Physics Division

Description and Technical Information for the Stream readout Time to Digital Converter board (STDC)

Updated on: Feb. 22, 2018

Table of Contents

1 Introduction

The STDC (Stream readout Time to Digital Converter board) is being developed based on the VETROC base 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 Board-mount 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.

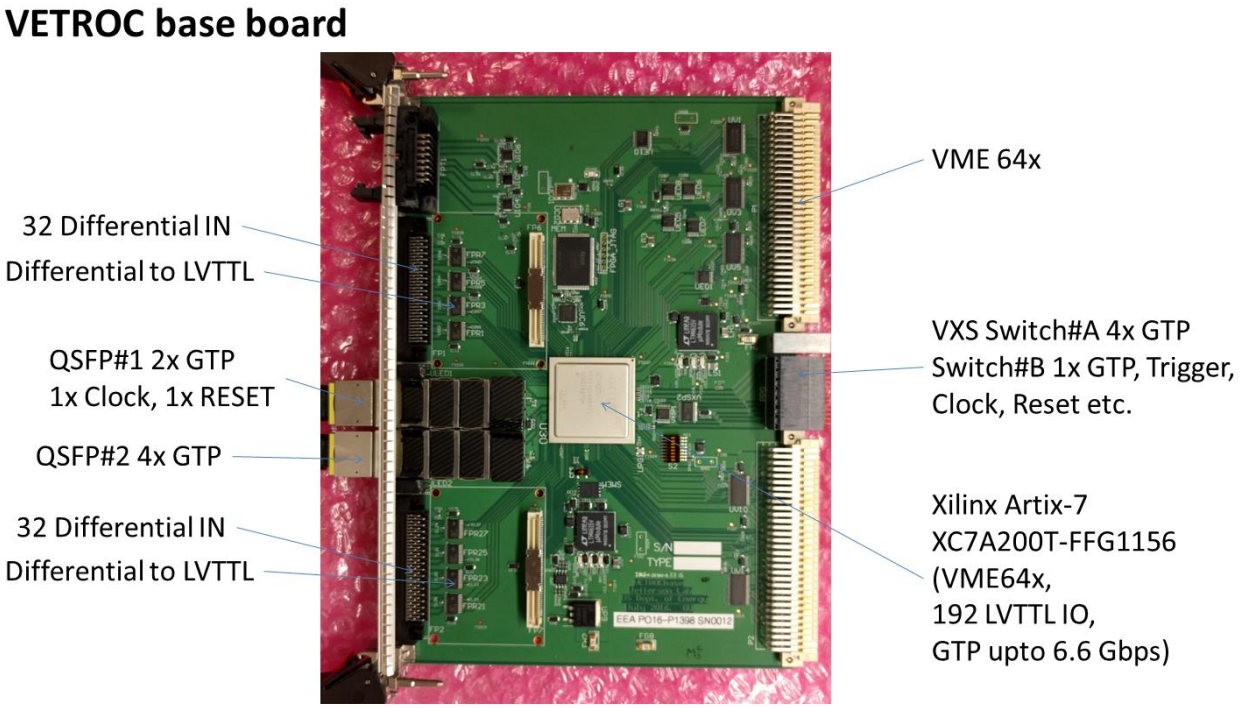


Figure 1 STDC board and its major components

The STDC is a 32 channel TDC, which uses the bottom front panel connector as inputs, and lower QSFP (#2) as streaming data output.

2 PCB design of the STDC module

STDC is designed as a VXS payload board. It also works in a standard 6U VME64 crate. It receives 32 any-level differential signals. Figure 2 shows a diagram of the VETROC board. The STDC design uses a subset of the VETROC connections. That is: one 3M P50E-68P1-RR1 connector for TDC channel input, one QSFP (4* 6Gbps) for streaming data readout, on-board 250 MHz oscillator for time measurement, and several VME registers for board control.

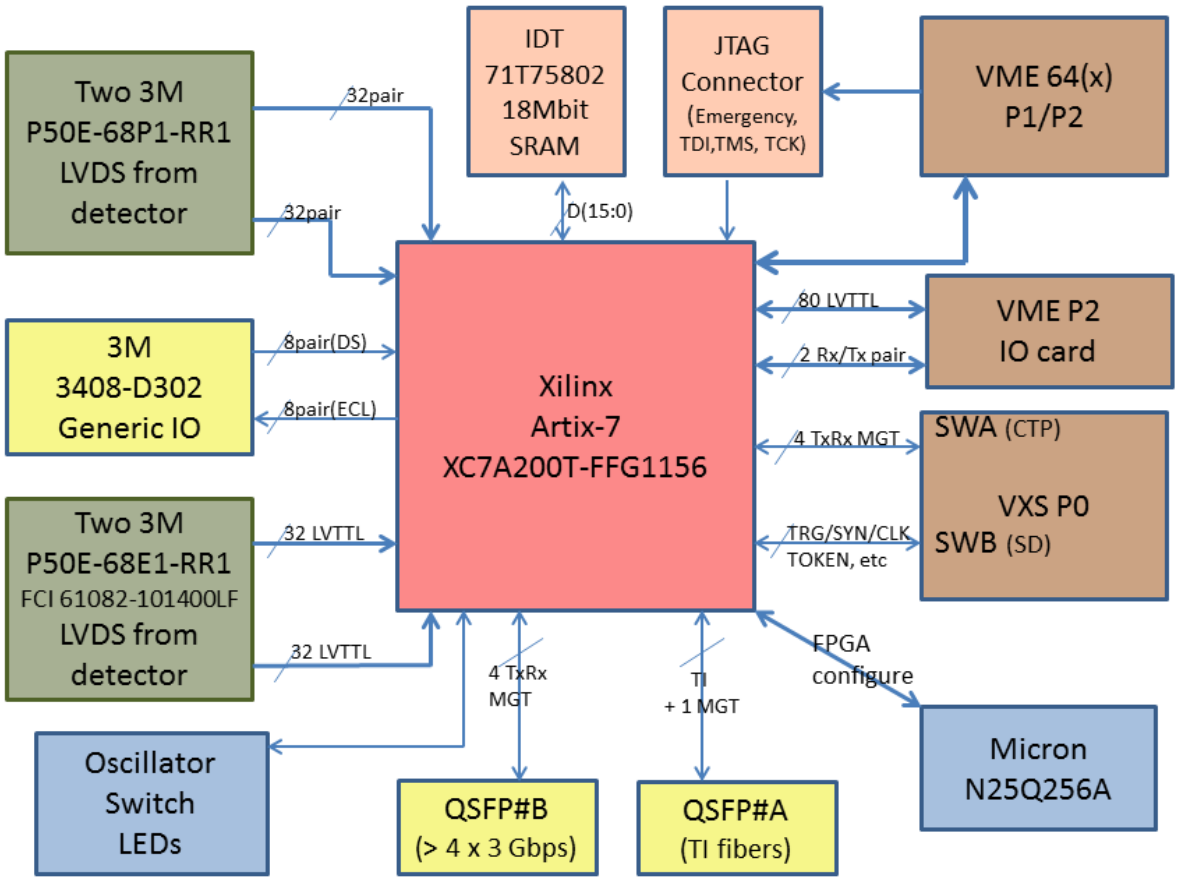


Figure 2 vfTDC fuctional diagram

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.

Two QSFP connectors are also loaded on the front panel. One QSFP connector has 4 MGTs (Multi-Gigabit Transceivers) connected to the FPGA directly, which is capable of more than 3 Gbps each. This QSFP is used for streaming data readout. The other QSFP is compatible with Trigger Interface (TI). It could be used for clock and control input (instead of the on-board oscillator and VME)

The board is compatible with VME64x backplane. It can also work without any backplane by supplying a signal +5V through the VME P1 connector.

2.1 FPGA programming

The FPGA XC7A200T needs about 80Mbit to configure. This configure data is saved in the Micron N25Q256, which is 256 Mbit. There are two N25Q256 chips to save two versions of the FPGA configure file. The FPGA configure is chosen by a three-position sliding switch. This can easily change the board functionality by loading another FPGA firmware.

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 STDC can also 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. 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 bit of A24 address should match with the geographic address, and the lower A24 address is set to be 0x0FFFC.

2.2: Clock Distribution

There are four main clock sources for the 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, two on-board oscillators are implemented, one is 250 MHz, and another one is I2C programmable with default frequency of 125 MHz. The forth source is the optional front panel TI fiber input. Only one clock source is selected as the FPGA clock, which pipelines the time conversion and streaming data readout. 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 third one for the south MGT blocks; and one clock to the front panel QSFP when the TI interface is used. The sources for these four outputs (from SY58040) can be selected independently.

2.3 VME interface

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

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 JLAB 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: Streaming Readout logic

The readout is automatic. When the signal edge (rising edge and/or falling edge) is detected, the digitized edge time (TDC) will be buffered in the memory, and read out through the QSFP with the peak rate of 20 Gbps (four channels with 6.25 Gbps 8b/10b encoded each).

3 TDC and FPGA design

The functional diagram and data flow of the STDC is shown in figure 3. The detailed functions are shown in the figures later in this section.

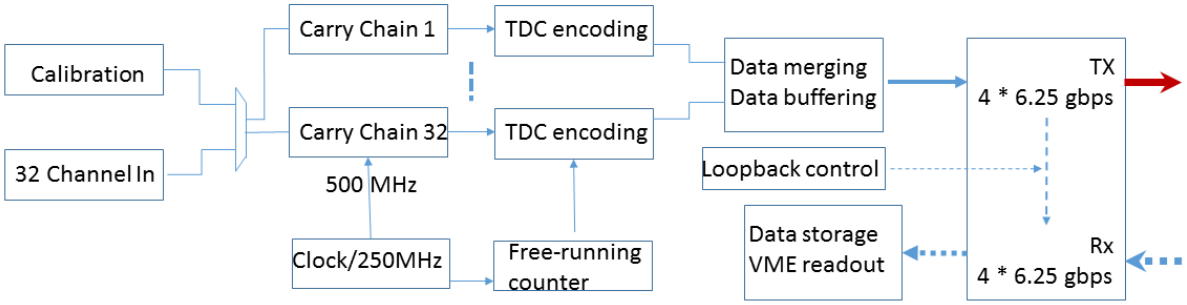


Figure 3 STDC functional diagram and streaming data flow

3.1 FPGA based TDC design

The TDC measurement in FPGA is pretty straight forward. 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 (or the fine time within the clock cycle). After combining the coarse measurement and the fine measurement, the TDC can reach a precision of ~35 ps and a range of many seconds.

The TDC fine measurement is based on the FPGA carry chain delays [3]. The delay unit (carry chain) is about 17 ps for the Artix-7 FPGA. Figure 3 shows the design of one TDC channel.

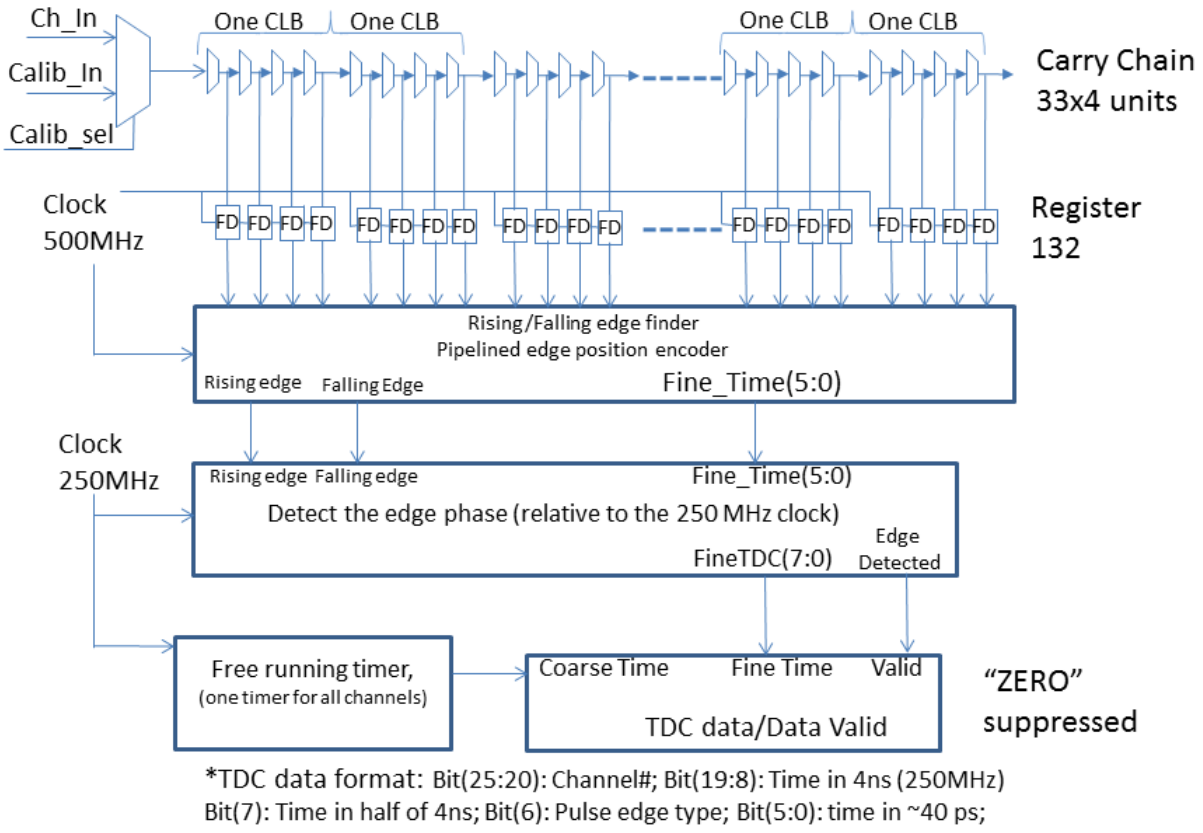


Figure 4 Diagram of one TDC channel

Each channel uses 33 FPGA slices (132 delay elements) for fine delay. To simplify the encoding logic, there should be no more than one edge in the delay elements, which is about 2.3 ns. Each channel will generate data at 250 MHz with zero suppression (Data Valid). The single channel data (26 bits) has a range of about 16us.

Every four channels are combined to form a 128 bit data. In addition to the four channel TDC data (104 bits), 20 more common timer bits are added, which extend the TDC measurement range to more than two seconds (with three timer bits overlapping with the individual channel). The four channels data are buffered in the FPGA internal memory.

The eight buffers (32 channels with 4 channels per buffer) are further merged into a streaming output buffer. The output buffer feeds data to the MGT (Multi Gigabit Transceivers), where the data streaming is four channels of 6.25 Gbps each (8b/10b encoded). Figure 4 shows the diagram of the STDC streaming data readout.

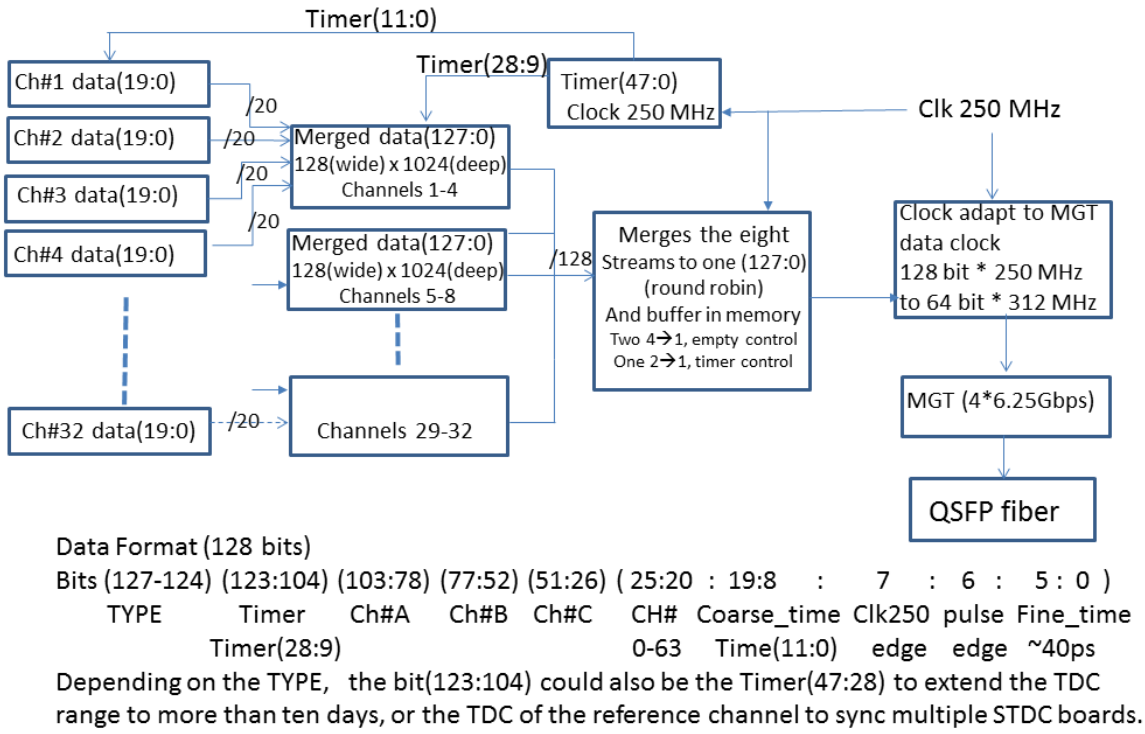


Figure 5 STDC data flow: data merging and data readout

The data are merged in two stages. The first stage merges four (4) TDC channels, and stores the data in a 1 K deep buffer at full speed (250 MHz as data generation). The second stage merges eight buffers from the first stage at 250 MHz. The merged data are streamed to the four channels of the QSFP connectors.

3.2 STDC data format

The STDC stream data is read out as FIFO (First In First Out), but not exactly in the order of the TDC measurement because of the data merging logic.

The data are formatted (aligned) in 128-bit boundaries. Table 1 show the data format. Table 2 shows the data format for one channel. By combining the data in bits(123:104) and the coarse time and fine time in each channel, the TDC can be in the range of ~10 days with the LSB of about 35ps.

Table 1 STDC data format

Data	Bit(127:124)	Bit(123:104)	Bit(103:78)	Bit(77:52)	Bit(51:26)	Bit(25:0)
format	DataType	Timer/RefData*	Chan#A TDC	Chan#B TDC	Chan#C TDC	Chan#D TDC

*Depending on the "DataType", this could be the timer(47:28), timer(28:9), or the reference channel TDC data.

Table 2 Channel TDC data format

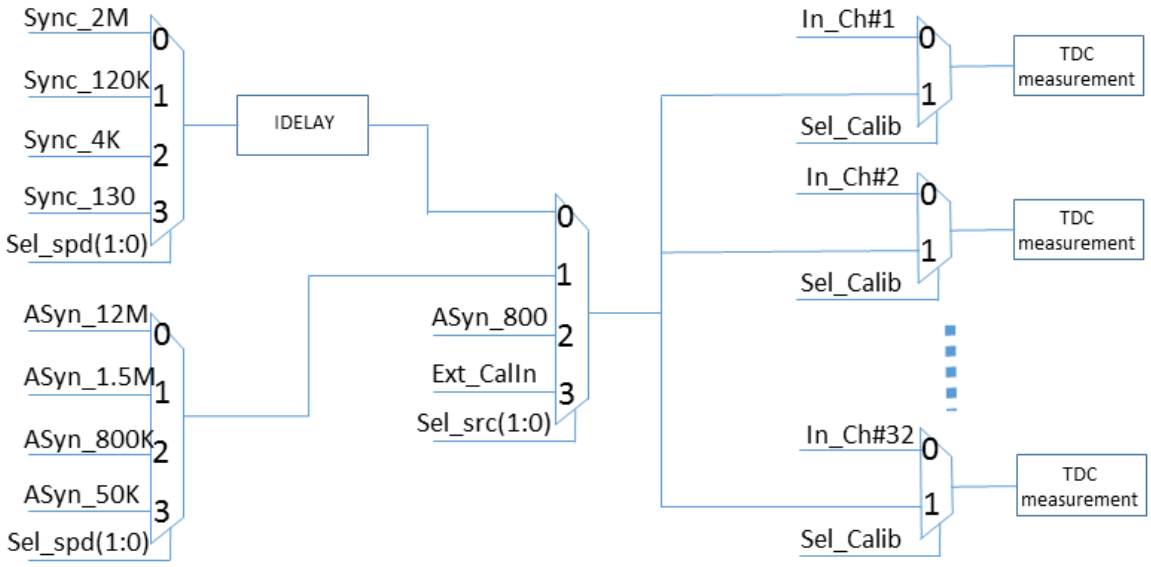
Chan data	Bit(25:20)	Bit(19:8)	Bit(7)	Bit(6)	Bit(5:0)
format	Chan# 0-31	Coarse Time 16 us in 4 ns steps	2ns phase	Edge type 0: rising, 1: falling	Fine time ~2.3 ns in ~35 ps steps

3.3 STDC calibration

The STDC needs to be calibrated properly to work correctly. A good quality clock source is required as the measurement depends on the clock (coarse time is the number of clock cycles, fine time is the phase of the clock).

3.3.1 Calibration hardware/firmware implementation

There are three calibration sources with adjustable rates. Figure 5 shows the calibration diagram. All the 32 channels have the same calibration signal, and the calibration source can be IDELAY adjusted synchronous pulse (Sync_#, having a fixed phase as the TDC clock), asynchronous pulse (Asyn_#, having no phase relationship with the TDC clock), or an external calibration input (Ext_CalIn). The synchronous calibration uses the IDELAY to tune the signal, which can add an incremental delay of about 78ps with a total range of 2.5ns. The external calibration input can be especially useful when multiple STDC boards are being used.



*Sel_spd(1:0): Calibration pulse speed (frequency) selection by VME register offset 0x20, bit (5:4);

*Sel_src(1:0): Calibration source selection by VME register offset 0x20, bit(7:6);

*Sel_Calib: Calibration selection (vs connector inputs for TDC measurements) VME register offset 0x20, bit(0).

Figure 6 The calibration signal connection and calibration signal speed selection.

By using these calibration, the linearity, and TDC functionality can be checked.

3.3.2 The LSB real time calibration

The TDC readout LSB (Least Significant Bit) is calibrated real-time. Because of the total carry chain (shown in figure 3) is longer than the clock period, some pulse edges will be registered in two clock cycles (one near the beginning, the other near the end). These measurements give a real-time total delay value. It can be used as real-time temperature and supply power corrections.

The measurement LSB is achieved by the double measurements of the same edge. This supplies a natural range calibration in real time. The inverse of the range is the measurement LSB. On average, 10% of the inputs will be measured twice as mentioned in calibration section. This range calibration can correct the carry chain temperature effect, and voltage drifting etc.

3.3.3 More calibrations offline

Offline data analysis is required to calibrate the input offset. This offset is the channel delay on the STDC PCB (Printed Circuit Board), and inside the FPGA (from FPGA pad to the first cell of carry chain delay). The firmware tries to location-lock the carry chain inside the FPGA to minimize the shifts as the firmware updates.

This offset may also depend on the environment, such as the temperature and power supplies. The effect from power supply variation should be minor as the powers used are from the on-board regulators (isolated from the external +5V supply).

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. It can also work in a standalone mode requiring +5 only.

4.2 High speed serial P0 inputs and outputs (not used for STDC design):

- 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 (one set of 32 inputs is used for STDC);
- 8 generic differential signal inputs (one input is used for external calibration input);
- 8 generic ECL outputs (not used).
- 4 channels of MGT on the bottom QSFP for streaming data output.
- TI or TImaster fiber IO on the top QSFP (not used for STDC).
- 2 optional mezzanine board with up to 32 channels (LVTTL) each (not used for STDC).

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 acknowledgement;
  - Bit 3: Pulse edge detected (Readout trigger as vf^{2}TDC);
  - Bit 4: MGT Rx error or board in reset;
- Set #2:
  - Bit 1: Streaming readout enabled (ON), MGT Tx FIFO is full when flashing;
  - Bit 2: Data activity on streaming output (TX);
  - Bit 3: Streaming receiver enabled (ON), MGT Rx FIFO is full when flashing;
  - Bit 4: Data activity in streaming input (RX).

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 visions 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
- Standalone mode (or non-VME64x): +5V @ 3A.
- Local regulators for other required voltages: +1.0V, +1.2V, +1.8V, +2.5V, and -5V (for certain types of mezzanine card only).

4.7 Environment:

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

5 STDC operation procedures:

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

5.1 STDC Power supply:

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

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

(1). The fuse, FG1 is stuffed;
(2). The 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). The fuse, FG1 is removed;
(2). The UP2 is stuffed.

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

5.2 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.

5.3 STDC 1-bit three-position slide switch CES setting:

When the switch is in the left position (closer to the front panel), the vf^{2}TDC function (firmware) is chosen and the board functions as a 192 channel trigger (event) based TDC.

When the switch is in the middle position, the FPGA will not be loaded as both flash memories are disconnected.

When the switch is in the right position (closer to the VME backplane connectors), the STDC function (firmware) is chosen. The board functions as a 32-channel streaming readout TDC.

5.3 STDC 8-bit switch S2 setting:

When the board functions as a vf^{2}TDC 192-chanel TDC, Bit[8:4]: set the VME A24 address space A[23:19] when the vf^{2}TDC 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.

The 8-bit switch HIGH/LOW setting:

Bit(8:3) (top six switches): Left: HIGH (1), right: LOW (0), LVTTL compatible;

Bit(2:1) (bottom two switches, closer to S2 marking): left : LOW (0), right: HIGH (1) on the PCB. These two bits are polarity reversed inside the FPGA, so it will be the same as Bit(8:3): left HIGH(1).

When the board functions as a STDC board, the switch is used as board slow control, to set the VME register assuming that the board is in standalone mode (power supply only). The table shows the function of the switch. When using this switch, the Switch bit(7:1) are set first, then flip switch bit#8 to set the register, or take action. (Setting high and ACTION for RESET, setting low will not do anything).

Action Bit(8)	Address Bit(7:5)	Register Bit(4:1)
0→1 or 1→0	000	Resets: FIFO Reset, Clk250 DCM reset, RxSoftReset, TxSoftReset if '1';
	001	Resets: RxReset, RxLPMreset, RxReset, MGTReset if '1';
	010	ClkOut#2 input selection(1:0), ClkOut#1 input selection(1:0)
	011	ClkOut#4 input selection(1:0), ClkOut#3 input selection(1:0)
	100	General Reset (toggle "0→1" or "1→0"), Falling Edge Enable, Rising Edge enable, Calibration select if '1'
	101	Calibration source selection(2:1), Calibration speed selection(2:1)
	110	Rx_enable (basically enable the Rx FIFO write) if '1', no use, no use, no use
	111	Tx_Enable if '1', MGT loopback setting(2:0)

5.4 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. This is the default setting.

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

The STDC 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]=b'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]. Most of the following registers are for vf^{2}TDC only. The registers relevant to STDC are marked bold.

- 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: vf^{2}TDC;

- Address offset: 0x00004: vf^{2}TDC readout window setting:

  Bit 7-0 (R/W): width of the TDC readout window (in 4 ns steps)

- Address offset: 0x00008: Interrupt setting:

  Bit 7-0 (R/W): Interrupt ID; Reset default 0xC8
  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.

- 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.
  For STDC, the register is defined as:
    Bit 0: Calibration Select; SwitchAdd(100), SwData(1)
    Bit 1: TDC rising edge enable; SwitchAdd(100), SwData(2);
    Bit 2: Falling Edge enabler; SwitchAdd(100), SwData(3);
    Bit 3: Generic reset, SwitchAdd(100), SwData(4);
    Bit 5:4: Calibration speed: SwitchAdd(101), SwData(2:1)
      00: CountS(1), CountAS(1);
      01: CountS(5), CountAS(4);
      10: CountS(10), CountAS(5);
      11: CountS(15), CountAS(9);
    Bit 7:6: Calibration source selection: SwitchAdd(101), SwData(4:3)
      00: Cal_Sync / CountS(#) as chosen by Bit4:3;

01: Cal_ASync / CountAS(#) as chosen by Bit4:3;
10 : Slow_CalASync, == CountAS(15);
11: External Calibration selection

Bit 10:8: **VME register overwrite (higher priority)**

Bit 11: Rx_Enable; SwitchAdd(110), SwData(4)

Bit 14:12: Loopback setting, directly connects to MGT loopback(2:0); SwitchAdd(111), SwData(3:1)

Bit 15: Tx_Enable; SwitchAdd(111), SwData(4)

- 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). Reset default 00; SwitchAdd(010), SwData(2:1);
Bit[1:0] = 00: QSFP TI front panel clock;
Bit[1:0] = 01: no use;
Bit[1:0] = 10: on-board oscillator;
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). Reset default 00; SwitchAdd(010), SwData(4:3);

Bit 5-4 (R/W): software bit switch to control the clock source for ClkOut(2). Reset default 00; SwitchAdd(011), SwData(2:1);

Bit 7-6 (R/W): software bit switch to control the clock source for ClkOut(3). Reset default 00; SwitchAdd(011), SwData(4:3);

- Address offset: 0x00040 (R/W): Trigger logic channel mask (off);

Bit 31-0: Input Channel#(32:1) masks for trigger(edge detected) logic, if '1', the channel will be disabled (masked off);

- 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: 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;

- 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 5: if '1', Tx Reset, softreset; SwitchAdd(000), SwData(1);
  Bit 6: if '1', Rx Reset, softreset; SwitchAdd(000), SwData(2);

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 8: if '1', Clk250 DCM reset; SwitchAdd(000), SwData(3);
Bit 9: if '1', Fifo Reset; SwitchAdd(000), SwData(4);
Bit 10: if '1', Reset the MGT (MultiGigabit Transceiver,) inside the FPGA.
Bit 10: if '1', MGT Reset = EyeScanReset; Switch(001), SwData(1);
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 12: if '1', RxReset; SwitchAdd(001), SwData(2);
Bit 13: if '1', RxLPM reset; SwitchAdd(001), SwData(3);
Bit 14: if '1', RxReset; SwitchAdd(001), SwData(4);
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.

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

7.3 3M P50E-68E1-RR1 68-pin connector Pinout Table (FP2), USED for STDC

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

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)

3	EdgeTrig_P (ECL)	4	EdgeTrig_N (ECL)
5	OUT3_P (ECL)	6	OUT3_N (ECL)
7	OUT4_P (ECL)	8	OUT4_N (ECL)
9	OUT5_P (ECL)	10	OUT5_N (ECL)
11	OUT6_P (ECL)	12	OUT6_N (ECL)
13	OUT7_P (ECL)	14	OUT7_N (ECL)
15	OUT8_P (ECL)	16	OUT8_N (ECL)
17	TriggerIn_N (DS)	18	TriggerIn_P (DS)
19	ResetIn_N (DS)	20	ResetIn_P (DS)
21	Ext_Calib_N (DS)	22	Ext_Calib_P (DS)
23	IN4_N (DS)	24	IN4_P (DS)
25	IN5_N (DS)	26	IN5_P (DS)
27	IN6_N (DS)	28	IN6_P (DS)
29	IN7_N (DS)	30	IN7_P (DS)
31	IN8_N (DS)	32	IN8_P (DS)

Appendix A: STDC board setup example using VME:

Step 1, VME write to offset 0x20, data 0x500, to set the VME as priority (over SWITCH);

Step 2, VME write to offset 0x2C, data 0x55, to set the clock sources (on-board oscillator);

Step3, VME write to offset 0x100, data 0x100, to reset the DCM;

Step 4, VME write to offset 0x20, data 0x579, for board reset and internal calibration setup, but no edge is enabled;

Step 5, VME write to offset 0x20, data 0x571, for board reset, and set the reset bit (#3) to low;

Step 6, VME write to offset 0x100, data 0x20, 0x40 with ~1ms pauses in between, to reset the MGT transceivers;

Step 7, VME write to offset 0x100, data 0x200, to reset (clear) the FIFOs in STDC;

Extra register write examples:

VME write to offset 0x20, data 0x8573, to enable the fiber transmitter, and pulse rising edge TDC;

VME write to offset 0x20, data 0xD71, to enable the fiber receiver;

Appendix B: STDC board setup using switch (standalone mode):

Step 0, set VME register offset 0x20, data bits(10:8) to “000”, if they were set to “101”.

Step 1, Switch(7:1) set to “0100101”, then toggle Switch(8),

Step 2, Switch(7:1) set to “0110101”, then toggle Switch(8), to setup the FPGA/board clock;

Step 3, Switch(7:1) set to “0000100”, then toggle Switch(8), to reset FPGA DCM;

Step 4, Switch(7:1) set to “1001000”, then toggle Switch(8), to reset FPGA;

Step 5, Switch(7:1) set to “1000000”, then toggle Switch(8), to reset the reset bit;

Step 6, Switch(7:1) set to “0000001”, then toggle Switch(8),

Step 7, Switch(7:1) set to “0000010”, then toggle Switch(8), to reset the MGT transceivers;

Step 8, Switch(7:1) set to “0001000”, then toggle Switch(8), to reset the FPGA FIFOs;

Extra register write exasmples:

Switch(7:1) set to “1101000”, then toggle Switch(8), to set the STDC as a stream data receiver;

Switch(7:1) set to “1010111”, then toggle Switch(8), to set the internal calibration pulse rate;

Switch(7:1) set to “1000011”, then toggle Switch(8), to set to calibration, and enable rising edge TDC;

Switch(7:1) set to “1111000”, then toggle Switch(8), to enable the stream data transmission;

**Switch(8) toggle: either flip the switch ‘from OFF to ON’, or ‘from ON to OFF’.*

Appendix C: Document revision history:

Aug. 1, 2014: Initial document for vf^{2}TDC;

Feb. 26, 2018: Modified for STDC design;

Mar. 7, 2018: Added the example board setup sequence in Appendix A and Appendix B;

Appendix D: References:

1. Gu etc., vf^{2}TDC design: https://coda.jlab.org/drupal/content/download
2. Xilinx FPGA http://www.xilinx.com/