will now have a maximum of 1365 points. The pulse definition allows for 256 different pulses to be defined for a channel. If they each require unique points then each waveform can only be 16 points but there is nothing to stop one waveform using 1024 points and the others sharing the remaining storage. Multiple pulses can use the same waveform, or parts of a waveform, while defining different gain and time factors.

A state machine will generate the addresses to read the Pulse Definition RAM, store the parameter outputs, and then generate the required Waveform RAM addresses. The amplitude scale factor will be used to multiply the Waveform table data points, which will then be shifted into a 16-bit value before being output to the DAC. There should be the following three states controlling the wave outputs: S_WAVE_RISE, S_WAVE_TOP, and S_WAVE_FALL. The state machine generates addresses for the wavetable RAM to increment to, the ‘Flat-top’ time to wait when the addresses have incremented, and then decrements the waveform RAM address back to its starting address to complete a pulse output.

The Pulse Definition RAM defines the addresses that would be used to store a pulse’s rise, flat, or fall (reverse of rise). Once the wavetable RAM’s address comes back to its starting address,a pulse is completed, the Pulse Definition address will be incremented by four to obtain parameters for the next pulse. The state machine then waits until the new pulse start time is reached then begins the new pulse output. Unused Pulse Definition entries should have a time value of zero.

One pulse definition RAM entry will require four 32-bit memory locations to store parameters describing the pulse, are stored in the Pulse Definition RAM. E.g. Entry 0 (addresses 0-3) will store parameters for the first pulse. Entry 1 (addresses 4-7) store parameters for the second pulse. The start time of pulse N+1 must be greater than the start time of pulse N.

We can have one wavetable but using different scale for different “looks” of the output wave

Possible cases to consider:

- pulses start at the right time
- Is the waveform data output correct and the right length?
- Flat-top length?
- What happens when the flat-top length in the pulse definition file = 0
- How many pulses have you generated in a sequence?
- What happens if the start time of pulse N is less than that of pulse N-1?
- What happens when the waveform start address and waveform length are > 4096?
- Are you checking the result by looking at the output waveform data bus or does your testbench calculate the expected values? i.e. is it self-checking?
- What happen if we start a pulse right after another

TODO:
Tb roundings?

To sara: should waveform output zero after output finish or stay at last value (possibly non-zero)

Should wave overlap?

TODO: error handling

1. start right after previous pulse, neet to wait for v_pulsetime + v_wavesteps + v_wavetopwidth + v_wavesteps + 10 cycles
2. Time factor addition overflow
3. Scale factor multiplication overflow
4. Wavetable address overflow
5. Time factor too small (<1)
6. Scale factor too big (>1)
7. (maybe.. Need to test) addr stopped/started at > stop addr
8. (maybe.. Need to test) addr stopped/started at > start addr

TODO: more advanced tb
Two paralle archetechture for hardware and simulation