**Local oscillator and phase detector I/O ports and use description**

**Local oscillator (lo)**

|  |  |
| --- | --- |
| **Port** | **Description** |
| *clk* | Clock input, used as reference in the oscillator |
| *rst* | Reset input |
| *setFreq* | Loads the parameters into the internal oscillator, active high |
| *ain* | Oscillator parameters, output frequency responds to the following equation: |
| *bin* |
| *min* |
| *enable* | Enable input, activates the internal oscillator, active high |
| *sin (data(x5) + valid)* | Sine output |
| *cos (data(x5) + valid)* | Cosine output |

**Example of use**

We have a reference clock frequency of 100Mhz and want to achieve an output frequency of 5Mhz.

Doing the math, we need to set the (a+b/m) to a value of 209715.2, which leads us to set the following values: *a* = 209715, *b* = 2, *m* = 10.

To load the values, set the *setFreq* input signal to one for at least one cycle.

Once the values are loaded, simply set the *enable* signal to ‘1’ and few cycles later the output signals will appear.

The module will work as long as the *enable* signal is set to ‘1’.

**Phase detector (atan\_phase\_detector\_x5)**

|  |  |
| --- | --- |
| **Port** | **Description** |
| *clk* | Clock input |
| *rst* | Reset input |
| *enable* | Enable input, enables the phase detector |
| *DataIn (data(x5) + valid)* | Input data |
| *sin (data(x5) + valid)* | Reference sine input |
| *cos (data(x5) + valid)* | Reference cosine input |
| *mult\_coeff* | Multiplication coefficient input, result of dividing the closest greatest power of two to the product of 5\*acc\_value and multiplying the result by 16384 |
| *shift\_value* | Internal shift value input, bit position of the previously calculated power of two |
| *acc\_value* | Accumulator value, expressed in x5 samples |
| *phase* | Phase output, from -180 to 180 degrees, 2’s complement |
| *module* | Module output |
| *module\_phase\_valid* | Module and phase valid output |

**Example of use**

Continuing with the previous example, we want to configure the phase detector to work with a 5Mhz signal.

First we select the *acc\_value*. We need the center frequency of the internal average\_filter to be 5Mhz or a submultiple. The center frequency is 100(Mhz)/*acc\_value*. In this case, the minimum value is 20.

The closest greatest power of two to the product of 5x20 is 128, which results in a value of 8 for the *shift\_value* (bit counting starts at 1).

Finally the value for *mult\_coeff* is (128/100)\*16384 = 20971 (no decimals are written).

Once the parameters are set at the inputs, enable the phase detector by setting the enable signal to ‘1’.

Note that the latency depends on the number of samples used by the internal accumulator at the average filter.