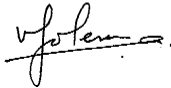


## Project Specification

**Project Name: ATLAS Calorimeter First Level Trigger-  
Cluster Processor Chip**

**Version: 1.0**

**August 2000**

Approval	Name	Signature	Date
Project Manager	V. Perera		3/8/2000

### Distribution for all updates:

<b>Project Manager:</b>	<b>V. Perera</b>
<b>Customer:</b>	<b>A. R. Gillman</b>
<b>Group Leader responsible for Project:</b>	<b>R. J. Halsall</b>
<b>Project Managers of related projects:</b>	<b>R. Staley</b>
<b>Account Manager:</b>	<b>W. J. Haynes</b>

## 1. Scope

This document defines the specification for the cluster processing (CP) chip (ASIC or FPGA) which implements the electron/photon and tau/hadron cluster finding algorithms.

## 2. Related projects and documents

2.1 ATLAS First-Level Trigger Technical Design Report, CERN/LHCC/98-14 ATLAS TDR-12, 30 June 1998:

2.2 Prototype ROD module Specification at <http://hepnts1.rl.ac.uk/Atlas-L1/Modules/Modules.html>

2.3 Serialiser FPGA Specification at <http://hepnts1.rl.ac.uk/Atlas-L1/Modules/Modules.html>

2.4 Prototype CPM Specification at <http://hepnts1.rl.ac.uk/Atlas-L1/Modules/Modules.html>

2.5 Generic Test Module at <http://hepnts1.rl.ac.uk/Atlas-L1/Modules/Modules.html>

## 3. Technical Aspects

### 3.1 Requirements

The most complex part of the logic required for the electron/photon and tau/hadron trigger processor will be implemented on the cluster-processing chip. The cluster processor chip will process a 2 x 4 region of 4 x 4 x 2 trigger tower window and will provide cluster hit information and regions of interest (RoI) information.

There will be eight threshold sets for the cluster and isolation requirements for electron/photon trigger, and for flexibility another set of eight thresholds which could be selectable between tau/hadron and electron/photon trigger via slow controls (control register).

A chip select, 16-bit data bus, 10-bit address bus and read/write strobes will provide all slow controls to the cluster-processing chip.

**Note:** Throughout this document the co-ordinate system used is  $(\phi, \eta)$  and the convention used to describe the calorimeter space is  $\phi \times \eta \times \text{layer}$ , e.g. 2 x 4 region of 4 x 4 x 2 trigger tower window



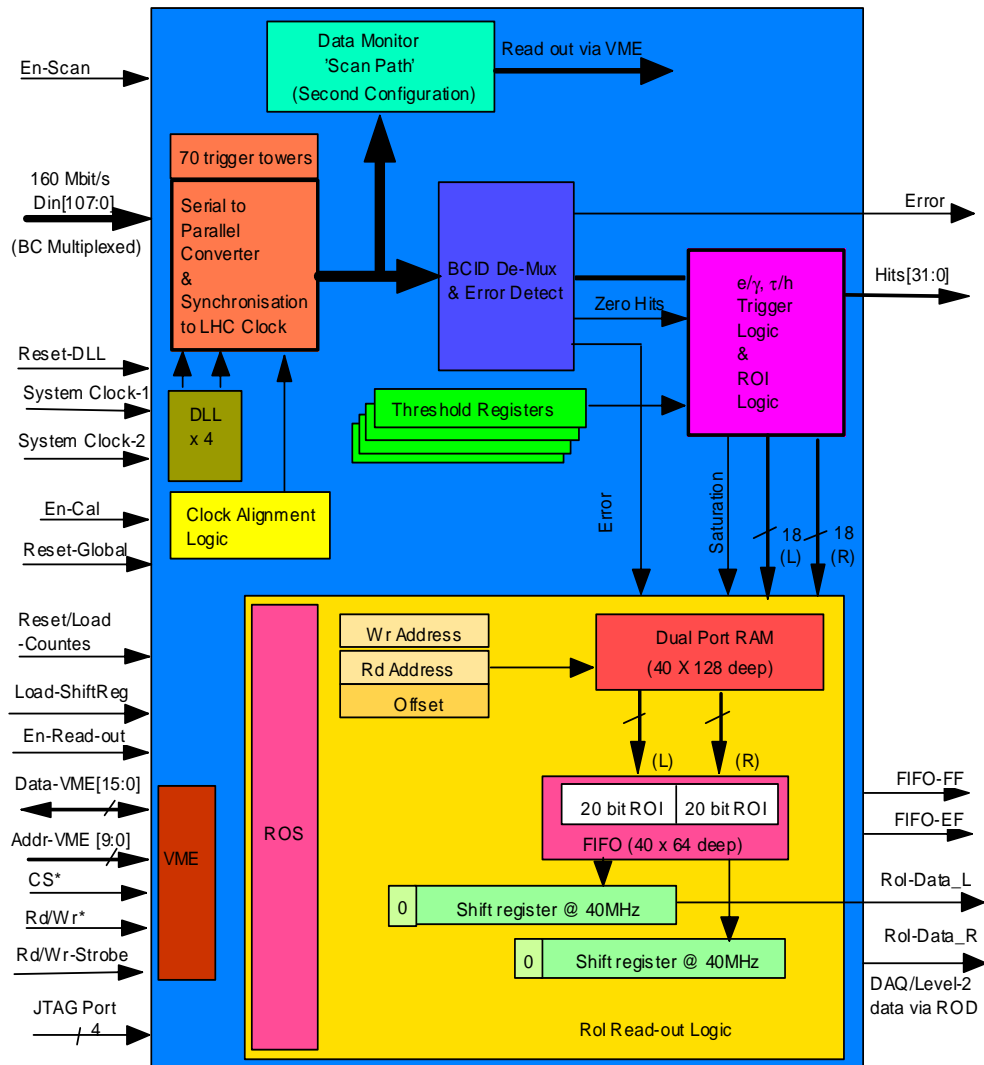


Figure 2. Cluster Processor Chip Block Diagram

### 3.3 Logic Blocks

#### 3.3.1 Real Time Data Path

The real time data path consists of the serial to parallel and clock alignment logic, BC De-Multiplexing and error detection logic, and the algorithm logic block.

The latency estimate for the real time path in the TDR [2.1] is 150 ns (six LHC clock ticks) for an ASIC solution of the CP chip. 150 ns will be the design target for the FPGA solution as well.

#### 3.3.2 Serial to Parallel and Clock Alignment Logic

Unlike the high-speed serial links, where the clock is encoded with the data and is recovered at the receiving end, the 160 Mbit/s data does not have a clock encoding and clock recovery scheme. The function of the clock alignment logic is to select the appropriate 160 MHz clock phase (four phases) to capture the incoming serial data at

160 Mbit/s correctly using different phases of the 160 MHz clock, and then to synchronise to the 25 ns clock period using 6.25 ns delay elements. Once the clock and data is aligned then each four bits within the 25 ns period will be converted to a parallel 'nibble' for the next stage of processing. See Appendix B for details.

### 3.3.3 BC De-Multiplexing and Error Detection

The BC-de-multiplexing [2.1] logic examines 10 bits of data, consisting of an eight-bit field of calorimeter data and the associated flag and error bits. If non-zero calorimeter data are found, two consecutive bunch crossings are processed and the data are assigned to a trigger tower and a bunch crossing according to their flag bits. Those trigger towers that are not assigned data for a bunch crossing are set to zero, with error bits set to indicate no error. BC de-multiplexing transforms each 10-bit field into an 18-bit field, containing data for two trigger towers. See Appendix C for more details.

This block also carries out error checking on the received bits. This should be done in parallel to the BC-de-multiplexing logic. If the error logic finds any of the trigger tower data in error, then an 'error' signal can be used to zero the trigger towers in error so that these trigger towers do not cause false triggers. If a parity error is discovered all data that may be corrupted by this error are zeroed. For the BC-multiplexed data, this consists of the following: data from the current bunch crossing; data from the previous bunch crossing if the error is found in the second bunch crossing of a BC-multiplexed pair; data from all bunch crossings following the parity error until the next empty bunch crossing. This last measure is necessary as corrupt data and BC flags may cause the CP chip to lose synchronisation with the BC-multiplexing scheme and assign data to trigger towers incorrectly. Synchronisation is re-established by empty bunch crossings. The RoI field will indicate if an error was detected.

A mask register (42 bit) is provided so that the error checking can be disabled per pair of trigger towers. A 42 bit register (one per trigger tower pair) will be provided to capture the error condition, which then can be readout (and cleared) via VME to ascertain problematic 160 Mbit/s data links. These errors may be due to problems on the high-speed serial data links between pre-processor and the cluster processor system or due to noise on the backplane links between two adjacent modules. Logical OR of all errors are counted within the CP chip and if the counter is  $> 0$  then this will be indicated on a device pin (Error) so that the CPM can monitor this signal before it needs to read the error register. If there are any problems with a particular data link/links, then those data links causing un-acceptable error rates can be masked out (cleared to zero) using the data link mask register (see memory map).

### 3.3.4 Trigger Algorithms

The cluster processor chip will process a 2 x 4 tower trigger window from em and hadron calorimeters, requiring a (2+3) x (4+3) x 2 environment from em and hadron calorimeters, as shown in figure 3. The trigger towers 11, 12, 13, 14, 21, 22, 23 and 24 references the trigger tower windows within one cluster processor chip.

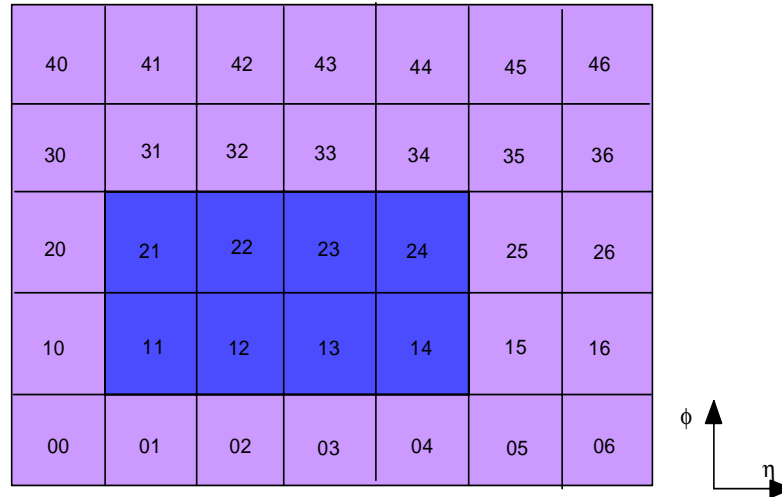


Figure 3. CP cluster processor chip Processing Window

The algorithm, using a 4 x 4 sliding window will search for isolated electromagnetic (em) energy clusters and tau candidates and provide triggers and regions-of-interest (RoIs). Figure 4 shows the 4 x 4 sliding window below applied to the reference trigger tower 11. The cluster processor chip has to apply this sliding window in steps of one trigger tower in both  $\phi$  and  $\eta$  directions to trigger towers 11, 12, 13, 14, 21, 22, 23 and 24 (see figure 3 above).

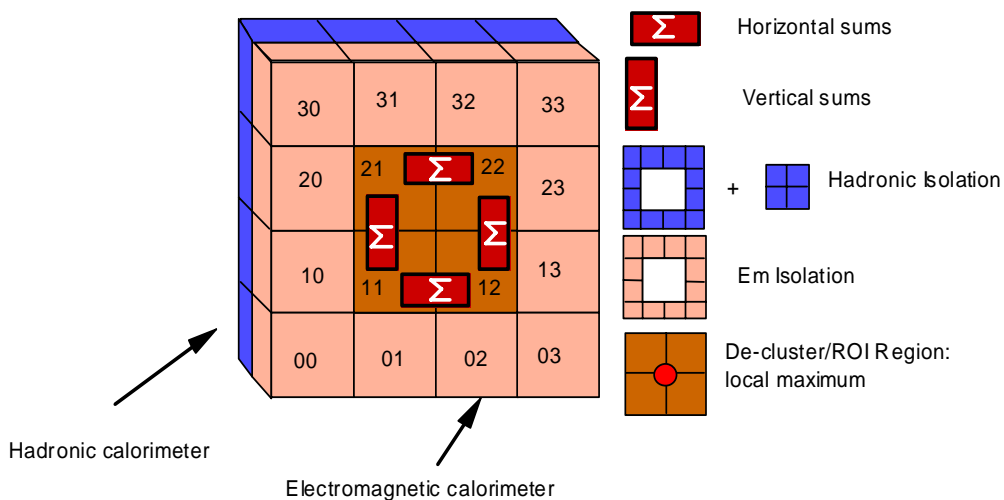


Figure 4. 4 x 4 x 2 arrangement of trigger towers used for the Algorithm

### 3.3.4.1 Isolated electromagnetic (em) energy cluster trigger:

The following shows the logic required to process the trigger window referenced by trigger tower 11, which covers a 2 x 2 RoI cluster formed by trigger towers 11, 12, 21 and 22 (em and had).

(a) The isolation requirement is based upon three  $ET$  sums, one based on em calorimeter and the other two based upon the hadron calorimeter:

$$(i) \text{ IsoSum}_{em} = (00 + 10 + 20 + 30 + 31 + 32 + 33 + 23 + 13 + 03 + 02 + 01)$$

$$(ii) \text{ IsoSum-1}_{had} = (00 + 10 + 20 + 30 + 31 + 32 + 33 + 23 + 13 + 03 + 02 + 01)$$

$$(iii) \text{ IsoSum-2}_{had} (11 + 21 + 22 + 12)$$

All above isolation sums will have a 6-bit range, which is the maximum value for an isolation threshold. On overflow these sums will be saturated to 3F Hex.

(b) Cluster condition:

$$\begin{aligned} & 11_{em} + 12_{em} > \text{cluster threshold (a)} \\ \text{OR} & 11_{em} + 21_{em} > \text{cluster threshold (a)} \\ \text{OR} & 21_{em} + 22_{em} > \text{cluster threshold (a)} \\ \text{OR} & 12_{em} + 22_{em} > \text{cluster threshold (a)} \\ \text{AND} & \text{ IsoSum}_{em} \leq \text{isolation\_threshold (b)} \\ \text{AND} & \text{ IsoSum-1}_{had} \leq \text{isolation threshold (c)} \\ \text{AND} & \text{ IsoSum-2}_{had} \leq \text{isolation threshold (d)} \end{aligned}$$

Cluster sums (e.g.  $11_{em} + 12_{em}$ ) will have an 8-bit range, which is the maximum value for a cluster threshold. On overflow these sums will be saturated to FF Hex

Note: a = eight possible 8-bit cluster threshold values + up to eight more selectable.  
b, c, d = eight possible 6-bit isolation threshold values + up to eight more selectable. These selections will be done via a control register.

(c) De-cluster condition: (11-bit range)

De-clustering (RoI local maximum) is the process of eliminating double counting of trigger objects by selecting the window with the best cluster containment.

The algorithm requires that the Et sum (em and had) in the centre 2 x 2 cluster (11, 12, 21, 22) be greater than (or equal to) all eight other 2 x 2 clusters which can be formed within the same 4 x 4 window, and which coincide with the RoI cluster of the eight neighbouring windows. (9 possible 2 x 2 clusters within 4 x 4 window). For the electron/photon as well as tau/hadron algorithm, the de-clustering is performed on both the em and hadron layers.

$$\begin{aligned} \text{Sum}_{em + had} (11, 12, 21, 22) & \geq \text{Sum}_{em + had} (00, 01, 10, 11) \\ \text{AND} & \\ & \geq \text{Sum}_{em + had} (10, 11, 20, 21) \text{ AND} \\ & \geq \text{Sum}_{em + had} (20, 21, 30, 31) \text{ AND} \\ & \geq \text{Sum}_{em + had} (01, 02, 11, 12) \text{ AND} \end{aligned}$$

- >  $\text{Sum}_{em + had} (21, 22, 31, 32)$  AND
- >  $\text{Sum}_{em + had} (02, 03, 12, 13)$  AND
- >  $\text{Sum}_{em + had} (12, 13, 22, 23)$  AND
- >  $\text{Sum}_{em + had} (22, 23, 32, 33)$

(d) Trigger condition:

For a given threshold set, the trigger hit bit is set if the cluster condition AND the de-cluster condition are met by a single window.

Since there could be only two possible hits in a 2 x 4 - window cluster processing chip, and since there are eight electron/photon cluster thresholds, the number of trigger hit bits out of the cluster processor chip for the electron/photon will be 16 (two hit bits per threshold set). However if other thresholds out of the eight selectable thresholds are assigned for the electron/photon then the number of hit bits will be up to 2 x the number of thresholds selected for the electron/photon cluster logic.

### 3.3.4.2 Tau trigger

Considering the same 4 x 4 trigger towers as before but the sums are formed as shown below then discriminated at multiple thresholds (up to eight). The isolation rings on the two layers are treated independently.

(a) The isolation requirement is based upon two  $E_T$  sums as in section 3.3.5.1 above.

(i)  $\text{IsoSum}_{em}$

(ii)  $\text{IsoSum}-1_{had}$

(b) Cluster condition:

$$\{ \text{Sum}_{em} (11, 12) + \text{Sum}-2_{had} \} > \text{cluster threshold (e)}$$

OR  $\{ \text{Sum}_{em} (11, 21) + \text{Sum}-2_{had} \} > \text{cluster threshold (e)}$

OR  $\{ \text{Sum}_{em} (21, 22) + \text{Sum}-2_{had} \} > \text{cluster threshold (e)}$

OR  $\{ \text{Sum}_{em} (12, 22) + \text{Sum}-2_{had} \} > \text{cluster threshold (e)}$

AND  $\text{IsoSum}_{em} \leq \text{isolation threshold (f)}$

AND  $\text{IsoSum}-1_{had} \leq \text{isolation threshold (g)}$

Cluster sums will have an 8-bit range, which is the maximum value for a cluster threshold. On overflow these sums will be saturated to FF Hex

Note: e = up to eight possible 8-bit threshold values.

f, g = up to eight possible 6-bit isolation threshold values



(b) De-cluster condition:

The de-clustering is performed using em and hadron information summed together as for the electron/photon algorithm and the logic is identical to the electromagnetic cluster algorithm above.

(c) tau trigger condition:

The trigger hit bit is set if the cluster condition AND the de-cluster condition is met. Since there could be two hits and there could be up to eight thresholds, the number of trigger bits out of the cluster processor chip for tau triggers could be up to 16.

### 3.3.4.3 Handling Saturation

Saturation condition could be due to:

1. The trigger towers that are sent from the pre-processor are already in saturation (analogue saturation), which the pre-processor indicates by setting the trigger tower to FF hex.
2. Saturation due to overflow within the sums used in the algorithm.

To guarantee that a saturated trigger tower gives rise to hit, the isolation veto requirement should be switched off for at least for one threshold set from either the electron/photon or tau/hadron trigger

If saturation occurs due any of the above conditions then the appropriate hit bits will be set. Only for case one above (FF hex received from the pre-processor) will the saturation condition be indicated in the RoI field.

### 3.3.4.4 Region of Interest (RoI)

RoIs are generated for each hit. The hit information (multiplicity) is passed on to the CTP via the merger module, whereas the RoI information is passed on to the second level trigger. The RoI information consists of the location of the RoI ( $\eta$ ,  $\phi$  indices), and bits indicating the thresholds it has passed. Within 2 x 2 group of windows, the RoI logic will select the one with the most energetic RoI cluster (2 x 2 x 2 towers, see de-clustering). In a cluster processor chip which processes 2 x 4 trigger windows, there are two such groups ([reference towers 11,12,22,21] = 'L', and [reference towers 13,14,24,23] = 'R'). In each 2 x 2 group there could only be one RoI, and it could have passed any or all of the 16 thresholds for em or for tau trigger. If there were no RoIs then all the threshold bits will be set to zero.

If there were any saturated trigger towers (see section 3.3.4.3 above) then this will be indicated with a bit set in the RoI field as shown in figure 5. Similarly, if an error is detected in the BC De-Multiplexing and Error Detection logic (3.3.4) then a bit will be set in the RoI field as shown in figure 5.

The RoI information will be transferred out of the chip serially on two pins in the format shown in figure 5.

All the RoI information within one cluster processor module will be merged and transferred out using a single G-Link onto a read-out driver (ROD) module for processing (zero suppression) and transferring to the second level trigger.

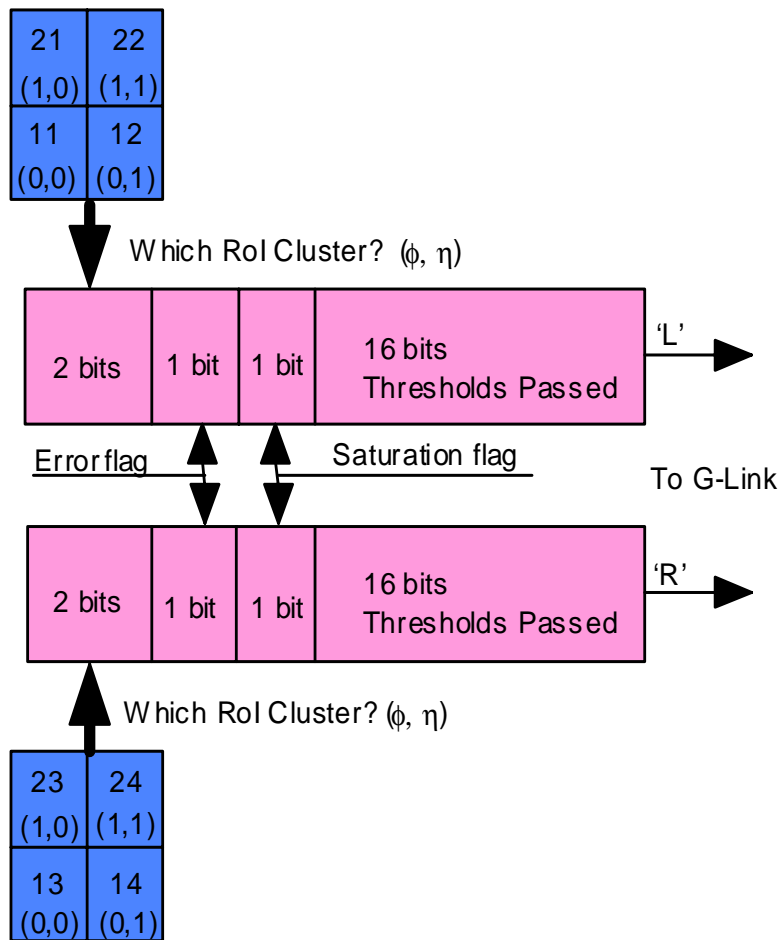


Figure 5. RoI Format

### 3.3.5 RoI Read out Logic

Figure 6 shows a block diagram of the read-out associated logic on the cluster processor chip. On each bunch crossing the data from the algorithm block will be written on to the dual-port memory. The results (RoI co-ordinates and hits) must be sent on to two destinations, to Level-2 trigger, and to the DAQ via the ROD and ROB modules. Since the RoI information includes the hit information, only the RoI records will be written on to the dual-port memory.

The purpose of the dual-port RAM is to hold a history of RoI data. RoI data are written to the dual-port RAM and read from it at addresses given by the Wr Address Counter and Rd Address Counter respectively. These 7-bit counters increment every bunch crossing and roll over to zero; RoI data is thus stored for 128 bunch crossings before being over-written. The Rd Address Counter is offset from the Wr Address Counter by the value held in the Offset register. This offset should be set up to compensate for the latency of the trigger system from the time at which RoI data is written into the dual-port RAM, to the time at which the En-readout signal for that RoI data is received by the CP chip. The Rd Address Counter and Wr Address counter are reset and reloaded, respectively, by the Reset/Load Counters signal.

The read out of the RoI data is controlled by the En-readout and Load-ShiftReg signals, which are generated on the CPM by the read-out controller (ROC) and are sampled by the CP chip synchronously with the rising edge of the system clock. If, on this clock edge, En-readout is high, the data pointed to by the Rd Address Counter is copied from the dual-port RAM to the FIFO buffer. The CP chip output FIFO-EF indicates the status of the FIFO buffer: if FIFO-EF flag is low the FIFO contains data.

If, on a rising clock edge, Load-ShiftReg is high, RoI data from the FIFO are transferred to the shift register and shifted out of the CP chip at 40 MHz. This shift register is permanently enabled, and once the valid 20 bits of data are shifted out zeros will follow.

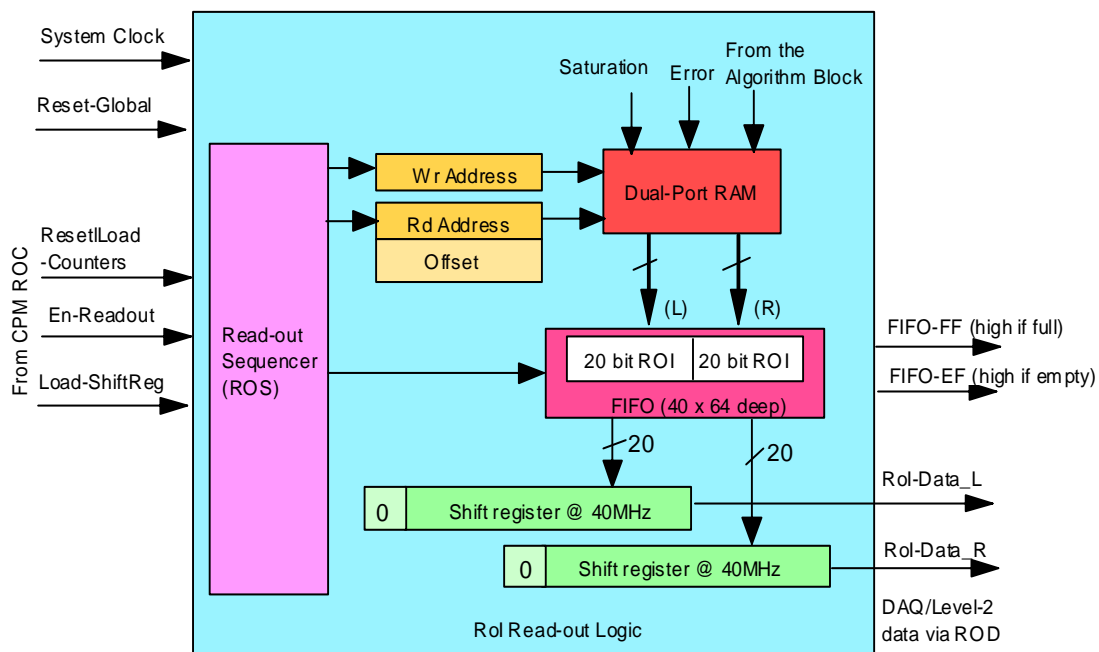


Figure 6. RoI Read-Out Associated Logic

### 3.3.5.1 Requirements of the CP chip for the CPM ROC

The readout controller (ROC) logic on the CPM controls the RoI readout of the CP chip. The following signals are used:

- En-readout: input to the CP chip from the ROC
- Load-shift: input to the CP chip from the ROC
- Reset/Load Counters: input to the CP chip from the ROC.
- FIFO-EF: output from the CP chip to the ROC.

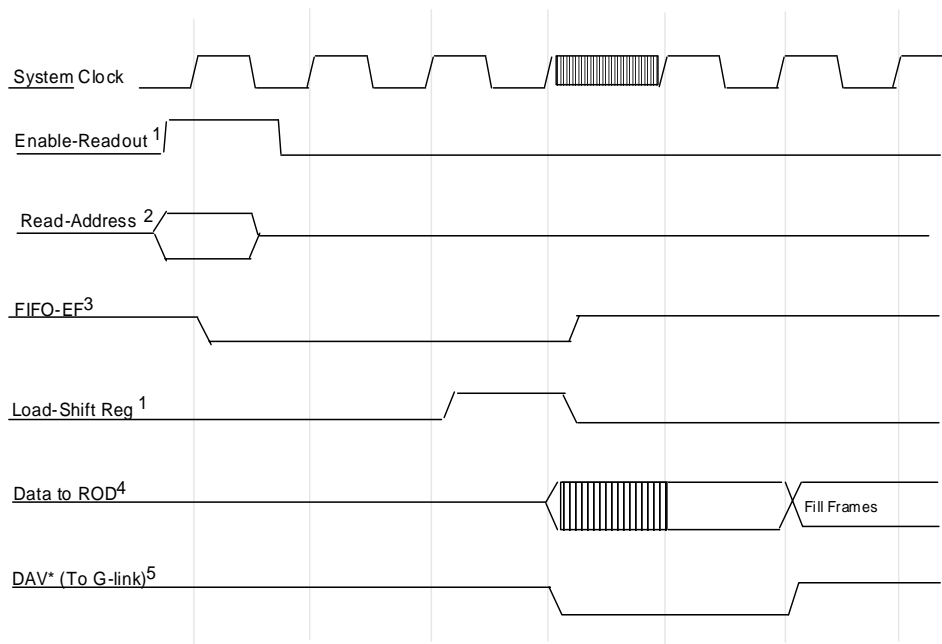
The CP chip samples the En-readout, Load-ShiftReg and Reset/Load Counter signals synchronously with the rising edge of the board clock. On receipt of an L1A the ROC should read out the RoIs from the CP chip using the following procedure.

1. Assert the En-readout signal to the CP chip. The RoI data will be transferred within the CP chip from the dual-port RAM to the FIFO buffer.

2. Monitor the FIFO-EF output from the CP chip. When this is low there are data in the CP chip FIFO and the Load-ShiftReg signal may be asserted (see below).
3. Assert Load-ShiftReg for one bunch crossing. This causes the RoI data to be read from the FIFO and loaded into the shift register. The data is then clocked serially out of the CP chip at 40 MHz.
4. Wait for 20 bunch crossings. This is the time taken to output the RoI data from the CP chip. En-readout may be asserted again during this period (which should happen if another L1A arrives), but Load-ShiftReg should not, as this would overwrite the data contained in the shift register.

Figure 7 shows a timing diagram for the signals needed to control the read out of the CP chip.

In addition to the above process, the ROC also supplies the Reset/Load Counters signal to the CP chip. This clears the Wr Address Counter and loads the Rd Address Counter with the value contained in the Offset register. To avoid discontinuities in the data written to the dual port RAM, the Wr Address Counter should only be cleared when it is due to roll over to zero. This will also correspond to the point at which the Rd Address Counter is due to reach its Offset value. In normal operation, therefore, the Reset/Load Counters signal will have no effect; it will only effect counters that have lost synchronisation.



- Note:
1. Input signals to the Serialiser FPGA from the ROC logic on the CPM
  2. Internally generated on the Serialiser FPGA
  3. Output from Serialiser FPGA to the ROC
  4. Data output to ROD from the G-Link via the ROC
  5. Data available signal from the ROC to G-link to frame the DAQ data

Figure 7. Timing diagram shows the signalling requirements for controlling the readout sequence.

### **3.3.6 PLL or DLL**

The PLL or DLL block (depends on choice of technology) will generate the 160 MHz clock from the 40 MHz system clock and will provide the different phases of the 160MHz clock internal to the cluster processor chip for capturing the incoming 160 Mbit/s data.

If the CP chip is implemented on Xilinx Virtex-E FPGA, then it will require two 40 MHz clock inputs with an offset between them of  $22.5^{\circ}$  (1.56ns). This is because the FPGA can produce only two phases of the 160 MHz clock from a single input ( $0^{\circ}$  and  $180^{\circ}$ ). Therefore, to generate four phase of a 160 MHz clock, the two 40MHz clocks described above are required.

### **3.3.7 Scan Path for Monitoring Input Data**

Once the data have been converted to parallel and re-synchronised to the 40 MHz system clock, these data go directly to the BC de-multiplexing logic and algorithm logic and are not available for checking the synchronisation process. For this reason a scan path is provided to capture the data (560 bits per clock cycle) and to read them out serially, 16 bits at a time via VME. This check on the synchronisation process need only be carried out at the initial set-up time. Therefore this check can be done with a second FPGA configuration (see appendix D) without implementing the 'checking' logic with the 'real time' cluster processor logic, saving many gates.

## **3.4 Technology**

### **3.4.1 ASIC or FPGA**

A synthesisable model will be developed and targeted to Xilinx Virtex-E FPGA family. If cost and latency issues are acceptable, then given the flexibility of FPGAs the choice will be confirmed. Alternatively if these issues are not satisfactory, then ASIC technology such as Chip Express will be targeted.

### **3.4.2 Input Output Technology**

3.4.2.1 All inputs and outputs will be 3.3 volts CMOS compatible

3.4.2.2 Set-up and hold times will be specified after selecting the appropriate input buffers.

3.4.2.3 Clock to output times will be specified after selecting the appropriate output drivers.

### 3.4 Operating Conditions

Parameter	Symbol	Min	Typ	Max	Unit
Supply Voltage core	Vddc	- 5%	1.8	+ 5%	V
Supply Voltage I/O	Vddi/o	- 5%	3.3	+ 5%	V
Output Drive Low/High	$I_{OL}/I_{OH}$			24	mA
Output Low Voltage	$V_{OL}$	GND		0.6	
Output High Voltage	$V_{OH}$	2		3.3	
Input Low Voltage	$V_{IL}$	GND		0.8	V
Input High Voltage	$V_{IH}$	2.0		3.3	V
Clock	CLK	-	-	40	MHz
Junction Temperature	$T_I$			85	$^{\circ}\text{C}$
Power (estimate)	P(3V3)			100	mW
Power (estimate)	P1V8)			4.5	W

### 3.6 Signal Definition

Name	Pin	Type	Signal
Addr-VME[9:0]		I	10 bit address lines for accessing memory and control registers
CS*		I	Chip select signal, enables read/write access to the memory and registers via slow controls.
Data-VME[15:0]		I/O	Bi directional data bus for slow controls.
En-Calibration		I	Enables the calibration data pattern
En-Readout		I	Initiates the readout sequence
En-Scan		I	Enables the Scan read-out logic
Error		O	Parity error detected in data transmission from PPr
FIFO-EF		O	FIFO Empty flag (high if empty)
FIFO-FF		O	FIFO Full flag (high if full)
DIN[107:0]		I	108 bits @ 160 Mbit/s
Hits[31:0]		O	16 bit hits (two bits per threshold)
Load-ShiftReg		I	Loads the shift register from the FIFOs
Reset-DLL		I	Resets the DLLs
Reset-Global		I	Resets all registers except for the control registers.
Reset/Load Counters		I	Signal to reset the internal write counter and load the read counter with the offset value
Rd/Wr-Strobe		I	Dual port memory read and write strobe.
Rd/Wr*		I	Dual port memory read and write enable signal.
RoI-Data_L		O	Serialised read-out data
RoI-Data_R		O	Serialised read-out data
System Clock-1		I	40 MHz system clock input.
System Clock-2		I	Delayed 40 MHz system clock input.
TD0		O	JTAG output
TDI		I	JTAG input
TCK		I	JTAG Clock
TMS		I	JTAG Mode select
Vdd-1		Supply	+ 3.3 Volts supply for periphery
Gnd-1		Supply	Ground periphery
Vdd-2		Supply	+ 1.8 Volts supply for core
Gnd-2		Supply	Ground core

NOTE: I = input; O = output; I/O = bi-directional

Total number of I/O = 187

## 3.7 Programming Model

### 3.7.1 Guidelines

These are to aid the software control of the cluster processor chip.

1. All registers can be read by the computer, hence there are no 'write only' registers.
2. All Status Registers shall be Read-Only registers.
3. All Control Registers shall be Read/Write registers.
4. Data integrity cannot be guaranteed if a computer tries to read or write a value that the cluster processor chips itself is able to modify at the same time.
5. The power-up condition of all registers will be all zeros, unless otherwise stated.

### 3.7.2 Memory Map

The cluster processor chip is addressed using a chip select (CS). It has 16 bit data field and a 10 bit address bus.

Register type	Register Name	Bits	Description
RO	Version Register	16	Firmware version and revision
RW	Control Register	16	BC Mux On/Off, em/tau threshold selection, etc
RO	Status register	08	FIFO Full, Empty flags, etc
RW	Threshold [16..1]	08	16 Cluster Threshold Registers
RW	Isolation [48..1]	12	48 Isolation Threshold Registers
RO	Error_counter	16	Count errors from 42 error registers
RO	FIFO Write/Read Counters	16	Two 7-bit registers for FIFO Read [0:6] and Write [8:14] counters
RW	Error-Mask	16	Error Mask Bits [16:1]
RW	Error-Mask	16	Error Mask Bits [32:17]
RW	Error-Mask	16	Error Mask Bits [42:33]
RW	Error-register	16	Error registers to record errors on [16:1]
RW	Error-register	16	Error registers to record errors on [32:17]
RW	Error-register	16	Error registers to record errors on [42:33]
RW	Link-Mask	16	Mask links [16:1]
RW	Link-Mask	16	Mask links [32:17]
RW	Link-Mask	16	Mask links [42:33]
RW	Clock Phase[22:1]	10	22 x 10 bit registers for selecting a clock phase. Two bits per 160 Mbit/s serial line
RW	Clock delay [22:1]	10	22 x 10 bit registers for selecting 6.25 ns delay. Two bits per 160 Mbit/s serial line
RO	Sync Done [11:1]	10	11 x 10 bit registers to read back synchronisation done signal
RW	128 x 40 Dual-port RAM	16	Dual port RAM bits [15:0]
RW	128 x 40 Dual-port RAM	16	Dual port RAM bits [31:16]
RW	128 x 40 Dual-port RAM	08	Dual port RAM [39:32]



RW	40 x 64 FIFO	16	FIFO bits [15:0]
RW	40 x 64 FIFO	16	FIFO bits [31:16]
RW	40 x 64 FIFO	08	FIFO bits [39:32]
RW	Scan Register	16	Scan path registers to read out 16 time slices x 432 bits of data from the synchronisation block

RO means that the computer can only read the value of this register, writing has no effect.

RW means that the computer can affect the state of the cluster processor chip by writing to this register.

- **Note:** The Memory map and Pin-out may change during the design phase to accommodate any design requirements not foreseen at the time of writing this document

### 3.8 Handling Precautions

The device will be static sensitive, hence must be handled with proper care.

### 3.9 Design Methodology

The designs will be fully synchronous, and will be described in VHDL and/or Verilog at register-transfer-level (RTL) then synthesised to gate level.

### 3.10 CAE

CAE tools such as; Cadence, Synopsis, VHDL/Verilog, Renoir, Leonardo and Modelsim will be used depending on the vendor support.

### 3.11 Design Verification

Functional, post synthesis, post layout, timing verifications and design rule checks will be carried out before the design is signed off.

### 3.12 Testability

To make it easy to test the cluster processor chip, the following will be considered and implemented where possible. If any of these cannot be implemented, then the reasons should be documented and an alternative approach taken.

#### 3.12.1 Power up and Reset Strategy

The following issues will be considered in the design:

1. Asynchronous reset
2. Provision for internal/external power-up reset
3. Provision for resetting individual logic blocks for debugging purposes where necessary.
4. Reset and start up for PLL/DLL

### **3.12.2 PLL/DLL bypass**

Facility to bypass the phase-locked loop will be required if you intend driving the chip at lower frequencies. Minimum frequency depends on the choice of technology.

### **3.12.3 Boundary scan, internal scan path and ATPG**

JTAG boundary scan will be provided. This could be used for initial board level testing (interconnect)

### **3.13 Built-in-self-tests (BIST) (only if ASIC)**

Wherever RAM blocks are used, use BIST to test the RAMs without the need for hand coded test vectors.

### **3.14 Manufacturing (only if ASIC)**

RAL will provide the relevant design files to the vendor (Chip Express for example) after the design has been verified (3.8). The vendor will manufacture prototype devices for test and verification before a final order is placed for the production devices. The CPM system requires approximately 500 devices for the final system.

### **3.15 Testing**

#### **3.15.1 Strategy**

The testing will be done in two stages:

1. The manufacturers will carryout testing at 1MHz with the test vectors provided by RAL. (only if ASIC)
2. Using a purpose built test card (Generic Test Module), full speed tests will be carried out.

#### **3.15.2 Test equipment**

- (1)VME crate
- (2)16/32 bit VME interface
- (3) Computer to run software
- (4) Generic Test Module
- (5) Logic analyser
- (7) Oscilloscope
- (8) VME Extender

### **3.16 Storage, Shipping & Installation**

The devices will be stored in the anti-static storage area at RAL in R25a, and will be sent to the customer or to the board manufacturer when required. A stock control sheet will be maintained at RAL.

### 3.17 Maintenance and further orders

Approximately 10 devices are required for the prototype CPMs. The production modules require approximately 500 devices. Device obsolescence needs to be considered when placing the order for the production CPMs to include the appropriate number for spares.

### 3.18 Software

A test engineer from the System Support Group will develop the LabView test software to drive the test card.

## 4. Project Management

### 4.1 Personnel

		RAL Ext.	RAL Location
Customer:	A. R. Gillman	5521	R1, 1.54
Project Manager:	V. Perera	5692	R68, 2.31
Project Engineer	J. Edwards	6814	R68, 2.09

### 4.2 Deliverables

#### 4.2.1 To the Customer:

1. Prototype cluster processor chips
2. Specification document

#### 4.2.2 From the Customer:

Software to drive the Generic Test Module may be required

### 4.3 Project plan (Milestones)

1. Preliminary Design Review (PDR) Q4 1999
2. Final Design Review (FDR)
3. Start Testing Q4 2000
4. Concluding Review (CR)

### 4.4 Design Reviews

The customer must be present at the PDR and the CR. If the customer wishes, he may attend the FDR.

The progress of the project will be reported on a monthly basis in the ATLAS Calorimeter First-level Trigger project monitor form.

### 4.5 Training

Training will be carried out as required on the job.

## **4.6 Costs and Finance**

All manufacturing, assembly and component costs will be charged to FK40000.

## **4.7 Intellectual Property Rights (IPR) and Confidentiality**

All background and foreground Intellectual Property Rights in this project will remain with CLRC. The customer will have unrestricted rights to items listed under deliverables (4.2). If the customer requires other data, then an appropriate protective agreement should be in place before releasing such data.

## **4.8 Safety**

General laboratory safety codes apply.

## **4.9 Environmental impact**

None

### ***4.9.1 Disposal***

RAL will dispose of the devices at end of their life.

### ***4.9.2 EMC***

Since these are components, they will be outside the scope of the EMC regulations. However the modules incorporating these devices must function as designed, without malfunction or unacceptable degradation of performance due to electromagnetic interference (EMI) within their intended operational environment, the electronic modules shall comply with specifications intended to ensure electromagnetic compatibility.

## Appendix A - Input Pin Count

To process the trigger window of  $2 \times 4$ , the cluster processor chip will require information from  $((4 + 3) \times (2 + 3)) \times 2 = 70$  em and hadron trigger towers. The 8-bit trigger tower data will be serialised onto two 160 Mbit/s data streams to reduce the high number of input pin requirements on the cluster processor chip and the backplane. Serialising the trigger towers will reduce the cluster processor chip pin count from 560 to 140 pins. Using the BC multiplexing scheme [2.1] this number can be further reduced to 108 pins.

With the BC multiplexing scheme five 160 Mbit/s lines are required to carry serialised data for four trigger towers or 3 lines for two trigger towers (data + Bc Mux information [control])

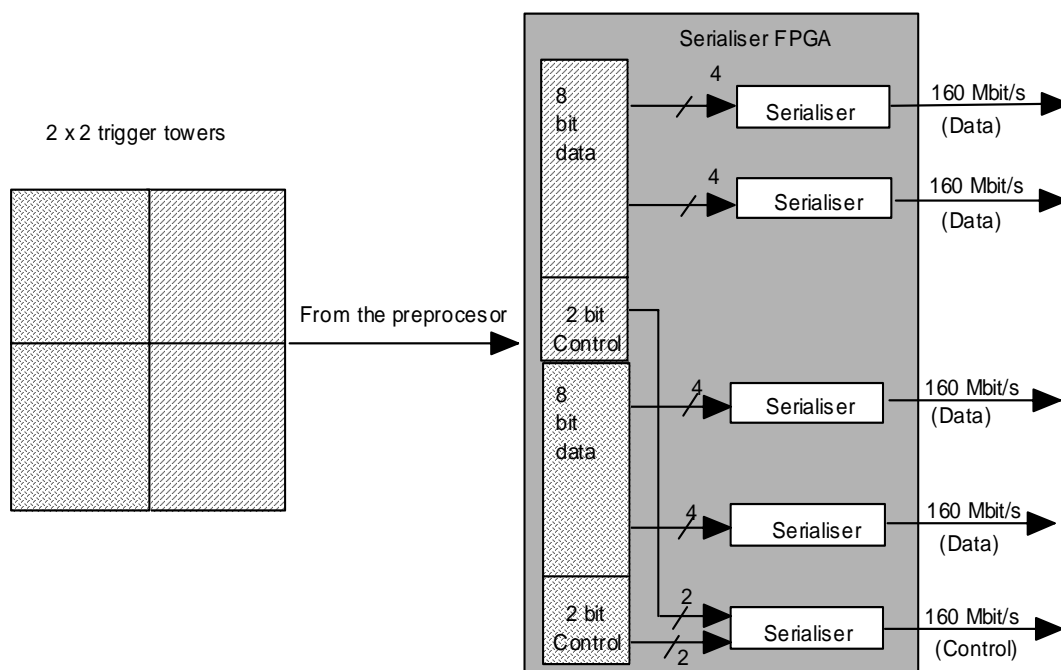


Figure A1. Five lines to carry two  $1 \times 2$  BC multiplexed pairs arranged in  $2 \times 2$

Although only a  $5 \times 7 \times 2$  of trigger towers are required for the CP algorithm, due to the above arrangement of data transmission to the CP chip,  $6 \times 7 \times 2$  trigger towers will be taken into the CP Chip as follows.

The nine  $2 \times 2$  trigger towers (e.g. 00, 01, 10, 11) will need  $9 \times 5$  lines  $\times 2 = 90$  pins, and the remaining column (06, 16, 26, 36, 46, 56) consisting  $2 \times 1$  trigger towers will require  $3 \times 3$  lines  $\times 2 = 18$ . Therefore a total of 108 pins will be required to bring the trigger towers to the CP chip.

Note: the data from the top row (50, 51, 52, 53, 54, 55, 56) is not required and therefore will be disregarded in the chip

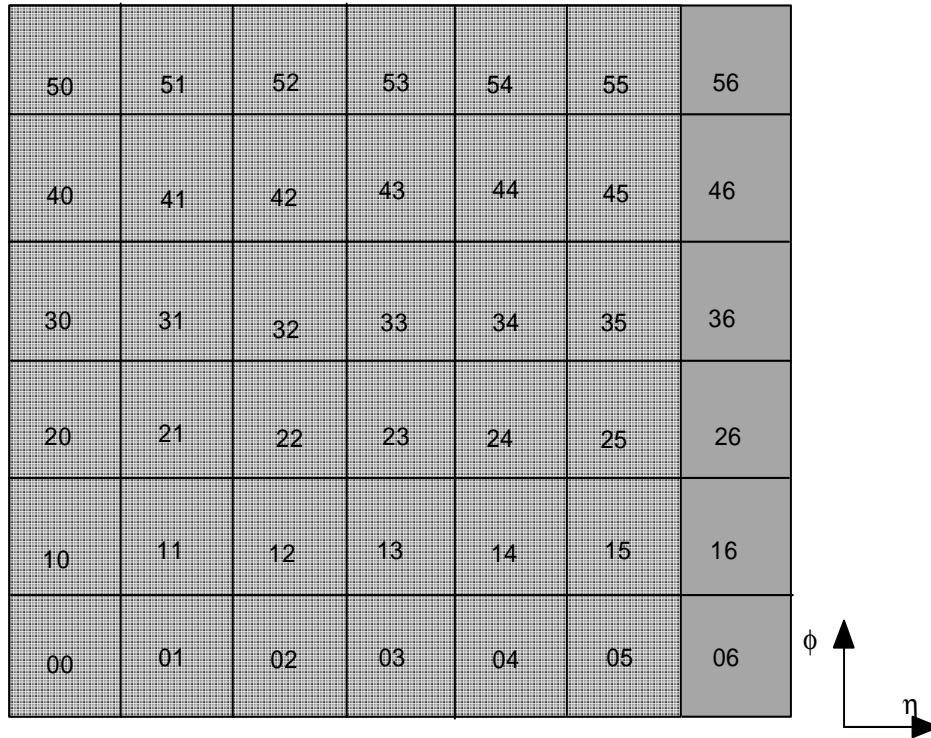


Figure A2. 6 x 7 trigger towers are required to bring 5 x 7 trigger towers

## Appendix B - Clock Calibration Scheme

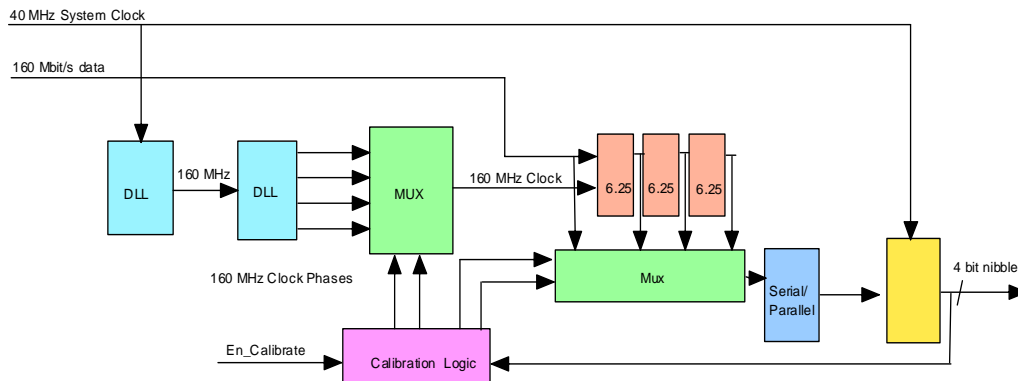


Figure B1. Clock and Data Alignment Logic

A repetitive pattern of 10100101 must be transmitted for the duration of the calibration run. Note this pattern is implemented on the Serialiser FPGA. Calibration is achieved in two stages:-

### Stage 1 - Lock onto the "best" clock phase

A histogram is constructed by varying the clock select bits (four phases) and sampling the output of the serial to parallel converter. The algorithm allows 7 samples to be taken at each delay setting. The peak is used at the "best" clock phase. Figure B2 shows the block diagram of this procedure.

### Stage 2 - Set the delay select bits

Once the correct clock has been selected a sample is taken from the serial to parallel converter with the delay bits set to zero. Depending on the word read the correct delay setting (steps of 6.25 ns) could be calculated.

### Calibration check

An internal check is carried out to ensure that calibration has been successfully achieved and flagged via a dedicated output pin. This is done by waiting for up to 4 samples after calibration then checking that in the next 8 samples the 4 bit word is 1010 or 0101.

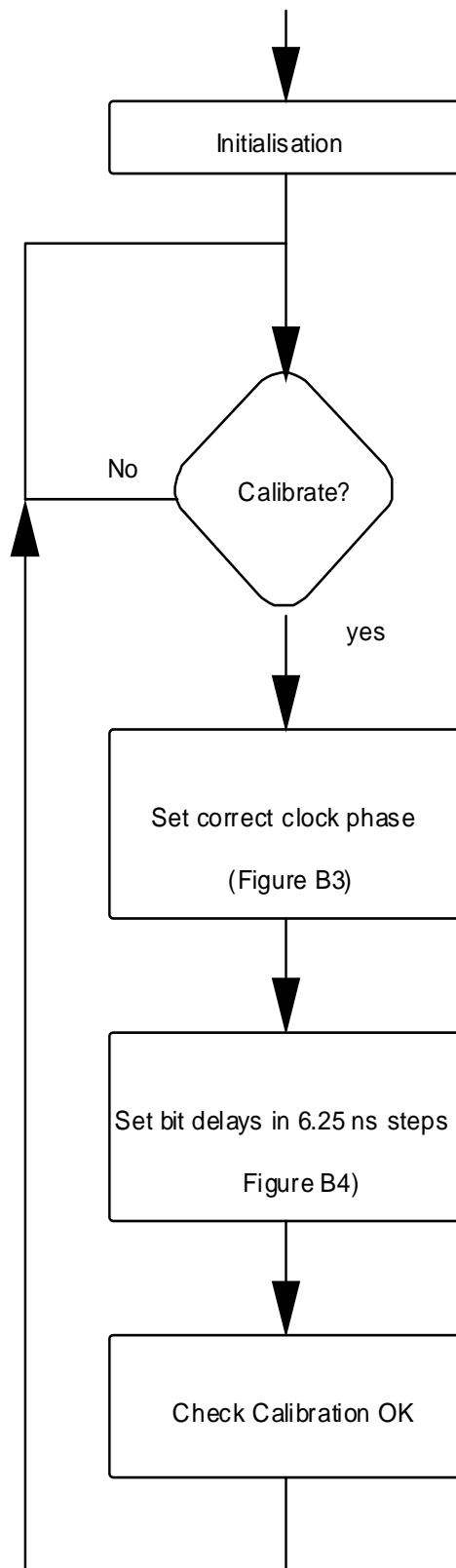


Figure B2. Clock Calibration Scheme



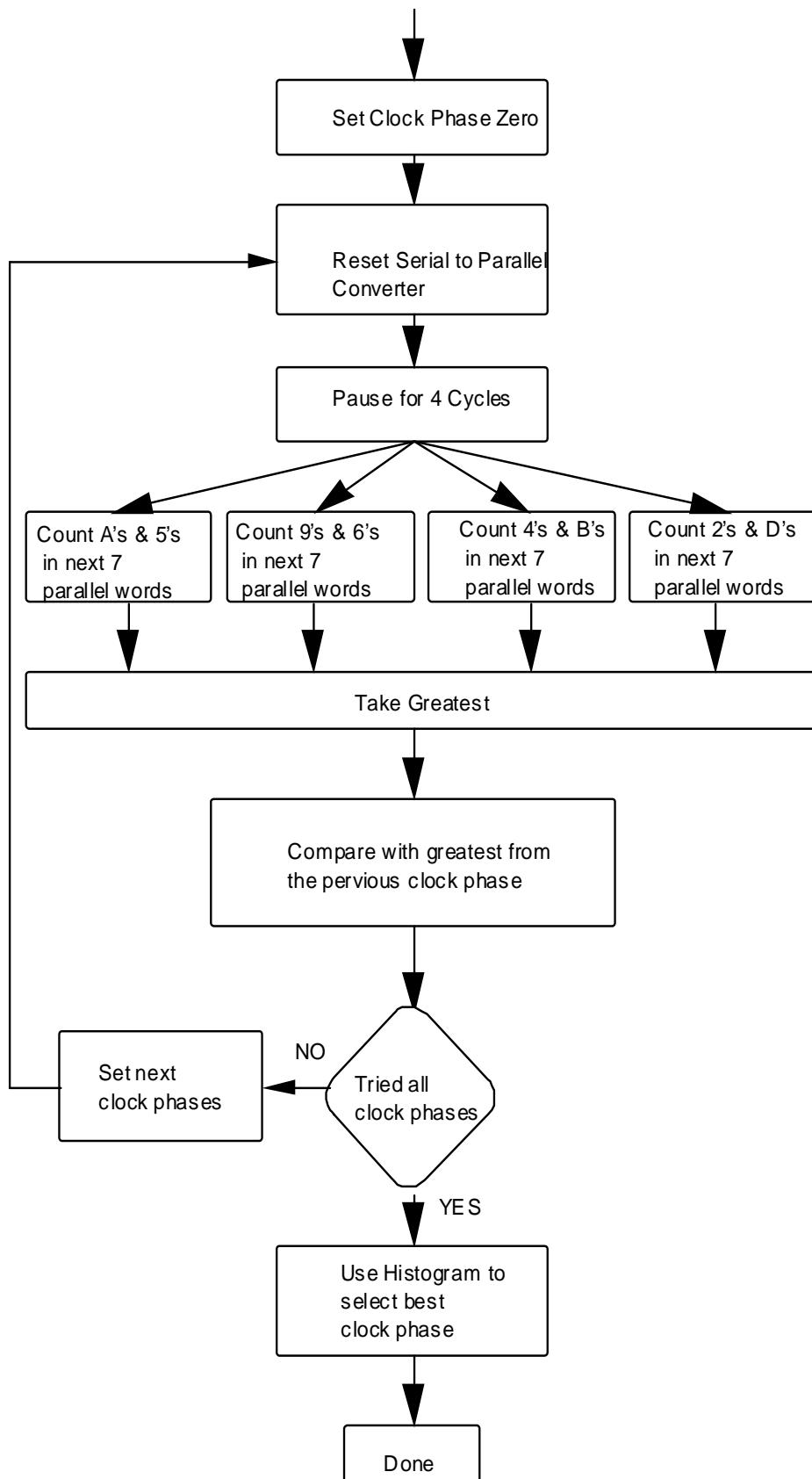


Figure B3. Clock Select Stages

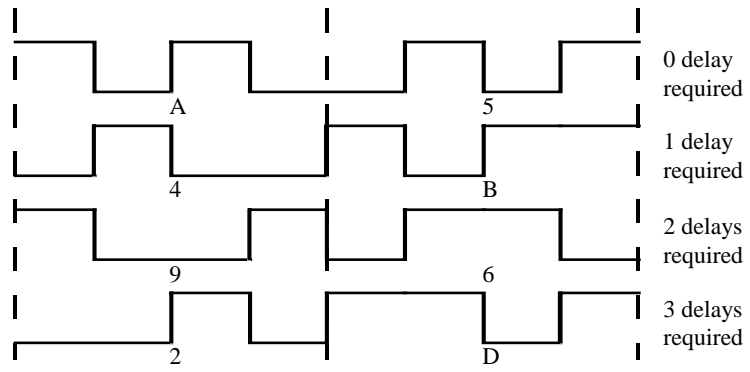


Figure B4. 6.25 ns Delay Adjustment

## Appendix C — BC de-multiplexing

Figure C1 shows the logic required to de-multiplex one channel of BC-multiplexed data. This logic receives as input the following:

- eight bits of trigger-tower data and one parity bit (grouped together as DATA in the diagram);
- one bit comprising the BC multiplexing flag (BCflag in the diagram).

The output from this logic is the data for two trigger towers.

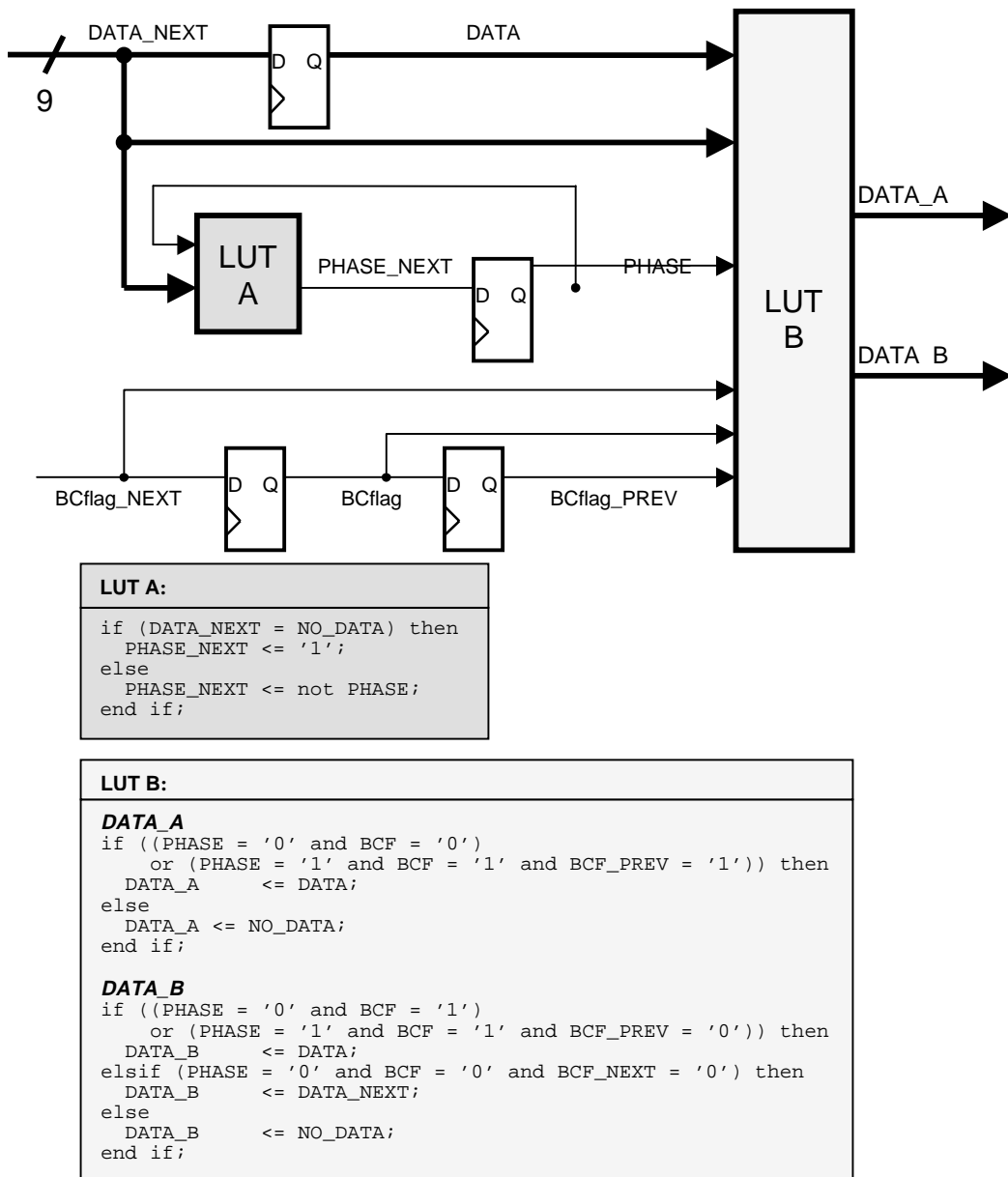


Figure C1. The BC de-multiplexing logic. VHDL is used to describe the contents of LUT A and LUT B. The constant NO\_DATA, used in the VHDL, is defined as zero input data with odd parity.

## Appendix D - Scan Path Logic

This will be a separate configuration to monitor the correct operation of serial to parallel conversion and synchronisation logic. There will be 432 bits per time slice from the serial to parallel & synchronisation block. Sixteen time slices will be recorded and read out serially via the VME as shown in the following diagram.

