EUDET-Memo-2009-4

Description of the JRA1 Trigger Logic Unit (TLU), v0.2c

D. Cussans*

September 11, 2009

Abstract

This document is an updated version of EUDET-Memo-2008-50. It describes the interfaces and operation of the EUDET JRA1 Trigger Logic Unit (TLU v0.2c) with firmware version 253. The TLU is intended for test-beam use and provides an interface between the beam-trigger, the DAQ and the device-under-test.[1]

*University of Bristol, UK

[1]Any good ideas are due to the collaborative efforts of members of the JRA1 working group. Any errors in documenting or implementing them are due to the author.

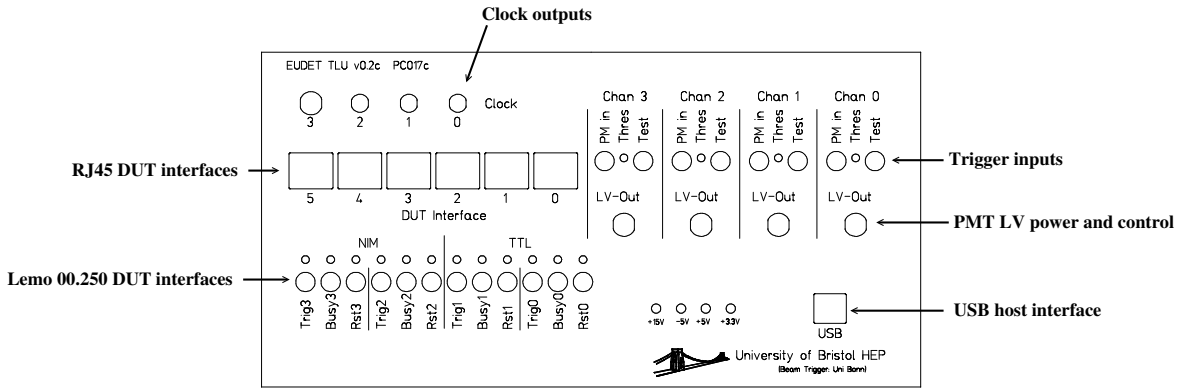

Figure 1: Front Panel of the TLU

1 Introduction

The TLU has LVDS and/or TTL interfaces to the beam-telescope readout and any devices under test, PMT signal and/or NIM level signal interfaces to the beam-trigger and a USB interface to the DAQ. It uses an FPGA housed on an "off-the-shelf" FPGA board[1]. The most up-to-date version of this document can be found at `https://svn.phy.bris.ac.uk/svn/uob-hep-pc017a/trunk/www/Docs/tlu_v0_2_manual_eudet_note.pdf` (username/password = anonymous/anonymous). The TLU v0.2c is a development of TLU v0.1. Firmware and software written for TLU v0.1 can be used with the TLU v0.2 without damage. However with v0.1 firmware the Busy input multiplexing on a TLU v0.2 can not be controlled (it defaults to the RJ45 inputs) and the LEDs can not be controlled. Circuit schematics for the mother-board, clock-board and LEMO-IO daughter-board are available online[2, 3, 4].

2 Power Requirements

5V @ 1A supplied on a 2.5mm connector on rear panel. Centre conductor positive.

3 Front Panel

The front panel for the TLU v0.2 is shown in figure 1

4 Interfaces

4.1 Control

The ZestSC1 FPGA module uses a Cypress EZ-USB micro-controller to implement a USB 2.0 interface. There are two modes of operation register and block transfer. The

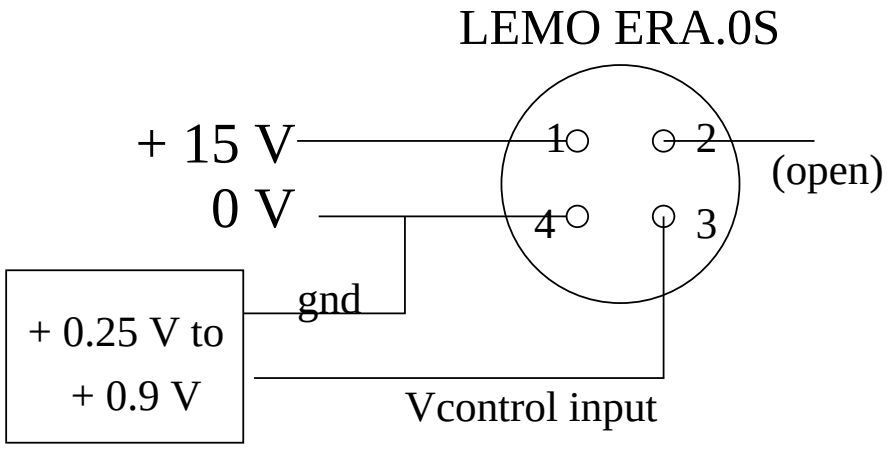

Figure 2: PMT low-voltage connector pin-out

register interface is one byte wide. The block transfer mode transfers blocks of 16-bit words. There is a "B" type connector on front panel.

4.2 Clock I/O

There are four clock connectors. Three Lemo 00.250 connectors carrying TTL levels and one Lemo 0B two-pole connector carrying LVDS levels. By default they are all configured as outputs.

4.3 Beam Trigger

Four Lemo 00.250 single pole connectors on front panel, terminated into 50Ω. Negative going pulses. The discriminator produces fixed length output pulses. If the pulses are too short and/or too close to the threshold narrow "glitches" rather than full length output pulses are produced. Pulse of greater than 3ns at -500mV are sufficient to produce correctly recognized pulses. The signal from the "Test" outputs can be used to monitor the output of the discriminators (terminate test outputs into 50Ω at oscilloscope for a 21:1 probe of the discriminator output). Details of the discriminator unit can be found elsewhere[5].

4.4 PMT power

Four 4-pole LEMO-OS connectors, marked LV-out, provide power for internally powered PMT bases. Each connector has a supply voltage (by default 15V) and also a control voltage (0 - 1V). The control voltage on each output can be independently controlled using an I^2C DAC. Figure 2 shows the pin-out of the connector together with the recommended control voltage when using Hamamatsu H5773 PMT modules.

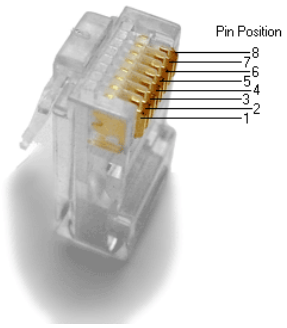

Figure 3: 8P8C (RJ45) connector pin-out

4.5 LVDS DUT Interface

Six RJ45 connectors on front panel, terminated into 100Ω. LVDS levels. Pin out at TLU[2]:

1. TRIGGER_CLOCK- (input to TLU)
2. TRIGGER_CLOCK+ (input to TLU)
3. BUSY- (input to TLU)
4. RESET- (output from TLU)
5. RESET+ (output from TLU)
6. BUSY+ (input to TLU)
7. TRIGGER- (output from TLU)
8. TRIGGER+ (output from TLU)

Connector pins numbering illustrated figure 3

4.6 Lemo DUT Interfaces

Four of the six DUT interfaces (0 to 3) can be connected to Lemo 00.250 connectors. Trigger, Busy and Reset but not Trigger-Clock signals are connected. Outputs (Trigger, Reset) can be active at the same time as the corresponding LVDS outputs, but only one Busy input (either the LVDS or the Lemo) can be active for each DUT. The source of the Busy input (LVDS / Lemo) is selected using the I^2C bus (see section 10).

[2]There is an error in the note describing the TLU v0.1 dated 30th August 2007 and earlier. The pinout direction is reversed (ie. 1 is swapped with 8, 2 with 7 etc). TLU's have never been manufactured in this configuration.

4.6.1 NIM LEMO DUT Interfaces

Lemo interfaces 2 and 3 use (pseudo) NIM levels.
The Busy inputs are terminated into 50Ω to ground. The threshold is -0.5V. The busy inputs can tolerate indefinitely an input voltage of $\pm 5V$. The reset and trigger outputs produce a 0V, -1V level when terminated into 50Ω to ground.

4.6.2 TTL LEMO DUT Interfaces

Lemo interfaces 0 and 1 use TTL levels.
The Busy inputs are terminated into 50Ω in series with 100pF to ground. They are protected against negative voltages by a Schottky diode clamp and can tolerate indefinitely being driven with a NIM "1" level. They can be connected to 5V TTL signal levels or 3.3V TTL signal levels (with lower noise immunity).
The Reset and Trigger outputs supply 3.3V LVTTL signals. They are series terminated by 50Ω. They can drive 50Ω to ground without damage, though in this case the high level output voltage is reduced due to the series termination by a factor of two.

5 Handshake between TLU and DUT

There are three modes of hand-shake between the TLU and the DUT. A "no-handshake" mode , a "simple handshake" and a "trigger data handshake" mode where data is transferred from the TLU to the DUT on each trigger.

5.1 Trigger Data Handshake

1. TLU receives trigger from beam scintillators
2. TLU asserts TRIGGER
3. On receipt of TRIGGER going high, the detector asserts BUSY
4. On receipt of BUSY going high, TLU de-asserts TRIGGER and switches the TRIGGER line to the output of a shift register holding the trigger number/data.
5. The DUT clocks data out of the shift register by toggling TRIGGER_CLOCK. Data changes on the rising edge of TRIGGER_CLOCK[3]. The least significant bit of the trigger data is shifted out first. Only the bottom 15-bits of the 32-bit trigger counter are clocked out. If more than 15 clock pulses are issued on the TRIGGER_CLOCK line the TRIGGER output is set to zero. The DUT should issue 16 clock pulses which will clock out the bottom 15-bits of the trigger number

[3]The Trigger-Clock line is sampled by the internal clock and the rising edge used as a clock enable for a shift register clocked by the internal clock. Hence for an internal clock of 48MHz the the Trigger-Clock should be less than 24MHz and the practical maximum frequency of Trigger-Clock would be in the region of 10MHz.

and return the TRIGGER line to logical low. This will avoid glitches on the TRIGGER line when the DUT returns the BUSY line to logical low.

6. After clocking out the trigger number (and the detector being ready to take more data, the DUT de-asserts BUSY)

7. System is ready for triggers again.

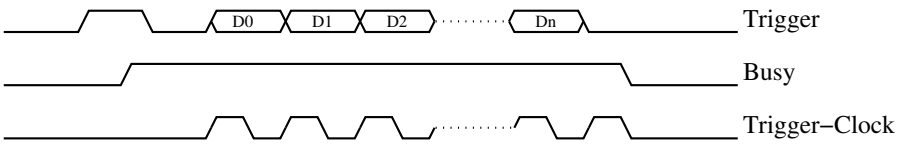

Figure 4: Timing of signals in "Trigger Data Handshake"

5.2 Simple Handshake

1. TLU receives trigger from beam scintillators

2. TLU asserts TRIGGER

3. On receipt of TRIGGER going high, the detector asserts BUSY

4. On receipt of BUSY from DUT, the TLU de-asserts TRIGGER

5. On receipt of TRIGGER going low and the detector being ready to take more data, the DUT de-asserts BUSY

6. System is ready for triggers again. (state of Trigger-Clock is irrelevant in this mode)

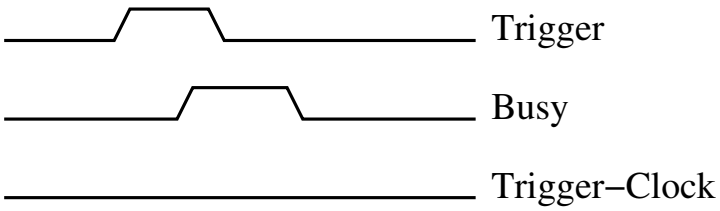

Figure 5: Timing of signals in "Simple Handshake"

5.3 No-Handshake

In this mode the TLU issues a fixed-length pulse on the trigger line (default pulse width is two cycles of the internal clock).

5.4 Selecting Handshake Mode

The "no-handshake" mode for a DUT is done by writing '0' into the corresponding bit in the register at HANDSHAKE_MODE_ADDRESS . Either the "trigger-handshake-mode" or the "simple-handshake" mode can be used if the bit in HANDSHAKE_MODE_-ADDRESS is set.

5.5 Using TRIGGER_CLOCK line to veto triggers

The DUT can raise the TRIGGER_CLOCK line outside a Trigger/Busy handshake sequence. If the corresponding bit in the ENABLE_DUT_VETO register is set then when a TRIGGER_CLOCK line goes high no further triggers are issued by the TLU until the TRIGGER_CLOCK line is brought low again. However, the DUT should still monitor the TRIGGER line for a few hundred nano-seconds after raising its TRIGGER_CLOCK line, since due to the latency in the cable between DUT and TLU it is possible for the TLU to issue a trigger even after the DUT has raised the TRIGGER_CLOCK line.

6 Triggering

Triggers are generated by combinations of the four trigger inputs. There are two ways of specifying what combinations will cause a trigger: triggers masks and trigger patterns.

6.1 Trigger Masks

There are two trigger masks, the "AND" mask and the "OR" mask:

```
otrig <= ( trigger(0) and omask(0) ) or
         ( trigger(1) and omask(1) ) or
         ( trigger(2) and omask(2) ) or
         ( trigger(3) and omask(3) ) ;

atrig <= ( trigger(0) or not amask(0) ) and
         ( trigger(1) or not amask(1) ) and
         ( trigger(2) or not amask(2) ) and
         ( trigger(3) or not amask(3) );

trigger <= otrig or atrig;
```

For example, to demand that a trigger on all four of the inputs before issuing a trigger write 0xF to BEAM_TRIGGER_AMASK_ADDRESS. To produce an output trigger if a trigger arrives on any of the inputs write 0xF to BEAM_TRIGGER_OMASK_ADDRESS. Writing 0x00 to BEAM_TRIGGER_AMASK_ADDRESS disables the "AND-mask".

6.2 Trigger Patterns

More complicated trigger combinations can be produced by writing to the 16-bits at TRIGGER_PATTERN_ADDRESS. Each bit of the pattern corresponds to a state of the trigger inputs. For example bit 0xA (binary "1010") of the 16-bit word controls the output of the trigger when inputs 0 and 2 are low and inputs 1 and 3 are high. For example to produce a trigger if there is a trigger on any of the inputs write 0xFFFE to the trigger pattern. To demand that there is a trigger on all of the inputs write 0x8000. Obviously don't write 0xFFFF to the trigger pattern.

6.3 Auxiliary Trigger

There is an axillary trigger, programmed to writing to the 16-bits at AUX_PATTERN_-ADDRESS. The number of pulses produced by the axillary trigger can be read at REG-ISTERED_AUX_COUNTER_ADDRESS. The auxiliary trigger was mainly introduced to allow the efficiency of the input scintillators to be estimated.

6.4 Trigger Scalers

There is a 16-bit scaler connected to each of the trigger inputs. There is a 32-bit scaler connected to the output of the trigger logic, accessible at REGISTERED_PARTICLE_-COUNTER_ADDRESS. These scalers are always active, with a dead-time determined by the pulse width of the input discriminators (typically 400ns, controlled by R,C values in the discriminator circuit).

7 Dead-time

The TLU is sensitive to triggers except when:

- The trigger veto is set by writing to bit '0' of TRIG_INHIBIT_ADDRESS. Writing '1' to bit '0' (LSB) vetoes triggers. Writing '0' to bit 1 re-enables triggers Reading gives current state of veto in bit-0 and current state of overall veto (including vetoes caused by beam_trigger) in bit-1
- One or more DUTs are holding their BUSY lines high in response to a TRIGGER output.
- One or more DUTs are holding their TRIGGER_CLOCK lines high and the corresponding bits in ENABLE_DUT_VETO are set.
- The timestamp buffer is full. Writing '0' to bit '0' of BUFFER_STOP_MODE_-ADDRESS will turn off this behaviour and triggers will continue even after the timestamp buffer overflows and the buffer pointer wraps round to the start of the timestamp buffer.

8 Timing

8.1 Clock

By default a 48MHz clock on the Zest-SC1 is used for timing. However, a clock board can be fitted to the TLU which contains a CDE929 PLL chip. Depending on configuration (selectable by changing links on a header) the PLL generates clock either from an input reference clock or a quartz crystal. One LVDS and two TTL output signals are connected to the front panel. An additional internal clock signal is connected to the TLU FPGA. Hence the TLU time-stamps can be synchronized to a clock also distributed to the devices under test. By default the clock from the clock board is 40MHz, but this can be varied over a wide range under I^2C control.

8.2 Time-stamps

A 64-bit time-stamp is recorded for each trigger issued. The timestamp resolution is 1/8 of the clock period. So, for example, with a 40MHz clock the time-stamp resolution is 3.125ns. When the timestamp counter is reset a pulse is sent to active DUTs on the RESET line. This together with the clock outputs from TLUs equipped with a clock board enable the DUT to synchronize with the TLU time-stamps. The state of the trigger inputs can be recorded for each trigger. If bit zero of WRITE_TRIGGER_-BITS_MODE_ADDRESS is set to 1 the value of the four trigger inputs will be written to the top four bits of the timestamp.

8.3 Trigger Latency

The time interval between the beam-trigger arriving at the TLU and trigger(s) being issued to the DUT(s) is $27.3 \pm 3ns$. The uncertainty is an indication of the difference in the delay time between different DUT outputs (different logic routing inside the FPGA) and shifts due to fluctuation in temperature and supply voltage. The discriminators used are housed on a daughter-board and are of a fixed-threshold design. There will be "timing-walk" (variation in latency) with varying pulse heights. With negative going pulses of 1ns fall-time the latency varies by about 1ns for pulse heights between -100mV and -600mV For a given DUT output at a fixed pulse-height, temperature and supply voltage the fluctuation in trigger latency between successive triggers is much lower than this. This trigger latency jitter is 31ps RMS, measured the lab. Electrically noisy environments may result in a larger jitter than this. The latency measurements were made using only a single TLU and there will be some variation between TLUs.

8.4 Timing Strobe

As an additional method of synchronization for slower devices the TLU can produce a timing strobe. By default this 3.3V LVTTL output is connected to the "Clock 0" connector. The sequence of commands for enabling the strobe are as follows:

1. Write to STROBE_PERIOD_ADDRESS the repetition frequency of the strobe in units of clock cycles.
2. Write to STROBE_WIDTH_ADDRESS the time that the strobe line should be high in each cycle.
3. Write **1** to bit-0 of STROBE_ENABLE_ADDRESS
4. Reset the timestamp counter.

The strobe starts running simultaneously with the timestamp counter being reset. To turn off the strobe, write **0** to STROBE_ENABLE_ADDRESS and the strobe will stop immediately.

9 Host Interface

9.1 Memory Mapped Interface

Table 1 lists the registers in the memory mapped interface. Multi-byte quantities are accessed in a "Little-Endian" manner, i.e. "ADDRESS" stores the the least significant byte, ADDRESS+1 the next most significant byte and so on.

Location	Number of bytes	Read/ Write	Description
FIRMWARE_ID_ADDRESS	1	R	
DUT_BUSY_ADDRESS	1	R	Value of the busy lines coming from DUTs
DUT_RESET_ADDRESS	1	W	Asserts reset line on DUTs for one clock cycle.
DUT_TRIGGER_ADDRESS	1	W	Bits 0-5 asserts trigger line on the corresponding DUT(s) for one clock cycle. Active even when beam-triggers have been inhibited by writing to TRIG_INHIBIT_ADDRESS. Activates TRIGGER/BUSY handshake (for DUT where the HANDSHAKE_MODE is set to "1")
DUT_MASK_ADDRESS	1	R/W	Bits 0-5 sets the active DUT sockets.
TRIG_INHIBIT_ADDRESS	1	R/W	Writing "1" to bit 0 (LSB) vetoes triggers. Writing "0" to bit 1 re-enables triggers Reading gives current state of veto in bit-0 and current state of overall veto (including vetoes caused by beam_trigger) in bit-1
RESET_REGISTER_ADDRESS	1	W	Writing "1" to a bit issues a reset. Bit mapping: TIMESTAMP_RESET_BIT 0 TRIGGER_COUNTER_RESET_BIT 1 BUFFER_POINTER_RESET_BIT 2 TRIGGER_FSM_RESET_BIT 3 TRIGGER_SCALERS_RESET_BIT 6 CLOCK_GENERATOR_RESET_BIT 7
INITIATE_READOUT_ADDRESS	1	W	Puts block transfer state machine into INITIATE_TRANSFER state.

STATE_CAPTURE_ADDRESS	1	W	Writing any value to this address causes the buffer pointer, timestamp counter, trigger counter and trigger scalers to be registered
TRIGGER_FSM_STATUS_ADDRESS	1	R	Status of the finite state machines controlling the trigger outputs. 0=idle
BEAM_TRIGGER_FSM_STATUS_ADDRESS	1	R	Not used. Returns 0x07
DMA_STATUS_ADDRESS	1	R	Returns "1" if DMA controller is trying to transfer data.
REGISTERED_BUFFER_POINTER_ADDRESS	2	R/W	Location of time-stamp buffer that will written on next trigger
REGISTERED_TIMESTAMP_ADDRESS	8	R	64-bit value of the time-stamp captured when STATE_CAPTURE_ADDRESS is written to.
REGISTERED_TRIGGER_COUNTER_ADDRESS	4	R	32-bit value of trigger-counter
BUFFER_POINTER_ADDRESS	2	R	Next location of time-stamp buffer to be written. Like the other multi-byte unregistered counters can increment between reading high-byte and low-byte. Use registered buffer counter to ensure atomicity
TIMESTAMP_ADDRESS	8	R	timestamp - not registered
TRIGGER_COUNTER_ADDRESS	4	R	trigger counter - not registered
BEAM_TRIGGER_AMASK_ADDRESS	1	R/W	AND mask for input triggers. Bottom 4 LSB significant. Writing 0x00 disables this trigger.
BEAM_TRIGGER_OMASK_ADDRESS	1	R/W	OR mask for input triggers
BEAM_TRIGGER_VMASK_ADDRESS	1	R/W	Not used in v253
INTERNAL_TRIGGER_INTERVAL	1	R/W	Controls interval between internal triggers. Units of milli-seconds. Writing 0x00 disables internal triggers

BEAM_TRIGGER_IN_ADDRESS	1	W	Bit-0 is the output of input combination logic.
DUT_RESET_DEBUG_ADDRESS	1	R/W	Writing "1" to a bit sets RESET line of the corresponding DUT
DUT_DEBUG_TRIGGER_ADDRESS	1	R/W	Bits 0-5 control the TRIGGER line of the corresponding DUTs. This is a static level (unlike writing to DUT_TRIGGER_ADDRESS which causes a single clock cycle pulse). This level is "or-ed" with trigger signals generated in response to beam triggers before being output to the TRIGGER lines. Writing to this register does not activate the TRIGGER/BUSY handshake.
DUT_CLOCK_DEBUG_ADDRESS	1	R	State of DUT_CLOCK lines
DUT_I2C_BUS_SELECT_ADDRESS	1	R/W	Controls which of the I2C buses is controlled by accesses to DUT_I2C_BUS_DATA_ADDRESS
DUT_I2C_BUS_DATA_ADDRESS	1	R/W	Controls/Monitors state of SDA and SCL lines on the bus selected by DUT_I2C_BUS_SELECT_ADDRESS
CLOCK_SOURCE_SELECT_ADDRESS	1	R/W	Controls the source of the clock used for time-stamps. "'1" selects 48MHz crystal. "0" selects output of clock-generator board.
TRIGGER_IN0_COUNTER	2	R	Number of counts from input 0 since last reset.
TRIGGER_IN1_COUNTER	2	R	Number of counts from input 1 since last reset.

TRIGGER_IN2_COUNTER	2	R	Number of counts from input 2 since last reset.
TRIGGER_IN3_COUNTER	2	R	Number of counts from input 3 since last reset.
REGISTERED_PARTICLE_COUNTER_ADDRESS	4	R	Number of triggers that passed the input combination mask. Registered by writing to STATE_CAPTURE_ADDRESS. The number of triggers that would have occured if there was no dead-time
REGISTERED_AUX_COUNTER_ADDRESS	4	R	The numbers of triggers that passed the "aux-counter" combination mask.
HANDSHAKE_MODE_ADDRESS	1	R/W	Set bit to "1" for a DUT to use trigger/busy handshake. Set to "0" for no-handshake
BUFFER_STOP_MODE_ADDRESS	1	R/W	Set bit-0 to "1" and triggers will be inhibited when timestamp buffer fills.
WRITE_TRIGGER_BITS_MODE_ADDRESS	1	R/W	Set bit-0 to "1" and the value of the trigger inputs will be written to the top four bits of the timestamp
TRIGGER_PATTERN_ADDRESS	2	R/W	Treats the four trigger inputs as input to a LUT. Output of LUT is connected to TRIGGER
AUX_PATTERN_ADDRESS	2	R/W	Treats the four trigger inputs as input to a LUT. Output of LUT is connected to AUX_TRIGGER scaler.
STROBE_WIDTH_ADDRESS	4	R/W	Width of strobe output (in clock cycle periods)
STROBE_PERIOD_ADDRESS	4	R/W	Period of strobe output (in clock cycle periods)
STROBE_ENABLE_ADDRESS	1	R/W	Writing "1" to bit-0 enables strobe output

TRIGGER_FSM_STATUS_VALUE_ADDRESS	3	R	4-bits for each DUT. Each nibble shows state of corresponding trigger output FSM. 0x0 = IDLE 0x1 = WAIT_FOR_BUSY_HIGH 0x2 = TRIGGER_DEGLITCH_DELAY1 0x3 = TRIGGER_DEGLITCH_DELAY2 0x4 = WAIT_FOR_BUSY_LOW 0x5 = DUT_INITIATED_VETO 0xF = error
ENABLE_DUT_VETO_ADDRESS	1	R/W	Writing "1" allows the relevant DUT to assert the trigger veto by rising its DUT_CLOCK line

Table 1: Address map for TLU memory-mapped interface

Bit	Read/ Write	Function
0	W	Writing 0 pulls SDA line low.
1	R	State of SDA line.
2	W	Writing 0 pulls SCL line low
3	R	State of SCL line.

Table 2: Bit functions of DUT_I2C_BUS_DATA_ADDRESS register

9.2 Block Transfer Interface

The time-stamps are transferred using the block-transfer interface. Each 64-bit timestamp is transferred as four 16-bit words, the least significant word is transfer-ed first.

10 I2C bus

The multiplexing of the Busy inputs and the indicator LEDs is using I2C. In order to give the possibility of adding extra daughter-boards controlled by the existing motherboard there are a number of separate I2C buses. With the current version of the firmware a "bit-banging" interface for the I2C SDA and SCK lines is used. Which I2C bus is selected is controlled by writing to the DUT_I2C_BUS_SELECT_ADDRESS register. Data is written to and read from DUT_I2C_BUS_DATA_ADDRESS. Table 2 shows the format of the DUT_I2C_BUS_DATA_ADDRESS register. Table 3 enumerates the I2C devices.

11 Readout sequence

When the TLU receives a trigger it writes the current value of the 64-bit time-stamp into a buffer and the BUFFER_POINTER is incremented. The trigger counter is also incremented. To order to perform an "atomic" read on the value of BUFFER_POINTER, TRIGGER_COUNTER and TIMESTAMP, write (any value) to STATE_CAPTURE_ADDRESS. The values are then captured and stored in REGISTERED_BUFFER_POINTER, REGISTERED_TRIGGER_COUNTER and REGISTERED_TIMESTAMP. To read out the buffer of timestamps, the following sequence should be followed:

1. Write **1** to bit-0 of TRIG_INHIBIT_ADDRESS, this stops further triggers
2. Write **1** to INITIATE_READOUT_ADDRESS
3. Use TLUReadData function on host to transfer content of time-stamp buffer to the host.
4. Write to bit **2** of RESET_REGISTER_ADDRESS to reset the buffer pointer (so that further timestamps are written at the start of the buffer)

Function	I2C bus	I2C address	Device	Description
RJ45 Leds	Motherboard	0	PCA9555	LEDs 0..7 connected to IO0, LEDs 8..11 connected to IO1[3..0]
RJ45 trigger outputs	Motherboard	1	PCA9555	rj45_trigger_enable[3..0] connected to IO1[7..4] (DUT interfaces 4,5 are always enabled)
Lemo trigger outputs	Motherboard	1	PCA9555	lemo_trigger_enable[3..0] connected to IO1[3..0]
RJ45 reset outputs	Motherboard	2	PCA9555	rj45_reset_enable[3..0] connected to IO1[7..4] (DUT interfaces 4,5 are always enabled)
Lemo reset outputs	Motherboard	2	PCA9555	lemo_reset_enable[3..0] connected to IO1[3..0]
Busy LVDS/Lemo select	Motherboard	3	PCA9555	Mapping between DUT and Pins on PCA9555: 0 = IO0[1..0] 1 = IO0[3..2] 2 = IO0[5..4] 3 = IO0[7..6] "11" selects RJ45, "10" selects lemo.
LEMO green LEDs	Lemo	1	PCA9555	IO0[7..0]
LEMO bi-colour LEDs	Lemo	1	PCA9555	IO1[7..0]
PMT DAC	Display	2	AD5316	Four-channel 10-bit DAC. Vref = 0.5V . Set gain=2 to get control voltage between 0V and 1V
Clock generator	Motherboard	0	CDCE949	Consult data sheet for programming details.

Table 3: List of I2C devices.

5. Write **0** to bit-0 of TRIG_INHIBIT_ADDRESS to enable triggers.

12 Grounding and Shielding

The RJ45 connectors for the DUT interface allow shielded cable to be used. **It is recommended that shielded cable is used**. Unlike 10/100/1000-BaseT Ethernet where the signals are DC-balanced and can be AC-coupled, the interface between the TLU and the DUT is not DC-balanced and is DC-coupled. This limits the common-mode voltage between the DUT and TLU to less than $\pm 0.9V$[4]. Excessive current flow through any ground loops formed should be reduced using careful layout of cables and extensive use of ferrite cores round the cables.

Acknowledgements

This work is supported by the Commission of the European Communities under the 6^{th} Framework Programme "Structuring the European Research Area", contract number RII3-026126 and by the UK Particle Physics and Astronomy Research Council, grant number PP/E000673/1

References

[1] `http://www.orangetreetech.com/zestsc1.htm`

[2] `https://svn.phy.bris.ac.uk/svn/uob-hep-pc017a/trunk/www/Schematics/pc017c_toplevel_schematic.pdf`

[3] `https://svn.phy.bris.ac.uk/svn/uob-hep-pc017a/trunk/www/Schematics/pc017c_clock_board.pdf`

[4] `https://svn.phy.bris.ac.uk/svn/uob-hep-pc017a/trunk/www/Schematics/pc017c_lemo_io.pdf`

[5] `https://svn.phy.bris.ac.uk/svn/uob-hep-pc017a/trunk/www/Docs/TLU_Discriminator_Board_Documentation.pdf`

[4] Set by the sn65lvdm1676 transceivers used in the TLU. This common mode range will be even lower if the DUT has a lower common mode range.