9/19/2019

PCIe Implementation of Timing Slave System

Features:

1. GPS Time, 64 bit memory address or consistent 2x 32 bit memory address read in two (ordered) steps.
2. Start –Stop Register – Start on the next second boundary (gated). Gating time offset configuration option.
3. 16 Bit Digital I/O Card Interface
4. Every clock line to include rate, polarity, delay.
5. Support existing clocking header configuration in functionality <https://dcc.ligo.org/LIGO-D0902029>
6. Set Output Frequency, Clock Frequency
7. Duotone! Ability to turn this on and off by register.
8. Support LVDS (differential high seed clocks)
9. Support 10+ clock lines
10. ADNACO Board – Include timing lines which can be configured through the PCIe interface.

2/18/2020

Things to Consider:

1. New 2^19 Clock Frequency
2. Any form factor simplifications that might require modification to the existing IO chassis mounting scheme.
3. Other factors we are considering in the redesign.

Card Location:

1. We believe the card will take the place of the Contec Binary IO card in the IO chassis.
2. Form factor will be generally the same size as the Contec Binary IO card
3. It should take 3 months from start for development, 3 months for testing and programming.

Testing:

1. Initial testing to determine if we could see the PCIE card on the bus is completed (tested working)
2. Design a dummy counter FPGA program that can be read out by the PCIe bus as a proof of concept. Run this in the test stand to verify capabilities.

OK, see responses interspersed below:

Best, Rich

> On Feb 18, 2020, at 2:07 PM, Sigg, Daniel <[dsigg@caltech.edu](mailto:dsigg@caltech.edu)> wrote:

>

> ﻿

> With a PCI-E board, the timing fiber would be terminated internally and not at the front panel.

Nice

> This will work nicely with the new MPO fiber bundles/connectors. (Fan-out would move to CER).

Nice

>

> The binary IO board for the timing would disappear.

> The IRIG-B board would disappear.

Both nice

>

> Do you want a new backplane board? I am under the impression that

> supporting multiple clock frequencies for ADCs & DACs would require

> one. One could easily provide a separate clock & sync line for each

> board. Or, for 2, 4 or 8 slots together. Whatever makes sense. This is zero effort with an FPGA.

Yes. I think this will indeed be needed for any design requiring two or more clock frequencies.

>

> If there is a need for more binary lines, this would be straight forward too.

I can't think of why, but there's no cost to having a few land on the backplane board for a later use if we make provisions with a connector on the backplane board to fanout the binary lines for some future use.

>

> Daniel

>

>> Dear Marc,

>> We are gearing up to buy the parts needed to manufacture more IO chassis for Aplus and OPS. Part of this hinges on the Timing Slaves. We will need to know what to manufacture. It seems like the old timing slaves will be superseded by the new version you are designing.

>>

>> Please can you give us a description of what you plan to do, and when you will be finishing this new design? It would help us plan the next steps.

>>

>> Things to consider:

>> 1. New 2^19 clock frequency

>> 2. Any form-factor simplifications you may envision that might

>> require modifications to the existing IO chassis mounting scheme 3.

>> Other factors you are considering in the re-design

>>

>> Best, Rich

>

From Rolf 4-9-2020

The basic functions we need are:

- A minimum of 4 clock outputs:

- An ADC and a DAC clock to go to the IOC backplane at 65536Hz

- Another ADC and DAC clock via TBD connector to supply cards running at other than 65K. Rich would have info on that.

- The duotone signal to go to the IOC backplane

Via PCIe bus:

- Ability to set the clock output rates in steps of 2^n, from 65536 up to 1024KHz (2^16 - 2^19)

- Ability to start/stop the timing clocks being sent out. We presently do this by setting a bit in the Contec card, which then goes to the timing slave header via the I/O chassis backplane. When the start command is sent, the timing slave starts the clock outputs coincident with the next 1 sec tick so we always start synchronously on a one second boundary.

- Ability to read GPS time. We presently get this from the IRIG-B card. This is read out as two 32 bit words, GPS second and GPS nanosecond. On the Irig-B card, a read of the GPS second register will lock both GPS second and nanosecond until the code reads the nanosecond register.

- Ability to set the relative phase of the output clocks. We presently set the DAC output clock 1/2 cycle behind the ADC clock.

- Ability to read timing module state information eg clock on/off status, clocks rates, etc.Also, the status from the timing system, eg sync with master, etc. Guessing Daniel would know what timing system state info is available at the slave.

As this goes along, it would be good to see a layout for the available PCIe bus registers.  I think this would be helpful in reviewing the requirements ie see if anything is missing.

As for your question, what pins go where, Rich perhaps knows of the latest interconnect drawing from timing slave to IO backplane?

-rolf