

Nuclear Physics Division

# Description and Technical Information for the VXS FPGA-based Flash Time to Digital Converter board (vf2TDC)

Updated on: May 7, 2025

# Table of Contents

# Section Title Page

|  |  |  |
| --- | --- | --- |
| 1 | Introduction | 3 |
| 2 | PCB design of vf2TDC Module | 4 |
| 3 | Function Description | 7 |
| 4 | Specifications | 16 |
| 5 | Vf2TDC operation procedures | 20 |
| 6 | Programming Requirements | 20 |
| 7 | Pin out tables | 20 |
| Appendix A | Schematics | 33 |
| Appendix B | Fabrication Drawing | 51 |
| Appendix C | Bill of Materials | 54 |
| Appendix D Appendix E | Revision history References | 6265 |

# 1 Introduction

The vf2TDC (VXS FPGA-based Flash Time to Digital Converter board) is being developed based on the VETROC board, a generic IO board with the functionality defined by the FPGA firmware.

Figure 1 shows a picture of the assembled PCB board. Two sets of 32-pair differential signals are received using two 3M Pak 50 Boardmount Plug connectors (P50E-68P1-RR1-EA). The signals are converted to LVTTL on the PCB by TI SN65LVDT352, which translates any differential signals (LVDS, LVPECL, ECL) to LVTTL.

Two mezzanine board connectors with 32 channels each are built on the board, and connect directly to the FPGA IO. With the proper mezzanine boards, these can be inputs or outputs. The mezzanine connector uses FCI61082-101400LF, which is fully compatible with the CAEN 1495 mezzanine card.

To further expand the IO capability, the FPGA connects directly to the VME P2 backplane User Defined pins. A VME P2 backplane IO card can be developed, which can have up to 80 LVTTL signals and two pairs of high speed serial links.



Figure 1 vf2TDC board and its major components

The vf2TDC is a 192 channel TDC, which uses the two front panel connectors, two mezzanine boards as inputs, and 64 channels on VME P2 as inputs.

# 2 PCB design of the vf2TDC module

Vf2TDC is designed as a VXS payload board. It also works in a standard 6U VME64 crate. It receives 64 any-level differential signals and 128 LVTTL signals. The 128 LVTTL signals are dependent on the mezzanine cards and VME P2 IO card. Figure 2 shows a diagram of the vf2TDC board:



Figure 2 vfTDC fuctional diagram

The Xilinx Artix-7 XC7A200T-2FFG1156C FPGA is chosen for signal processing. It is less expensive, and it has enough IO and resources for signal processing. The Micron N25Q256 serial flash is used to save the FPGA configure data. This memory supports 4-bit wide FPGA load.

Two sets of four LEDs are used to indicate the board status, which is directly from the FPGA. The first set of the LEDs are ‘board ready’, ‘VME DTACK’, ‘Trigger’ and ‘Board error/Reset’. The second set of the LEDs indicate the board working modes.

One 3M 3408-D302 connector and two QSFP connectors are also loaded on the front panel. The 3M 3408-D302 connector has 8 generic differential signal inputs, and 8 ECL outputs, which are connected to the FPGA through TI SN65LVDT352 receiver and On-Semi MC100EP91 driver respectively. One QSFP connector has 4 MGTs (Multi-Gigabit Transceivers) connected to the FPGA directly, which is capable of more than 3 Gbps each. The other QSFP is compatible with Trigger Interface (TI). Depending on the application, this QSFP can accept the TI input from a TD/TImaster, or drive a TI for a small self-contained test setup.

An external memory, IDT71T75802 (18 Mbit) is added to expand the FPGA memory capacity.

The VME64x is implemented, which uses the Geographic address and +3.3V supply. The board is compatible with standard VME64 backplane. If the board is plugged in the VME64 crate, the onboard DC-DC converter can be used to get the +3.3V power from the +5 supply. Five bits of the on-board switch are used to set the A24 (A23-A19) address space.

**2.1 FPGA programming**

The FPGA XC7A200T needs up to 80Mbit to configure (, but for a typical design, it is much smaller). This configure data is saved in the Micron N25Q256, which is 256 Mbit. The memory can save two versions of the FPGA configure file. This can supply a fall back design, which is especially useful when the board is used in an area that is difficult to access (need be tested). But as the FPGA/Flash memory can be programmed remotely, this feature (two versions of firmware) is not much useful.

The FPGA is programmed in Master SPI mode with external clock of 50 MHz and 4-bit wide data loading. The expected FPGA program time is less than one second. The Micron memory can be loaded by the iMPACT software through the on-board JTAG connector. The iMPACT software will load a special firmware to the FPGA through the JTAG connector and program the memory through the special firmware, which the Xilinx calls as indirect flash memory programming.

The vf2TDC can be remotely programmed through VME. To make it more robust for remote programming, a hardware (discrete logic) VME to JTAG engine is implemented on the board (copied from TS/TI/TD design) using the custom defined address modifier code (AM = 19), which will not get confused with the standard (VME specified) A24 address modifier codes. This engine can load the FPGA firmware even if the memory is corrupted (or simply, the memory is empty) or the FPGA failed to be loaded by the memory. The engine has been tested successfully. It takes about 30 seconds to load the FPGA through VME. It takes about ten minutes to load the Flash memory (Micron N25Q256). In the JTAG engine, the VME data bit#1 is used for TDI, bit#0 is used for TMS, and all the other bits are unused. The higher bits (A[23:19]) of A24 address should match with the geographic address, and the lower A24 address (A[18:0]) is set to be 0x0FFFC.

For rev2 VETROC board (serial number >= 3), two Micron N25Q256 flash memories are implemented. A 3-position switch is used to choose which memory will be used. When the switch is set to the LEFT, it is the vf2TDC firmware, when the switch is set to the RIGHT, it is the trigger logic firmware developed by users, when the switch is in MIDDLE, neither firmware is chosen, and the FPGA will not be programmed.

**2.2: Clock Distribution**

There are three main clock sources for the rev1 PCB, and four clock sources for rev2 PCB. As a VXS payload board, it gets the clock (250 MHz) via VXS P0 backplane from SD/TI board. For the test or application without the VXS crate, an on-board oscillator (250MHz) is implemented for rev1 PCB and two on-board oscillators are implemented for rev2 PCB. The last source is the optional front panel TI fiber. Only one clock source is selected as the FPGA clock, which pipelines the trigger and readout logic. The clock source is selected by FPGA and buffered by the cross-point switch Micrel SY58040. The SY58040 will send three clocks to the FPGA (one for the FPGA internal logic, one for the north MGT blocks, and the other for the south MGT blocks), and one clock to the front panel QSFP when the TI interface is used.

**2.3 VME interface**

The vf2TDC board is a VXS payload slot board. It is compatible with VME64x backplane. Normally, it is a VME slave board, with interrupt capability.

The vf2TDC can also be a master VME board. It supports single level bus request (BR3, level 3) only, as we do not expect many boards to be a VME master in the crate. The VME master capability has not been tested (or implemented) yet.

For simplicity, three kinds of VME address modifier codes are implemented. (1), User defined address modifier. (0x19, 0x1A, 0x1C and 0x1D) This is similar to the A24 address modifier. It is used to load the FPGA by the onboard discrete logic (also called emergency JTAG engine). (2), Standard A24 address modifier. This is used to readout the registers on the FPGA, slow controls of the board. (3), A32 data transfer. This is used to transfer data to the ROC (Read Out Controller). This is implemented the same way as other ADC/TDC board. With token passing, the ROC needs only one read to get all the front end boards’ data out for higher efficiency.

**2.4: Readout logic**

The readout is initiated by the readout trigger, which can be from TI via SD and VXS P0 backplane, or TD/TImaster via the front QSFP trigger interface, or the front panel differential input (GenIn#7, pin#29/30). The TDC data are packed in data blocks, with each block contains one to 255 triggers. The data can be readout via VME A32, front panel QSFP fiber interface, or VXS P0 switch board (the QSFP and VXS paths need be implemented and tested).

**2.5: Trigger logic (front panel IO)**

Some simple trigger logic is added in the latest firmware (V7.2 and later). The input signals are rising edge detected (the polarity is inverted back to the front panel differential input), masked (by vme registers 0x60-0x74), and shaped/synced to the 250 MHz clock. All the 32 channels are ORed together to form a connector\_level trigger signal. The connector\_level ORed trigger signals are output at GenOut(6:1). Then all the six connector\_level trigger signals are ORed and shaped to the set width (by vme register 0x78 bits15-8), and output at GenOut(7). The FPGA also ORs the board\_level trigger signal with the external input (GenIn#5, masked by vme register 0x78 bit#4) to form the output at GenOut(8).

This way, the user has the flexibility of using output(6:1) for finer matching, and output(7) for simpler matching in high level trigger logic.

**2.6: Trigger logic (Switch Slot#A, Serial links, formware A9.1 and later)**

The channel hit (rising edge only, for now) timing are streamed to the VTP via the four serial link at 6.25 Gbps per lane. The channel hit time resolution is 8ns. Here are the ten words per 32ns time period:

|  |  |  |
| --- | --- | --- |
| Word#1 | Ch(63:0) hit valid, 64-bit | Top Mezz (63:32), Top on-board(31:0) |
| Word#2 | Ch(127:64) hit valid, 64-bit | Bottom Mezz (127:96), Bottom on-board(95:64) |
| Word#3 | Ch(191:128) hit valid, 64-bit | VIO\_bottom(191:160), VIO\_top(159:128) |
| Word#4 | Time-bit(0) of Ch(63:0), 64-bit |  |
| Word#5 | Time-bit(0) of Ch(127:64), 64-bit |  |
| Word#6 | Time-bit(0) of Ch(191:128), 64-bit | Channel-hit-time = Time-bit(1:0) \* 8 ns |
| Word#7 | Time-bit(1) of Ch(63:0), 64-bit |  |
| Word#8 | Time-bit(1) of Ch(127:64), 64-bit |  |
| Word#9 | Time-bit(1) of Ch(191:128), 64-bit |  |
| Word#10 | TDC data if any, 64-bit, | ”TDC0091” & Packet-bit(3:0) & TDCdata(31:0) |

For the 64 bits of TX word, bit(63:48), bit(47:32), bit(31:16) and bit(15:0) are mapped to Lane#4, lane#3, lane#2 and lane#1 respectively. On reset, the bits are standard IDLE (K28.5 and D16.2 pair). The word#10 is IDLE if no TDC data to be readout. The data are 8b/10b encoded. The total data rate is:

64 bitsPerWord \* 10 words / 8 bitPerByte / 32 ns 🡺 2.5 GB/s or 20 Gb/s

The serial link after 8b/10b encoding: 25 Gb/s, or 6.25 Gbps at 4 lanes.

Just a comment: the AMD xc7a200T-1 limit is 6.5 Gbps, the AMD xc7v-1 GTH limit is 8.5 Gbps, but the PCB routing and the VXS P0 connector may have further limit on the achievable rate.

The slower data rate thought: 1. Send the data every 64 ns, with 16 ns hit-time resolution 🡺 3.125 Gbps per lane; 2. Send the data every 64 ns, with 4 ns hit-time resolution 🡺 5 Gbps per lane.

# 3 TDC and FPGA design

**3.1 FPGA based TDC design**

The FPGA can be used as a Flash TDC. The input signals are measured in two steps. A free running clock is used to measure the coarse time of the signal (how many clock cycles); and the FPGA carry chain delay is used to measure the fine delay relative to the clock edge. After combining the coarse measurement and the fine measurement, the TDC measurement can reach a precision of 10s ps and a range of many microseconds.

The TDC fine measurement is based on the FPGA carry chain delays [3]. The delay unit (carry chain) is about 20 ps, which depends on the FPGA and its speed grade. Figure 3 shows the design of one TDC channel.



Figure 3 Diagram of one TDC channel

Each channel uses 32 FPGA slices (128 delay elements) for fine delay measurement, one 18 Kbits ring buffer and one 18 Kbits buffer for data storage. The ring buffer will not cause any dead time, but limits the trigger look back to ~4 us. The data buffer can hold several hundred pulses (rising edge and falling edges), which should not cause any dead time if the occupancy is not too high.

When the readout trigger arrives, the trigger time is recorded (in 4ns position), and a look back window is opened (the look back time and the width were set by an VME registers). The data is readout from the ring buffer, and stored in the channel readout buffer. The data is further merged in three stages. If there is no pulse during the readout window, the data will be suppressed. Figure 4 shows the diagram of the TDC data readout.



Figure 4 vf2TDC data flow: data merging and data readout

The data are merged event by event in three stages. The first stage merges eight (8) TDC channels, and stores the data in a 2 K deep buffer. The second stage merges four buffers from the first stage. The second stage data corresponds to a full connector, which is a group. The third stage merges six buffers from the second stage, which includes all the 192 channels of the vf2TDC.

**3.2 vf2TDC data format**

The vf2TDC supports blocking mode to increase the VME readout efficiency. Each block can have one (1) to 255 events (triggers), which is set by a VME register.

The data are formatted in blocks. Each block has N events (defined as the block level), and wrapped by the two block headers and one block trailer. An extra word may be added to make sure that the block is aligned in 64-bit boundary. For 2ESST VME readout, two more words may be padded to the block for the block data to be aligned in 128-bit boundary. Table 1 show the data format.

Table 1 vf2TDC data format

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Type | Data format | | | | | | | | | | | |
| Block header1 | Bit(31:27) 10000 | | | | Bit(26:22) Slot# (1-21) | | Bit(21:18) board ID: 0x9 | | Bit(17:8) Block # | Bit(7:0) Block level (1-255) | | |
| Event 1 | Header | Bit (31:27) | Bit(26:22) | | | Bit(21:0) | | | | | | |
| 10010 | Slot#(1-21) | | | Event number Nevt(21:0) | | | | | | |
| TimeTrgTime#1 | Bit(31:24)10011000 | | | | Bit(23:0)Time stamp T(23:0) | | | | | | |
| Time#2 | 00000000 | | | | Time stamp T(47:24) | | | | | | |
| Data 1 | Bit(31:27) | | | (26:24) | (23:19) | | (18) | (17:8) | | (7) | (6:0) |
| Xxxxx 10111 | | | Group#0-6 | Chan#0-31 | | EdgeType 1: R; 0: F | Coarse T 4\*(0-1023) | | 2ns 0-1 | Fine Time 0-127 |
| Data 2… | 10111 | | Same as above………….. | | | | | | | | |
| Event 2… | Same as above | | | | | | | | | | | |
| Block Trailer | Bit(31:27) 10001 | | | Bit(26:22) Slot# (1-21) | | | | Bit(21:0) Word count (no Block header/trailer) | | | | |
| Filler\_A | 11111 | | | Slot# (1-21) | | | | Block number | | | | |
| Filler\_B | 11111 | | | Slot# (1-21) | | | | 0011110001000100010000 (0F1110) | | | | |
| Filler\_B | 11111 | | | Slot# (1-21) | | | | 0011110001000100010000 (0F1110) | | | | |

# \*Filler\_A to align the data in 64-bit boundary, Filler\_B to align the data in 128-bit boundary.

\* for 192 Channel case (firmware highest bits = 3 [0’b11]): Group#: 0 for reference channels inputs (Channel 0-3 corresponds to reference input #1-4, and Chan 4-7 are duplicate of input#1-4), 1 for connector A (on-board upper Differential input), 2 for connector B (upper LVTTL input), 3 for connector C (on-board lower differential input), 4 for connector D (lower LVTTL input), 5 for connector E (VME P2 row#A input), 6 for connectorF (VME P2 row#C input).

\*for 96 channel case (firmware highest bit = 1): Group#: 0 for reference channels input, 1 for connector A Channel 1-16, and duplicate of Channel 1-16; 2 for connector A channel 17-32, and duplicate 17-32; 3 for connector C channel 1-16 and duplicate 1-16; 4 for connector C channel 17-32 and duplicate of 17-32; 5 for connector E channel 1-16 and duplicate 1-16; 6 for connector E channel 17-32 and duplicate 17-32. The connector B, D and F are not used.

\*\* The differential inputs (The on-board connectors, the backplane IO card, and the mezzanine boards) are polarity inverted by the differential receiver (TI SN65LVDT352) as the TTL output is high if the differential input is 0V. The TDC data bit#18 edge is really the opposite of the differential signal to the PCB. So, 0 is for rising edge of the input/PCB, falling edge of the TTL/FPGA signal.

**3.3 vf2TDC data readout**

The vf2TDC can interface with up to eight ReadOut Controllers (ROC) simultaneously. Each ROC interface has its own secondary data buffer. The vf2TDC will copy its data to all the ROC interface buffers. These paths include, but not limited to, VME readout, front panel fiber readout, and VXS P0 switch slot#A readout.

So far, only the VME readout has been tested on the vf2TDC board. And two ROCs readout was tested on the TI.

The VME readout FIFO structure is shown in the following figure. The data buffer is about 21K words (4-bytes) at the event level with additional TDC storage at per channel, per 8-channel group, and per 32-channel connector levels.



# 4. Specification Sheet

**4.1 Mechanical**

* Single width VITA 41 Payload Module. It will be positioned in PP1-PP16 in VXS crate; it can also be plugged into any slots in standard VME crates without P0 connector.

**4.2 High speed serial P0 inputs and outputs:**

* Switch slot#A (CTP) four lane MGT connections
* Switch slot#B (SD) compatible connections.

**4.3 Front panel inputs and outputs:**

* 2 x 32 differential signal detector inputs;
* 8 generic differential signal inputs;
* 8 generic ECL outputs.
* 4 channels of MGT on one QSFP, TI or TImaster fiber IO on the other QSFP.
* 2 optional mezzanine board with up to 32 channels (LVTTL) each.

**4.4 LED Indicators: Front Panel (**FPGA controlled**):**

* Set #1:
  + Bit 1: (close to the PCB): FPGA programmed and the clock (DCM locked) is ready;
  + Bit 2: VME DTACK, VME activity;
  + Bit 3: Readout trigger is detected;
  + Bit 4: MGT Rx error;
* Set #2: (to be implemented)
  + Bit 1: On board IO activity;
  + Bit 2: Mezzanine board activity (on: mezzanine card plugged, flash: activity);
  + Bit3: VME P2 IO card activity (on: mezzanine card plugged, flash: activity);
  + Bit 4: TI interface activity.

**On board:**

* Power OK near each regulator and DC-DC converter (The LED is OFF when the power is OK);
* FPGA program DONE (The LED is OFF when programmed);

**4.5 Programming:**

* VME to JTAG A24D32 with user defined AM (Address Modifier) for remote FPGA firmware loading, and Flash memory (Micron N25Q256) programming.
* onboard JTAG connector to FPGA;
* Up to two revisions of the FPGA firmware can be stored in the memory simultaneously.

**4.6 Power requirements:**

* +5v @ 1 Amps; -12V @ 0.25 Amp; +3.3V @ 2 Amps
* With Optional DC-DC converters for +3.3V, +5V @ 3A, +3.3V is not required from backplane.
* Local regulators for other required voltages: +1.0V, +1.2V, +1.8V, +2.5V, and -5V.

4.7 Environment:

* Forced air cooling;
* Commercial grade components ( 0-75 Celsius or better)

# 5 vf2TDC operation procedures:

The vf2TDC needs to be properly set, and plugged into the proper crate and slot. Damage may happen to the vf2TDC, the crate, or other PCBs in the crate if the right procedure is not followed.

5.1 vf2TDC Power supply:

The vf2TDC can use +3.3V directly from VME64x crate. It can also generate its own +3.3V supply by a DC-DC converter from +5V. Proper settings are needed to avoid damage to the board or backplane.

If the VME64x crate +3.3V power is used for the vfTDC:

(1). Fuse, FG1 is stuffed;

(2). DC-DC converter UP2 is removed.

If the VME64x crate +3.3V power is not used, or +3.3V is not available from the backplane:

(1). Fuse, FG1 is removed;

(2). UP2 is stuffed.

The default setting for the vf2TDC is assuming that there is no +3.3V from the backplane.

* 1. FPGA program mode setting:

The FPGA program can be set to MasterSPI mode or JTAG mode. For MasterSPI mode:

(1). Remove RBJ3;

(2). Load RBJ4.

For JTAG mode:

(1). Remove RBJ4;

(2). Load RBJ3.

* 1. FPGA firmware selection on power up:

There are two SPI flash memory chips on board to save two versions of FPGA firmware on rev2 PCB. The firmware is selected by an onboard 3-position switch. The three switch settings:

LEFT: the vf2TDC firmware is selected, (required for vf2TDC functions)

MIDDLE: no flash is selected, the FPGA will not be programmed by the flash memories.

RIGHT: User firmware is selected, (For the VETROC board, it defaults to Ben’s trigger firmware)

5.4 vf2TDC 8-bit switch S2 setting:

Bit[8:4]: set the VME A24 address space A[23:19] when the vf2TDC is in non-VME64x crate. If it is in VME64x crate, the geographic address is used, the switch is not used.

Bit[2:1] is connected in a way, that the FPGA can drive it using LVTTL, and override the switch default setting for a LVPECL level selector. Meanwhile, if the FPGA is a receiver, the switch setting is compatible with LVTTL/LVCMOS.

5.5 VME to JTAG discrete logic:

For standard A24 address modifier (0x39 etc.), load RB41 and remove RB42; For user defined address modifier (0x19 etc.), load RB42 and remove RB41.

6. VME Programming Requirements (This part will be updated as the firmware develops)

The vf2TDC supports three categories of Address Modifier codes: the user-defined codes (A24) for emergency firmware loading; Standard A24 for FPGA register read/write and slow control; A32 block transfer for VME data readout.

6.1 VME to JTAG emergency loading:

The AM[5:0] user defined codes are used for this logic. This works even before the FPGA is programmed and working. It is almost the same as A24D32 mode. The valid AM codes are: 0x19, 0x1A, 0x1D and 0x1E. These AM codes are user defined, and similar to the AM codes 0x39, 0x3A, 0x3D and 0x3E.

The valid address bits are A[31:24] do not care; A[23:19]=GA[4:0] for VME64x crates, or A[23:19]=0 for non-VME64x crates; A[18:2]=0b’00011111111111111.

VME Data bit[1] is TDI; VME data bit[0] is TMS.

For example, if the board is in slot#5 (that is ~GA(4:0)= 11010), you need write to A(23:0)=0x28fffc. If data(1:0)=00, both TMS and TDI will be low; if data(1:0)=01, TMS is high, TDI is low; if data(1:0)=10, TMS is low, TDI is high; if data(1:0)=11, both TDI and TMS are high. The normal A24 address should try to avoid this address (0x0fffc).

A more advanced example: Instruction register shift (8-bit, shift in 0x5a) starting from/end up at the ‘reset idle’ mode: 14 consecutive writes to the address 0x28fffc with AM=0x19, 1a, 1d or 1e, the data are 1, 1, 0, 0, 0, 2, 0, 2, 2, 0, 2, 1, 1, 0 respectively.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Data | 1 | 1 | 0 | 0 | 0 | 2 | 0 | 2 | 2 | 0 | 2 | 1 | 1 | 0 |
| TMS | H | H | L | L | L | L | L | L | L | L | L | H | H | L |
| TDI | 0x | 0x | 0x | 0x | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0x | 0x |

* “TMS H” means logic High, “TMS L” means logic Low, “TDI 0” means 0 or Low, “TDI 1” means 1 or High, and “TDI 0x” means DO NOT CARE by the JTAG, but the set value is 0.

6.2 Configuration Registers:

A24D32 are used for register read/write. Similar to the emergency loading logic, the base address is determined by the Geographic Address in VME64x crate, and external switch for non-VME64x crate. That is, A[23:19]=GA[4:0], or SW[8:4].

* Address offset: 0x00000: Board ID:

Bit 7-0 (R/W): Crate ID; Reset default 0x00;

Bit 12-8 (R): A24 address, higher 5 bits; Reset default 000

Bit 31-16 (R): board ID: 0xF7DC: vfTDC;

* Address offset: 0x00004: vfTDC readout window setting:

Bit 15-0 (R/W): width of the TDC readout window (in 4 ns steps). But as the readout cannot deal with overlap events properly, try to limit the readout width to 8 bits (~1 us, or trigger at ~1 MHz) or less. Be sure to set the trigger rule#1 on TS (or TI\_master) to match with the width setting (no overlap events).

* Address offset: 0x00008: Interrupt setting:

Bit 7-0 (R/W): Interrupt ID; Reset default 0xC8 [0x08, BIT(7:0)]

Bit 10-8 (R/W): Interrupt level; Reset default 5;

Bit 16 (R/W): IRQ enable. Reset default: 0;

* Address offset: 0x0000C: Trigger delay:

Bit 9-0 (R/W): Look back since the readout trigger;

Bit 14 (R/W): Rising edge TDC measurement/readout disable (1 to disable). If this bit is set to ‘1’, the pulse (TTL as seen by the FPGA) rising edge TDC will not be readout. As the PCB differential inputs are inverted by the TI receiver, the differential (as seen by the PCB) falling edge is disabled.

Bit 15 (R/W): Falling edge TDC measurement/readout disable (1 to disable).

Generally, Bit#15 is set to 0, bit#14 is set to 1 to decrease the TDC data size, as the detector pulse rising edge is more useful (than falling edge).

* Address offset: 0x00010: A32 address space:

Bit 13-5 (R/W): Address Max; Reset default 0x1FF;

Bit 22:14 (R/W): Address Min; Reset default 0x000;

Bit 31-23 (R/W): Base Address. Reset default 0x100;

* Address offset: 0x00014: Block size:

Bit 7-0 (R/W): Block size. Reset default 0x01;

* Address offset: 0x0001C: VME setting; Reset default 0x011:

Bit 0 (R/W): ‘1’ enable Bus\_Error\_En, so the block read can be terminated by event block trailer;

Bit 1 (R/W): ‘1’ en\_token\_in is true, ‘0’ en\_token\_in is false;

Bit 2 (R/W): ‘1’ enable ‘Multi-board’ readout, ‘0’ disable ‘Multi-board’; asset to enable multi-board token passing protocol;

Bit 3 (R/W): ‘1’ enable en\_A32m, ‘0’ disable en\_A32m; assert to enable common A32 multi-board addressing of module;

Bit 4 (R/W): ‘1’ enable en\_A32, ‘0’ disable en\_A32;

Bit 7 (R/W): ‘1’ enable VME bus interrupt for module error?

Bit 8 (R/W): ‘1’ I2C device address 0x1101xxx, ‘0’ I2C device address 0x0000xxx;

Bit 9 (R/W): ‘1’ token\_in high, ‘0’ token\_in low; If both bit 9 and bit 1 are set high, the Token\_Out will be high (this is for SD test).

Bit 10 (R/W): ‘1’ first\_board true, ‘0’ first\_board false;

Bit 11 (R/W): ‘1’ last\_board true, ‘0’ last board false;

Bit 15 (R/W): ‘1’ disable data readout buffer full;

* Address offset: 0x00020: Trigger source register:

Bit 15-0 (R/W): Trigger source enables: Reset default 0x0000;

Bit 0: P0 trigger input;

Bit 1: HFBR#1 trigger input;

Bit 3: Front Panel trigger input;

Bit 4: VME trigger, calibration trigger;

Bit 7: Random Trigger.

Bit 31-16 (R): Trigger source monitor.

* Address offset: 0x00024: Sync Source register:

Bit 15-0 (R/W): Sync Source enables: Reset default 0x02;

Bit 0: P0 sync input (in Subsystem TS mode);

Bit 1: HFBR#1 sync input;

Bit 3: Front panel SyncReset enable;

Bit 4: VME syncReset;

Bit 31-24 (R): Sync source monitoring.

* Address offset: 0x00028: Busy source registers:

Bit 15-0 (R/W): Busy source enables:

Bit 0: ‘1’ enable the Switch Slot #A BUSY input, ‘0’ disable;

Bit 1: ‘1’ enable the Switch Slot #B BUSY input, ‘0’ disable;

Bit 2: ‘1’ enable the VME P2 BUSY input, ‘0’ disable;

Bit 3: ‘1’ enable the FTDC front panel BUSY input, ‘0’ disable;

Bit 4: ‘1’ enable the FADC front panel BUSY input, ‘0’ disable;

Bit 5: ‘1’ enable the Front Panel BUSY, which is the same as TsRev2 busy;

Bit 7: ‘1’ enable TS feed\_back BUSY, ‘0’ disable the busy. (useful in TM mode)

Bit 15-8: HFBR #8-#1 BUSY enables: ‘1’ enable the HFBR BUSY input, ‘0’ disable;

Bit 31-16 (R): FIFO full monitoring

Bit 16: FIFO full

* Address offset: 0x0002C: Clock source selection:

Bit 1-0 (R/W): software bit switch to control the clock source for ClkOut#0 (TI\_out). Reset default 00;

Bit[1:0] = 00: QSFP TI front panel clock;

Bit[1:0] = 01: on-board 250 MHz oscillator for Rev2 VETROC, no use for Rev1 board;

Bit[1:0] = 10: on-board programmable 125 MHz oscillator for Rev2, 250Mhz (only osc) for Rev1;

Bit[1:0] = 11: VXS P0 switch slot#B clock;

Bit 3-2 (R/W): software bit switch to control the clock source for ClkOut#1 (MGT Ref). Reset default 00;

Bit 5-4 (R/W): software bit switch to control the clock source for ClkOut#2 (FPGA). Reset default 00;

Bit 7-6 (R/W): software bit switch to control the clock source for ClkOut#3 (MGT Ref). Reset default 00;

Bit 15-8 (R): The actual clock source selection shifted down to the Clock chip (Micrel SY89540U).

* Address offset: 0x00030: Trigger1 scalars:

Bit 31-0 (R): Number of Trigger1 received (enabled by trigger source selection);

* Address offset: 0x0004C: Blocks for VME interrupt:

Bit 15-8 (R): Number of data blocks in the FIFO in VME block (VME readout).

Bit 23-16 (R): Number of data blocks ready for Interrupt Request.

Bit 31-24 (R): on TI: Number of events of a partial block (or, before the block is formed)

On TS: bit(15:8) of the number of data blocks ready for VME interrupt.

* Address offset: 0x00050: Trigger2 scalar:

Bit 31-0 (R): Number of Trigger 2 received;

* Address offset: 0x00054: Sync\_reset scalar:

Bit 31-0 (R): Number of Sync\_reset received by the vf2TDC;

* Address offset: 0x00058: Berr scalar:

Bit 31-0 (R): Number of Berr from the VME module;

* Address offset: 0x0005C: Board status:

Bit 0 (R): VME BusError;

Bit 1 (R): VME Token;

Bit 2 (R): VME BusErr\_N;

Bit 3 (R): VME Take\_Token;

Bit 4 (R): VME Read\_Token\_Out;

Bit 5 (R): VME Done\_Block;

Bit 6 (R): VME BusErr\_Status;

Bit 8 (R): VME First\_level\_buffer\_Full\_A;

Bit 9 (R): VME First\_level\_buffer\_Full\_B (should be the same as Bit#8);

Bit 10 (R): VME First\_level\_buffer\_Empty\_A;

Bit 11 (R): VME First\_level\_buffer\_Empty\_B (should be the same as Bit#10);

Bit 12 (R): VME Second\_level\_buffer\_Full\_A;

Bit 13 (R): VME Second\_level\_buffer\_Full\_B (should be the same as Bit#12);

Bit 14 (R): VME Second\_level\_buffer\_Prog\_Full\_B (almost full);

Bit 15 (R): VME Second\_Level\_Buffer\_Empty\_B;

Bit 16 (R): VME First\_Level\_Buffer\_Prog\_Full\_A (almost full);

Bit 23-20 (R): Firmware revision number;

Bit 29-24 (R): Firmware version number;

Bit 31-30 (R): Firmware type, “11” for 192 (6\*32) channels,

“10” for 144 (6\*24) channels,

“01” for 96 (6\*16) higher resolution channels;

“00” for Stream (trigger-less) TDC readout mode;

* Address offset: 0x00060: Connector#A (top onboard front panel) inputs OR logic mask:

Bit 31-0 (R/W): 32-channels Connector#A inputs mask, 0 to mask off the channels;

* Address offset: 0x00064: Connector#B (Top mezzanine board) inputs OR logic mask:

Bit 31-0 (R/W): 32-channels Connector#B inputs mask, 0 to mask off the channels;

* Address offset: 0x00068: Connector#C (bottom onboard front panel) inputs OR logic mask:

Bit 31-0 (R/W): 32-channels Connector#C inputs mask, 0 to mask off the channels;

* Address offset: 0x0006C: Connector#D (bottom mezzanine board) inputs OR logic mask:

Bit 31-0 (R/W): 32-channels Connector#D inputs mask, 0 to mask off the channels;

* Address offset: 0x00070: Connector#VA (VME P2 row#A/D connector) inputs OR logic mask:

Bit 31-0 (R/W): 32-channels Connector#VA inputs mask, 0 to mask off the channels;

* Address offset: 0x00074: Connector#VC (VME P2 row#C/D connector) inputs OR logic mask:

Bit 31-0 (R/W): 32-channels Connector#VC inputs mask, 0 to mask off the channels;

* Address offset: 0x00078: Reference channel inputs OR logic mask:

Bit 3-0 (R/W): 4 reference channel inputs mask, 0 to mask off the channels. These four bits are not used.

Bit 4(R/W): The extra input (front panel #5, pin#25/26) OR mask enable;

Bit 15:8 (R/W): The OR output width setting. (in 4 ns steps). This only affects GenOut#7.

* Address offset: 0x0009C (R/W): The FPGA running mode;

Bit 7-0: vfTDC running mode setting. Reset default 0x00;

0xF7: vfTDC is in running mode, no A24 registers write is permitted;

0xF8: Set the VME P2 28 rowA inputs and 4 rowD inputs (total 32) to internal calibration mode;

0xF9: Set the VME P2 28 rowC inputs and 4 rowD inputs (total 32) to internal calibration mode;

0xFA: Set the Front panel connector #A inputs (total 32) to internal calibration mode;

0xFB: Set the Front panel mezzanine connector #B inputs (total 32) to internal calibration mode;

0xFC: Set the Front panel connector #C inputs (total 32) to internal calibration mode;

0xFD: Set the Front panel mezzanine connector #D inputs (total 32) to internal calibration mode;

*Since vfTDC V3.4 (June 3, 2015)*

*Bit(7:6): Calibration signal selection:*

*00: logic low;*

*01: PreScaled ClkDlyRef with fine delay;*

*10: PreScaled ClkVme*

*11: External Calibration signal in.*

*Bit(5:0): Connector (group) to set to choose calibration signal (calibration mode)*

*XXXXXX: when 1, set to calibration mode. For VmeC, VmeA, D, C, B and A respectively.*

* Address offset: 0x000A8 (R): Trigger live timer:

Bit 31-0 (r): board live time counter. The real time is Bit(31:0)\*256\*30ns. (ScalarLatch is required.)

* Address offset: 0x000AC (R): Trigger busy (trigger dead) timer:

Bit 31-0 (r): TID busy (cannot accept trigger, or trigger dead) time counter. The real time is Bit(31:0)\*256\*30ns. This counter and the live time counter make up the total time counter, which is the total time since any one of the trigger sources is enabled.

* Address offset: 0x000D8 (R): Event number register

Bit 31-16: higher 16-bit (bit 47-32) of event number counter;

* Address offset: 0x000DC (R): Event number register

Bit 31-0: lower 32-bit (bit 31-0) of event number counter.

* Address offset: 0x000EC (R/W): ROC enable

Bit 7-0: ROC 8:1 enable, the default is 00000001

* Address offset: 0x00100 (W): Reset and one-shot registers. The signal will be one ClkVme cycle. If the ClkVme is 50 MHz, the one-shot will be 20ns wide. Positive logic.

Bit 0: not used;

Bit 1: if ‘1’, RESET signal to reset the VME\_to\_I2C engine;

Bit 4: if ‘1’, RESET signal to reset the VME registers (TID settings) to their default values;

Bit 5: if ‘1’, SyncReset, serves as the generic VME reset;

Bit 7: if ‘1’, this register will generate a BUSY reset, and Trg\_Ack pulse.

Bit 8: if ‘1’, Reset the CLK250/Clk200 DCM.

Bit 10: if ‘1’, Reset the MGT (MultiGigabit Transceiver,) inside the FPGA.

Bit 11: if ‘1’, Auto alignment of SYNC phase from HFBR#1; auto align P0 sync input for TD.

Bit 12: if ‘1’, generate a calibration trigger;

Bit 14: if ‘1’, Reset the IODELAY;

Bit 16: if ‘1’, this register will generate a ‘TAKE\_TOKEN’

Bit 17: if ‘1’, the available number of data blocks will decrease by 1,

Bit 24: if ‘1’, all the trigger input scalars are latched (ready for read out), the BusyTimer and LiveTimer are also latched;

Bit 25: if ‘1’, all the trigger input scalars are reset. (Bit 24 and Bit 25 can be set simultaneously). The event number is also reset by this.

* Address offset: 0x0100 (R): Connector#A (top on board) Channel#1 scaler 32-bit;
* Address offset: 0x0104 (R): Connector#A (top on board) Channel#2 scaler 32-bit;
* Address offset: 0x0108 (R): Connector#A (top on board) Channel#3 scaler 32-bit;
* Address offset: 0x010C (R): Connector#A (top on board) Channel#4 scaler 32-bit;
* Address offset: 0x0110 (R): Connector#A (top on board) Channel#5 scaler 32-bit;
* Address offset: 0x0114 (R): Connector#A (top on board) Channel#6 scaler 32-bit;
* Address offset: 0x0118 (R): Connector#A (top on board) Channel#7 scaler 32-bit;
* Address offset: 0x011C (R): Connector#A (top on board) Channel#8 scaler 32-bit;
* Address offset: 0x0120 (R): Connector#A (top on board) Channel#9 scaler 32-bit;
* Address offset: 0x0124 (R): Connector#A (top on board) Channel#10 scaler 32-bit;
* Address offset: 0x0128 (R): Connector#A (top on board) Channel#11scaler 32-bit;
* Address offset: 0x012C (R): Connector#A (top on board) Channel#12 scaler 32-bit;
* Address offset: 0x0130 (R): Connector#A (top on board) Channel#13 scaler 32-bit;
* Address offset: 0x0134 (R): Connector#A (top on board) Channel#14 scaler 32-bit;
* Address offset: 0x0138 (R): Connector#A (top on board) Channel#15 scaler 32-bit;
* Address offset: 0x013C (R): Connector#A (top on board) Channel#16 scaler 32-bit;
* Address offset: 0x0140 (R): Connector#A (top on board) Channel#17 scaler 32-bit;
* Address offset: 0x0144 (R): Connector#A (top on board) Channel#18 scaler 32-bit;
* Address offset: 0x0148 (R): Connector#A (top on board) Channel#19 scaler 32-bit;
* Address offset: 0x014C (R): Connector#A (top on board) Channel#20 scaler 32-bit;
* Address offset: 0x0150 (R): Connector#A (top on board) Channel#21 scaler 32-bit;
* Address offset: 0x0154 (R): Connector#A (top on board) Channel#22 scaler 32-bit;
* Address offset: 0x0158 (R): Connector#A (top on board) Channel#23 scaler 32-bit;
* Address offset: 0x015C (R): Connector#A (top on board) Channel#24 scaler 32-bit;
* Address offset: 0x0160 (R): Connector#A (top on board) Channel#25 scaler 32-bit;
* Address offset: 0x0164 (R): Connector#A (top on board) Channel#26 scaler 32-bit;
* Address offset: 0x0168 (R): Connector#A (top on board) Channel#27 scaler 32-bit;
* Address offset: 0x016C (R): Connector#A (top on board) Channel#28 scaler 32-bit;
* Address offset: 0x0170 (R): Connector#A (top on board) Channel#29 scaler 32-bit;
* Address offset: 0x0174 (R): Connector#A (top on board) Channel#30 scaler 32-bit;
* Address offset: 0x0178 (R): Connector#A (top on board) Channel#31 scaler 32-bit;
* Address offset: 0x017C (R): Connector#A (top on board) Channel#32 scaler 32-bit;
* Address offset: 0x0180 (R): Connector#B (top mezz.) Channel#1 scaler 32-bit;
* Address offset: 0x0184 (R): Connector#B (top mezz.) Channel#2 scaler 32-bit;
* Address offset: 0x0188 (R): Connector#B (top mezz.) Channel#3 scaler 32-bit;
* Address offset: 0x018C (R): Connector#B (top mezz.) Channel#4 scaler 32-bit;
* Address offset: 0x0190 (R): Connector#B (top mezz.) Channel#5 scaler 32-bit;
* Address offset: 0x0194 (R): Connector#B (top mezz.) Channel#6 scaler 32-bit;
* Address offset: 0x0198 (R): Connector#B (top mezz.) Channel#7 scaler 32-bit;
* Address offset: 0x019C (R): Connector#B (top mezz.) Channel#8 scaler 32-bit;
* Address offset: 0x01a0 (R): Connector#B (top mezz.) Channel#9 scaler 32-bit;
* Address offset: 0x01a4 (R): Connector#B (top mezz.) Channel#10 scaler 32-bit;
* Address offset: 0x01a8 (R): Connector#B (top mezz.) Channel#11scaler 32-bit;
* Address offset: 0x01aC (R): Connector#B (top mezz.) Channel#12 scaler 32-bit;
* Address offset: 0x01b0 (R): Connector#B (top mezz.) Channel#13 scaler 32-bit;
* Address offset: 0x01b4 (R): Connector#B (top mezz.) Channel#14 scaler 32-bit;
* Address offset: 0x01b8 (R): Connector#B (top mezz.) Channel#15 scaler 32-bit;
* Address offset: 0x01bC (R): Connector#B (top mezz.) Channel#16 scaler 32-bit;
* Address offset: 0x01c0 (R): Connector#B (top mezz.) Channel#17 scaler 32-bit;
* Address offset: 0x01c4 (R): Connector#B (top mezz.) Channel#18 scaler 32-bit;
* Address offset: 0x01c8 (R): Connector#B (top mezz.) Channel#19 scaler 32-bit;
* Address offset: 0x01cC (R): Connector#B (top mezz.) Channel#20 scaler 32-bit;
* Address offset: 0x01d0 (R): Connector#B (top mezz.) Channel#21 scaler 32-bit;
* Address offset: 0x01d4 (R): Connector#B (top mezz.) Channel#22 scaler 32-bit;
* Address offset: 0x01d8 (R): Connector#B (top mezz.) Channel#23 scaler 32-bit;
* Address offset: 0x01dC (R): Connector#B (top mezz.) Channel#24 scaler 32-bit;
* Address offset: 0x01e0 (R): Connector#B (top mezz.) Channel#25 scaler 32-bit;
* Address offset: 0x01e4 (R): Connector#B (top mezz.) Channel#26 scaler 32-bit;
* Address offset: 0x01e8 (R): Connector#B (top mezz.) Channel#27 scaler 32-bit;
* Address offset: 0x01eC (R): Connector#B (top mezz.) Channel#28 scaler 32-bit;
* Address offset: 0x01f0 (R): Connector#B (top mezz.) Channel#29 scaler 32-bit;
* Address offset: 0x01f4 (R): Connector#B (top mezz.) Channel#30 scaler 32-bit;
* Address offset: 0x01f8 (R): Connector#B (top mezz.) Channel#31 scaler 32-bit;
* Address offset: 0x01fC (R): Connector#B (top mezz.) Channel#32 scaler 32-bit;
* Address offset: 0x0200 (R): Connector#C (bottom on board) Channel#1 scaler 32-bit;
* Address offset: 0x0204 (R): Connector#C (bottom on board) Channel#2 scaler 32-bit;
* Address offset: 0x0208 (R): Connector#C (bottom on board) Channel#3 scaler 32-bit;
* Address offset: 0x020C (R): Connector#C (bottom on board) Channel#4 scaler 32-bit;
* Address offset: 0x0210 (R): Connector#C (bottom on board) Channel#5 scaler 32-bit;
* Address offset: 0x0214 (R): Connector#C (bottom on board) Channel#6 scaler 32-bit;
* Address offset: 0x0218 (R): Connector#C (bottom on board) Channel#7 scaler 32-bit;
* Address offset: 0x021C (R): Connector#C (bottom on board) Channel#8 scaler 32-bit;
* Address offset: 0x0220 (R): Connector#C (bottom on board) Channel#9 scaler 32-bit;
* Address offset: 0x0224 (R): Connector#C (bottom on board) Channel#10 scaler 32-bit;
* Address offset: 0x0228 (R): Connector#C (bottom on board) Channel#11scaler 32-bit;
* Address offset: 0x022C (R): Connector#C (bottom on board) Channel#12 scaler 32-bit;
* Address offset: 0x0230 (R): Connector#C (bottom on board) Channel#13 scaler 32-bit;
* Address offset: 0x0234 (R): Connector#C (bottom on board) Channel#14 scaler 32-bit;
* Address offset: 0x0238 (R): Connector#C (bottom on board) Channel#15 scaler 32-bit;
* Address offset: 0x023C (R): Connector#C (bottom on board) Channel#16 scaler 32-bit;
* Address offset: 0x0240 (R): Connector#C (bottom on board) Channel#17 scaler 32-bit;
* Address offset: 0x0244 (R): Connector#C (bottom on board) Channel#18 scaler 32-bit;
* Address offset: 0x0248 (R): Connector#C (bottom on board) Channel#19 scaler 32-bit;
* Address offset: 0x024C (R): Connector#C (bottom on board) Channel#20 scaler 32-bit;
* Address offset: 0x0250 (R): Connector#C (bottom on board) Channel#21 scaler 32-bit;
* Address offset: 0x0254 (R): Connector#C (bottom on board) Channel#22 scaler 32-bit;
* Address offset: 0x0258 (R): Connector#C (bottom on board) Channel#23 scaler 32-bit;
* Address offset: 0x025C (R): Connector#C (bottom on board) Channel#24 scaler 32-bit;
* Address offset: 0x0260 (R): Connector#C (bottom on board) Channel#25 scaler 32-bit;
* Address offset: 0x0264 (R): Connector#C (bottom on board) Channel#26 scaler 32-bit;
* Address offset: 0x0268 (R): Connector#C (bottom on board) Channel#27 scaler 32-bit;
* Address offset: 0x026C (R): Connector#C (bottom on board) Channel#28 scaler 32-bit;
* Address offset: 0x0270 (R): Connector#C (bottom on board) Channel#29 scaler 32-bit;
* Address offset: 0x0274 (R): Connector#C (bottom on board) Channel#30 scaler 32-bit;
* Address offset: 0x0278 (R): Connector#C (bottom on board) Channel#31 scaler 32-bit;
* Address offset: 0x027C (R): Connector#C (bottom on board) Channel#32 scaler 32-bit;
* Address offset: 0x0280 (R): Connector#D (bottom mezz.) Channel#1 scaler 32-bit;
* Address offset: 0x0284 (R): Connector#D (bottom mezz.) Channel#2 scaler 32-bit;
* Address offset: 0x0288 (R): Connector#D (bottom mezz.) Channel#3 scaler 32-bit;
* Address offset: 0x028C (R): Connector#D (bottom mezz.) Channel#4 scaler 32-bit;
* Address offset: 0x0290 (R): Connector#D (bottom mezz.) Channel#5 scaler 32-bit;
* Address offset: 0x0294 (R): Connector#D (bottom mezz.) Channel#6 scaler 32-bit;
* Address offset: 0x0298 (R): Connector#D (bottom mezz.) Channel#7 scaler 32-bit;
* Address offset: 0x029C (R): Connector#D (bottom mezz.) Channel#8 scaler 32-bit;
* Address offset: 0x02a0 (R): Connector#D (bottom mezz.) Channel#9 scaler 32-bit;
* Address offset: 0x02a4 (R): Connector#D (bottom mezz.) Channel#10 scaler 32-bit;
* Address offset: 0x02a8 (R): Connector#D (bottom mezz.) Channel#11scaler 32-bit;
* Address offset: 0x02aC (R): Connector#D (bottom mezz.) Channel#12 scaler 32-bit;
* Address offset: 0x02b0 (R): Connector#D (bottom mezz.) Channel#13 scaler 32-bit;
* Address offset: 0x02b4 (R): Connector#D (bottom mezz.) Channel#14 scaler 32-bit;
* Address offset: 0x02b8 (R): Connector#D (bottom mezz.) Channel#15 scaler 32-bit;
* Address offset: 0x02bC (R): Connector#D (bottom mezz.) Channel#16 scaler 32-bit;
* Address offset: 0x02c0 (R): Connector#D (bottom mezz.) Channel#17 scaler 32-bit;
* Address offset: 0x02c4 (R): Connector#D (bottom mezz.) Channel#18 scaler 32-bit;
* Address offset: 0x02c8 (R): Connector#D (bottom mezz.) Channel#19 scaler 32-bit;
* Address offset: 0x02cC (R): Connector#D (bottom mezz.) Channel#20 scaler 32-bit;
* Address offset: 0x02d0 (R): Connector#D (bottom mezz.) Channel#21 scaler 32-bit;
* Address offset: 0x02d4 (R): Connector#D (bottom mezz.) Channel#22 scaler 32-bit;
* Address offset: 0x02d8 (R): Connector#D (bottom mezz.) Channel#23 scaler 32-bit;
* Address offset: 0x02dC (R): Connector#D (bottom mezz.) Channel#24 scaler 32-bit;
* Address offset: 0x02e0 (R): Connector#D (bottom mezz.) Channel#25 scaler 32-bit;
* Address offset: 0x02e4 (R): Connector#D (bottom mezz.) Channel#26 scaler 32-bit;
* Address offset: 0x02e8 (R): Connector#D (bottom mezz.) Channel#27 scaler 32-bit;
* Address offset: 0x02eC (R): Connector#D (bottom mezz.) Channel#28 scaler 32-bit;
* Address offset: 0x02f0 (R): Connector#D (bottom mezz.) Channel#29 scaler 32-bit;
* Address offset: 0x02f4 (R): Connector#D (bottom mezz.) Channel#30 scaler 32-bit;
* Address offset: 0x02f8 (R): Connector#D (bottom mezz.) Channel#31 scaler 32-bit;
* Address offset: 0x02fC (R): Connector#D (bottom mezz.) Channel#32 scaler 32-bit;
* Address offset: 0x0300 (R): Connector#E (top on backplane IO card) Channel#1 scaler 32-bit;
* Address offset: 0x0304 (R): Connector#E (top on backplane IO card) Channel#2 scaler 32-bit;
* Address offset: 0x0308 (R): Connector#E (top on backplane IO card) Channel#3 scaler 32-bit;
* Address offset: 0x030C (R): Connector#E (top on backplane IO card) Channel#4 scaler 32-bit;
* Address offset: 0x0310 (R): Connector#E (top on backplane IO card) Channel#5 scaler 32-bit;
* Address offset: 0x0314 (R): Connector#E (top on backplane IO card) Channel#6 scaler 32-bit;
* Address offset: 0x0318 (R): Connector#E (top on backplane IO card) Channel#7 scaler 32-bit;
* Address offset: 0x031C (R): Connector#E (top on backplane IO card) Channel#8 scaler 32-bit;
* Address offset: 0x0320 (R): Connector#E (top on backplane IO card) Channel#9 scaler 32-bit;
* Address offset: 0x0324 (R): Connector#E (top on backplane IO card) Channel#10 scaler 32-bit;
* Address offset: 0x0328 (R): Connector#E (top on backplane IO card) Channel#11scaler 32-bit;
* Address offset: 0x032C (R): Connector#E (top on backplane IO card) Channel#12 scaler 32-bit;
* Address offset: 0x0330 (R): Connector#E (top on backplane IO card) Channel#13 scaler 32-bit;
* Address offset: 0x0334 (R): Connector#E (top on backplane IO card) Channel#14 scaler 32-bit;
* Address offset: 0x0338 (R): Connector#E (top on backplane IO card) Channel#15 scaler 32-bit;
* Address offset: 0x033C (R): Connector#E (top on backplane IO card) Channel#16 scaler 32-bit;
* Address offset: 0x0340 (R): Connector#E (top on backplane IO card) Channel#17 scaler 32-bit;
* Address offset: 0x0344 (R): Connector#E (top on backplane IO card) Channel#18 scaler 32-bit;
* Address offset: 0x0348 (R): Connector#E (top on backplane IO card) Channel#19 scaler 32-bit;
* Address offset: 0x034C (R): Connector#E (top on backplane IO card) Channel#20 scaler 32-bit;
* Address offset: 0x0350 (R): Connector#E (top on backplane IO card) Channel#21 scaler 32-bit;
* Address offset: 0x0354 (R): Connector#E (top on backplane IO card) Channel#22 scaler 32-bit;
* Address offset: 0x0358 (R): Connector#E (top on backplane IO card) Channel#23 scaler 32-bit;
* Address offset: 0x035C (R): Connector#E (top on backplane IO card) Channel#24 scaler 32-bit;
* Address offset: 0x0360 (R): Connector#E (top on backplane IO card) Channel#25 scaler 32-bit;
* Address offset: 0x0364 (R): Connector#E (top on backplane IO card) Channel#26 scaler 32-bit;
* Address offset: 0x0368 (R): Connector#E (top on backplane IO card) Channel#27 scaler 32-bit;
* Address offset: 0x036C (R): Connector#E (top on backplane IO card) Channel#28 scaler 32-bit;
* Address offset: 0x0370 (R): Connector#E (top on backplane IO card) Channel#29 scaler 32-bit;
* Address offset: 0x0374 (R): Connector#E (top on backplane IO card) Channel#30 scaler 32-bit;
* Address offset: 0x0378 (R): Connector#E (top on backplane IO card) Channel#31 scaler 32-bit;
* Address offset: 0x037C (R): Connector#E (top on backplane IO card) Channel#32 scaler 32-bit;
* Address offset: 0x0380 (R): Connector#F (bottom on backplane IO card) Channel#1 scaler 32-bit;
* Address offset: 0x0384 (R): Connector#F (bottom on backplane IO card) Channel#2 scaler 32-bit;
* Address offset: 0x0388 (R): Connector#F (bottom on backplane IO card) Channel#3 scaler 32-bit;
* Address offset: 0x038C (R): Connector#F (bottom on backplane IO card) Channel#4 scaler 32-bit;
* Address offset: 0x0390 (R): Connector#F (bottom on backplane IO card) Channel#5 scaler 32-bit;
* Address offset: 0x0394 (R): Connector#F (bottom on backplane IO card) Channel#6 scaler 32-bit;
* Address offset: 0x0398 (R): Connector#F (bottom on backplane IO card) Channel#7 scaler 32-bit;
* Address offset: 0x039C (R): Connector#F (bottom on backplane IO card) Channel#8 scaler 32-bit;
* Address offset: 0x03a0 (R): Connector#F (bottom on backplane IO card) Channel#9 scaler 32-bit;
* Address offset: 0x03a4 (R): Connector#F (bottom on backplane IO card) Channel#10 scaler 32-bit;
* Address offset: 0x03a8 (R): Connector#F (bottom on backplane IO card) Channel#11scaler 32-bit;
* Address offset: 0x03aC (R): Connector#F (bottom on backplane IO card) Channel#12 scaler 32-bit;
* Address offset: 0x03b0 (R): Connector#F (bottom on backplane IO card) Channel#13 scaler 32-bit;
* Address offset: 0x03b4 (R): Connector#F (bottom on backplane IO card) Channel#14 scaler 32-bit;
* Address offset: 0x03b8 (R): Connector#F (bottom on backplane IO card) Channel#15 scaler 32-bit;
* Address offset: 0x03bC (R): Connector#F (bottom on backplane IO card) Channel#16 scaler 32-bit;
* Address offset: 0x03c0 (R): Connector#F (bottom on backplane IO card) Channel#17 scaler 32-bit;
* Address offset: 0x03c4 (R): Connector#F (bottom on backplane IO card) Channel#18 scaler 32-bit;
* Address offset: 0x03c8 (R): Connector#F (bottom on backplane IO card) Channel#19 scaler 32-bit;
* Address offset: 0x03cC (R): Connector#F (bottom on backplane IO card) Channel#20 scaler 32-bit;
* Address offset: 0x03d0 (R): Connector#F (bottom on backplane IO card) Channel#21 scaler 32-bit;
* Address offset: 0x03d4 (R): Connector#F (bottom on backplane IO card) Channel#22 scaler 32-bit;
* Address offset: 0x03d8 (R): Connector#F (bottom on backplane IO card) Channel#23 scaler 32-bit;
* Address offset: 0x03dC (R): Connector#F (bottom on backplane IO card) Channel#24 scaler 32-bit;
* Address offset: 0x03e0 (R): Connector#F (bottom on backplane IO card) Channel#25 scaler 32-bit;
* Address offset: 0x03e4 (R): Connector#F (bottom on backplane IO card) Channel#26 scaler 32-bit;
* Address offset: 0x03e8 (R): Connector#F (bottom on backplane IO card) Channel#27 scaler 32-bit;
* Address offset: 0x03eC (R): Connector#F (bottom on backplane IO card) Channel#28 scaler 32-bit;
* Address offset: 0x03f0 (R): Connector#F (bottom on backplane IO card) Channel#29 scaler 32-bit;
* Address offset: 0x03f4 (R): Connector#F (bottom on backplane IO card) Channel#30 scaler 32-bit;
* Address offset: 0x03f8 (R): Connector#F (bottom on backplane IO card) Channel#31 scaler 32-bit;
* Address offset: 0x03fC (R): Connector#F (bottom on backplane IO card) Channel#32 scaler 32-bit;

6.3 VME data acquisition:

For data acquisition, the A32 block reads are used. The base address is set by the upper 9 bits of A24 register 0x00010, that is A[31:23] = RegData[31:23] of A24=0x00010.

7 Pin out tables:

7.1 VXS P0 Pinout Table

|  |  |  |  |
| --- | --- | --- | --- |
| Payload slot | | | |
| Pin name | Signal Description | Signal Level | Direction |
| DP1 (A1+, B1-) | CTPRX1 |  | 🡨 SWA |
| DP2 (D1+, E1-) | CTPTX1 |  | 🡪 SWA |
| DP3 (B2+, C2-) | CTPRX2 |  | 🡨 SWA |
| DP4 (E2+, F2-) | CTPTX2 |  | 🡪 SWA |
| DP5 (A3+, B3-) | CTPRX3 |  | 🡨 SWA |
| DP6 (D3+, E3-) | CTPTX3 |  | 🡪 SWA |
| DP7 (B4+, C4-) | CTPRX4 |  | 🡨 SWA |
| DP8 (E4+, F4-) | CTPTX4 |  | 🡪 SWA |
| SE1 (G1) | STAT\_OUT | LVTTL (+3.3V) | 🡪 SWA |
| SE2 (G3) | STAT\_IN | LVTTL (+3.3V) | 🡨 SWA |
| DP23 (B12+, C12-) | Readout TRIGGER | LVPECL(DP) | 🡨 SWB |
| DP24 (E12+, F12-) | SYNC | LVPECL(DP) | 🡨 SWB |
| DP25 (A13+,B13-) | CLOCK | LVPECL(DP) | 🡨 SWB |
| DP26 (D13+, E13-) | Trigger2 | LVPECL(DP) | 🡨 SWB |
| DP27 (B14+, C14-) | TOKEN\_IN | LVPECL(DP) | 🡨 SWB |
| DP28 (E14+, F14-) | TOKEN\_OUT | LVPECL(DP) | 🡪 SWB |
| DP29 (A15+,B15-) | SD\_Link | LVDS/MGT | 🡨 SWB |
| DP30 (D15+,E15-) | TrigOut | LVPECL/MGT | 🡪 SWB |
| SE7 (G13) | Busy\_Out | LVTTL | 🡪 SWB |
| SE8 (G15) | Stat\_IN | LVTTL | 🡨 SWB |

7.2 3M P50E-68E1-RR1 68-pin connector Pinout Table (FP1)

|  |  |  |  |
| --- | --- | --- | --- |
| Any level differential inputs | | | |
| Pin name | Signal Description | Pin name | Signal Description |
| 1 | A1\_P | 2 | A17\_P |
| 3 | A1\_N | 4 | A17\_N |
| 5 | A2\_P | 6 | A18\_P |
| 7 | A2\_N | 8 | A18\_N |
| 9 | A3\_P | 10 | A19\_P |
| 11 | A3\_N | 12 | A19\_N |
| 13 | A4\_P | 14 | A20\_P |
| 15 | A4\_N | 16 | A20\_N |
| 17 | A5\_P | 18 | A21\_P |
| 19 | A5\_N | 20 | A21\_N |
| 21 | A6\_P | 22 | A22\_P |
| 23 | A6\_N | 24 | A22\_N |
| 25 | A7\_P | 26 | A23\_P |
| 27 | A7\_N | 28 | A23\_N |
| 29 | A8\_P | 30 | A24\_P |
| 31 | A8\_N | 32 | A24\_N |
| 33 | A9\_P | 34 | A25\_P |
| 35 | A9\_N | 36 | A25\_N |
| 37 | A10\_P | 38 | A26\_P |
| 39 | A10\_N | 40 | A26\_N |
| 41 | A11\_P | 42 | A27\_P |
| 43 | A11\_N | 44 | A27\_N |
| 45 | A12\_P | 46 | A28\_P |
| 47 | A12\_N | 48 | A28\_N |
| 49 | A13\_P | 50 | A29\_P |
| 51 | A13\_N | 52 | A29\_N |
| 53 | A14\_P | 54 | A30\_P |
| 55 | A14\_N | 56 | A30\_N |
| 57 | A15\_P | 58 | A31\_P |
| 59 | A15\_N | 60 | A31\_N |
| 61 | A16\_P | 62 | A32\_P |
| 63 | A16\_N | 64 | A32\_N |
| 65 | AGND | 66 | AGND |
| 67 | AGND | 68 | AGND |

\*The differential receiver (LVDS 🡪 LVTTL) inverts the polarity of these inputs

7.3 3M P50E-68E1-RR1 68-pin connector Pinout Table (FP2)

|  |  |  |  |
| --- | --- | --- | --- |
| Any level differential inputs | | | |
| Pin name | Signal Description | Pin name | Signal Description |
| 1 | C1\_P | 2 | C17\_P |
| 3 | C1\_N | 4 | C17\_N |
| 5 | C2\_P | 6 | C18\_P |
| 7 | C2\_N | 8 | C18\_N |
| 9 | C3\_P | 10 | C19\_P |
| 11 | C3\_N | 12 | C19\_N |
| 13 | C4\_P | 14 | C20\_P |
| 15 | C4\_N | 16 | C20\_N |
| 17 | C5\_P | 18 | C21\_P |
| 19 | C5\_N | 20 | C21\_N |
| 21 | C6\_P | 22 | C22\_P |
| 23 | C6\_N | 24 | C22\_N |
| 25 | C7\_P | 26 | C23\_P |
| 27 | C7\_N | 28 | C23\_N |
| 29 | C8\_P | 30 | C24\_P |
| 31 | C8\_N | 32 | C24\_N |
| 33 | C9\_P | 34 | C25\_P |
| 35 | C9\_N | 36 | C25\_N |
| 37 | C10\_P | 38 | C26\_P |
| 39 | C10\_N | 40 | C26\_N |
| 41 | C11\_P | 42 | C27\_P |
| 43 | C11\_N | 44 | C27\_N |
| 45 | C12\_P | 46 | C28\_P |
| 47 | C12\_N | 48 | C28\_N |
| 49 | C13\_P | 50 | C29\_P |
| 51 | C13\_N | 52 | C29\_N |
| 53 | C14\_P | 54 | C30\_P |
| 55 | C14\_N | 56 | C30\_N |
| 57 | C15\_P | 58 | C31\_P |
| 59 | C15\_N | 60 | C31\_N |
| 61 | C16\_P | 62 | C32\_P |
| 63 | C16\_N | 64 | C32\_N |
| 65 | CGND | 66 | CGND |
| 67 | CGND | 68 | CGND |

\*The differential receiver (LVDS 🡪 LVTTL) inverts the polarity of these inputs

7.4 FCI61082-101400LF 100-pin connector Pinout Table (FP6)

|  |  |  |  |
| --- | --- | --- | --- |
| LVTTL (3.3V), or LVCMOS (3.3V) | | | |
| Pin name | Signal Description | Pin name | Signal Description |
| 1, 3, 5 | +5V | 2, 4, 6 | +5V |
| 7 | B\_SEL | 8 | B\_OE |
| 9 | NC | 10 | NC |
| 11 | GND | 12 | B1 |
| 13 | GND | 14 | B17 |
| 15 | GND | 16 | B2 |
| 17 | GND | 18 | B18 |
| 19 | GND | 20 | B3 |
| 21 | GND | 22 | B19 |
| 23 | GND | 24 | B4 |
| 25 | GND | 26 | B20 |
| 27 | GND | 28 | B5 |
| 29 | GND | 30 | B21 |
| 31 | GND | 32 | B6 |
| 33 | GND | 34 | B22 |
| 35 | GND | 36 | B7 |
| 37 | GND | 38 | B23 |
| 39 | GND | 40 | B8 |
| 41 | GND | 42 | B24 |
| 43, 45, 47, 49, 51, 53, 55, 57 | +3.3V | 44, 46, 48, 50, 52, 54, 56, 58 | NC |
| 59 | GND | 60 | B9 |
| 61 | GND | 62 | B25 |
| 63 | GND | 64 | B10 |
| 65 | GND | 66 | B26 |
| 67 | GND | 68 | B11 |
| 69 | GND | 70 | B27 |
| 71 | GND | 72 | B12 |
| 73 | GND | 74 | B28 |
| 75 | GND | 76 | B13 |
| 77 | GND | 78 | B29 |
| 79 | GND | 80 | B14 |
| 81 | GND | 82 | B30 |
| 83 | GND | 84 | B15 |
| 85 | GND | 86 | B31 |
| 87 | GND | 88 | B16 |
| 89 | GND | 90 | B32 |
| 91 | B\_ID0 | 92 | B\_ID1 |
| 93 | B\_ID2 | 94 | NC |
| 95, 97, 99 | -5.0V | 96, 98, 100 | -5.0V |

7.5 FCI61082-101400LF 100-pin connector Pinout Table (FP7)

|  |  |  |  |
| --- | --- | --- | --- |
| LVTTL (3.3V), or LVCMOS (3.3V) | | | |
| Pin name | Signal Description | Pin name | Signal Description |
| 1, 3, 5 | +5V | 2, 4, 6 | +5V |
| 7 | D\_SEL | 8 | D\_OE |
| 9 | NC | 10 | NC |
| 11 | GND | 12 | D1 |
| 13 | GND | 14 | D17 |
| 15 | GND | 16 | D2 |
| 17 | GND | 18 | D18 |
| 19 | GND | 20 | D3 |
| 21 | GND | 22 | D19 |
| 23 | GND | 24 | D4 |
| 25 | GND | 26 | D20 |
| 27 | GND | 28 | D5 |
| 29 | GND | 30 | D21 |
| 31 | GND | 32 | D6 |
| 33 | GND | 34 | D22 |
| 35 | GND | 36 | D7 |
| 37 | GND | 38 | D23 |
| 39 | GND | 40 | D8 |
| 41 | GND | 42 | D24 |
| 43, 45, 47, 49, 51, 53, 55, 57 | +3.3V | 44, 46, 48, 50, 52, 54, 56, 58 | NC |
| 59 | GND | 60 | D9 |
| 61 | GND | 62 | D25 |
| 63 | GND | 64 | D10 |
| 65 | GND | 66 | D26 |
| 67 | GND | 68 | D11 |
| 69 | GND | 70 | D27 |
| 71 | GND | 72 | D12 |
| 73 | GND | 74 | D28 |
| 75 | GND | 76 | D13 |
| 77 | GND | 78 | D29 |
| 79 | GND | 80 | D14 |
| 81 | GND | 82 | D30 |
| 83 | GND | 84 | D15 |
| 85 | GND | 86 | D31 |
| 87 | GND | 88 | D16 |
| 89 | GND | 90 | D32 |
| 91 | D\_ID0 | 92 | D\_ID1 |
| 93 | D\_ID2 | 94 | NC |
| 95, 97, 99 | -5.0V | 96, 98, 100 | -5.0V |

7.6 VME P2 connector Pinout Table (FP6)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| LVTTL (3.3V), or LVCMOS (3.3V) | | | | | |
| row | Z Signal description | A Signal description | B Signal description | C Signal description | D Signal description |
| 1 | VZ1 | RxA\_P | +5VME | RxC\_P | VD1 |
| 2 | GND | RxA\_N | GND | RxC\_N | VD2 |
| 3 | VZ2 | TxA\_P | Retry\_N | TxC\_P | VD3 |
| 4 | GND | TxA\_N | AV24 | TxC\_N | VD4 |
| 5 | VZ3 | VA1 | AV25 | VC1 | GND |
| 6 | GND | VA2 | AV26 | VC2 | +3.3V |
| 7 | VZ4 | VA3 | AV27 | VC3 | +3.3V |
| 8 | GND | VA4 | AV28 | VC4 | +3.3V |
| 9 | VZ5 | VA5 | AV29 | VC5 | +3.3V |
| 10 | GND | VA6 | AV30 | VC6 | +3.3V |
| 11 | VZ6 | VA7 | AV31 | VC7 | GND |
| 12 | GND | VA8 | GND | VC8 | GND |
| 13 | VZ7 | VA9 | +5VME | VC9 | VD5 |
| 14 | GND | VA10 | DV16 | VC10 | VD6 |
| 15 | VZ8 | VA11 | DV17 | VC11 | VD7 |
| 16 | GND | VA12 | DV18 | VC12 | VD8 |
| 17 | VZ9 | VA13 | DV19 | VC13 | GND |
| 18 | GND | VA14 | DV20 | VC14 | GND |
| 19 | VZ10 | VA15 | DV21 | VC15 | GND |
| 20 | GND | VA16 | DV22 | VC16 | GND |
| 21 | VZ11 | VA17 | DV23 | VC17 | -5V |
| 22 | GND | VA18 | GND | VC18 | -5V |
| 23 | VZ12 | VA19 | DV24 | VC19 | -5V |
| 24 | GND | VA20 | DV25 | VC20 | -5V |
| 25 | VZ13 | VA21 | DV26 | VC21 | -5V |
| 26 | GND | VA22 | DV27 | VC22 | GND |
| 27 | VZ14 | VA23 | DV28 | VC23 | GND |
| 28 | GND | VA24 | DV29 | VC24 | GND |
| 29 | VZ15 | VA25 | DV30 | VC25 | GND |
| 30 | GND | VA26 | DV31 | VC26 | GND |
| 31 | VZ16 | VA27 | GND | VC27 | GND |
| 32 | GND | VA28 | +5VME | VC28 | NC |

\*VZ#, VA#, VC# and VD#: Connected to FPGA for P2 IO card; DV# and AV#: standard VME64 pins. VA1-28, VD1-4, VC1-28 and VD5-8 are used as 64 P2 input channels

7.7 3M N3408-D302 connector Pinout Table (FPT1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Pin name | Signal Description | Pin name | Signal Description |  |
| 1 | OUT1\_P (ECL) | 2 | OUT1\_N (ECL) | OR connector#A |
| 3 | OUT2\_P (ECL) | 4 | OUT2\_N (ECL) | OR mezz#B |
| 5 | OUT3\_P (ECL) | 6 | OUT3\_N (ECL) | OR connector#C |
| 7 | OUT4\_P (ECL) | 8 | OUT4\_N (ECL) | OR mezz#D |
| 9 | OUT5\_P (ECL) | 10 | OUT5\_N (ECL) | OR vme row#A |
| 11 | OUT6\_P (ECL) | 12 | OUT6\_N (ECL) | OR Vme row#C |
| 13 | OUT7\_P (ECL) | 14 | OUT7\_N (ECL) | OR width-adj. |
| 15 | OUT8\_P (ECL) | 16 | OUT8\_N (ECL) | OR IN#5 |
| 17 | Ref#1 In\_P (DS) | 18 | Ref#1 In\_N (DS) |  |
| 19 | Ref#2 In\_P (DS) | 20 | Ref#2 In\_N (DS) |  |
| 21 | Ref#3 In\_P (DS) | 22 | Ref#3 In\_N (DS) |  |
| 23 | Ref#4 In\_P (DS) | 24 | Ref#4 In\_N (DS) |  |
| 25 | IN5\_N (DS) | 26 | IN5\_P (DS) | extra trigger in |
| 27 | Calib\_P (DS) | 28 | Calib In\_N (DS) | Calibration In |
| 29 | Trigger\_P (DS) | 30 | Trigger\_N (DS) | Readout trigger |
| 31 | Reset\_P (DS) | 32 | Reset\_N (DS) | Sync\_reset |

* Pin#1/2: lower right corner, Pin#17/18: lower left corner
* Ref#1, 2, 3, and 4 are polarity inverted on the PCB by the SN65LVDT352 receiver.
* Inputs pin#25-32 are also polarity inverted on the PCB, and the polarity is reversed back inside the FPGA.
* Trigger OUT(1:6) widths are 4 ns wide; OUT(7) width is adjustable; OUT(8) width is determined by the OUT(7) and IN#5.

##### Appendix D: Document revision history:

Aug. 1, 2014: Initial document;

Aug. 4, 2014: Added the data format and data readout

Oct. 3, 2014: Updated the VME emergency loading, which include FPGA and Flash memory loading.

Oct. 28, 2015: Updated Table1 for data format.

Nov. 11, 2015: Added registers 0x30, 0x50, 0x54, 0x58 and 0x5C for extra scalars and status readout.

Nov. 16, 2015: Added the comments under the table for vf2TDC data format.

June 22, 2017: Modified the connector pinouts for the production VETROC boards.

Oct. 12, 2017: updated the 0x2C (clock selection), and 0x5C (firmware, status) register, and section 5.3.

Oct. 20, 2017: updated the 0x60-0x78, and clarified the pulse polarity and edge definition. Added the simple trigger logic outputs (The very top connector on the front). TDC edge readout disable (to reduce the size).

Jan. 31, 2025: Added the Trigger timing section, 2.6, and the data format sent to VTP.

May 6, 2025: Added the buffer (FIFO) structure in section 3.3.

May 7, 2025: Added the scalers readout (valid from firmware V9.2)

##### Appendix E: References:

1. Gu etc. VETROC manual
2. Gu etc. vfTDC design
3. Xilinx FPGA http://www.xilinx.com/