The RD27 Muon Trigger Co-incidence Array Demonstrator ASIC

R. Bindra, B. Claxton, J. Dowdell, A. Letchford, V. Perera, S. Quinton
Rutherford Appleton Laboratory, Chilton, DIDCOT, OXON., OX11 0QX, UK

N. Filippini, E. Gennari, E. Petrolo, S. Veneziano
INFN Roma, 00185 Roma — Piazzale Aldo Moro, 2, ITALY

N. Ellis
CERN, Geneva, Switzerland

1. ABSTRACT

One aim of the RD27 project is to perform design and R&D work leading to a first level muon trigger for an experiment at the Large Hadron Collider (LHC) at CERN. This paper describes the design, implementation and testing of an ASIC for a trigger demonstrator system.

The trigger system is implemented using a set of seven chambers. The low momentum trigger requires hits in three out of the four inner chambers. The high momentum trigger requires a low momentum trigger and hits in two of three outer chambers. This scheme allows for chamber inefficiencies for real muons and reduces the trigger rate from neutron and photon-induced background in the detectors.

The core of the ASIC is an eight by twenty-four input 'double' co-incidence array allowing two momentum cuts to be applied. The ASIC has multiple inputs per axis and includes the multiplicity logic. The design of the ASIC is flexible enough to demonstrate fully combinatorial operation, fully pipelined operation, or any combination of the two.

The ASIC has been fabricated using a 34k gate, 0.5µm CMOS gate array from Fujitsu. Testing confirms it can be pipelined at above 80MHz or fully combinatorial with a propagation delay of 7ns, varying by up to 2ns depending on input pattern.

2. THE CO-INCIDENCE ARRAY

The coincidence array is a general purpose technique for comparing two vectors and has been used for muon triggering in several experiments [1,2].

Denote the input vectors as X and Y. X has n bits, Y has m. The output vector is denoted Z and has n bits, the same as X.

The coincidence array uses an array of bits, A, of dimension n by m, to specify which coincidences between X and Y are 'valid'.

The value of the coefficient A_{ij} determines the response of the array when inputs X_i and Y_j are '1' simultaneously (Z_{ij}). If the coefficient is '1' then this is a valid combination and the corresponding Z_{ij} output will be '1'. The array output Z_i is then the logical OR of Z_{ij} for all j. If the coefficient is zero then that combination is not valid and the Z_i output depends on the other combinations of the same X_i. If no combination of X_i with Y is valid then the output Z_i is '0'.

Figure 1 is a diagram of a generic coincidence array.

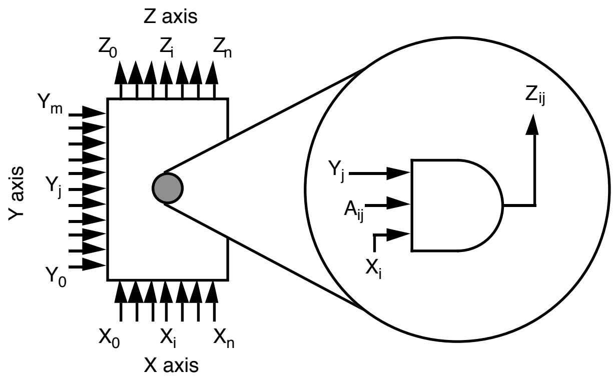

Figure 1 — A Generic Coincidence Array

Each intersection can be thought of as a three input AND gate, producing the AND of the corresponding bit of the input vectors and coefficient array. The Z_i output is then the OR of the outputs of all ANDs in the column (i.e. for the same X_i).

3. THE MUON TRIGGER ALGORITHM

Consider the simplified muon chamber system shown in Figure 2.

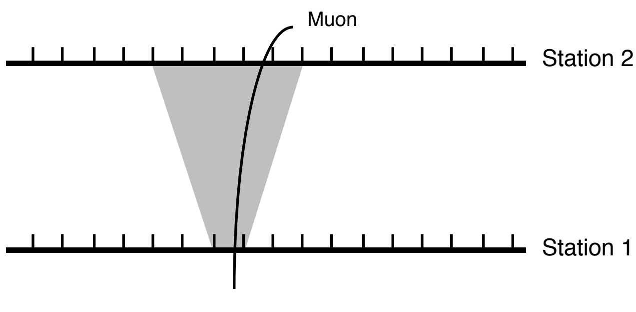

Figure 2 — A simplified muon chamber system

The 'stations' are divided into strips oriented into the page, corresponding to the vertical tick marks shown.

A magnetic field is used to exert a bending force on the muons, causing the curved path shown. The amount of bending is inversely proportional to the momentum of the muon; the lower the momentum, the larger the angle through which the muon is bent.

The trigger recognises muons by comparing the output of the two physically separated planes using a coincidence array. Each hit in station one (on the X axis) is compared with any in a ‘cone’ extended to station two (on the Y axis). The size of the cone is chosen to implement a momentum cut on the muons. It follows that the narrower the cone programmed into the coincidence arrays the higher the momentum of a muon must be for it to pass the trigger. The number of inputs in Y must be greater than that of X because each X input is compared with a number of Y inputs.

Because the size of the cone determines the momentum cut-off applied, we have implemented, in the one device, the equivalent of two coincidence arrays, processing the same inputs, to provide two decisions that correspond to different momentum thresholds.

Figure 3 shows a realistic chamber system.

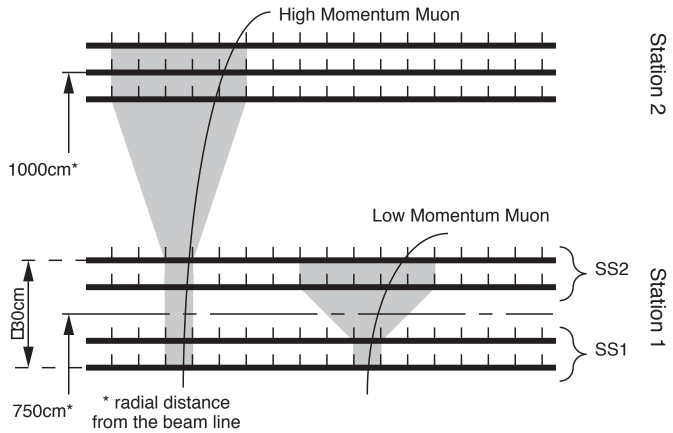

Figure 3 — A possible muon chamber layout

The demonstrator is to be designed to prove the viability of implementing both low and high momentum triggers based on this type of layout. Note that this diagram is not drawn to scale and in practice the width of the stations would be small relative to their separation.

In the baseline design the co-incidence array runs fully combinatorially, with its output sampled at 160MHz, four times the LHC bunch crossing clock. This has been shown in beam tests to produce a trigger that can uniquely identify the bunch crossing that produced the muon, when the chambers have good time resolution, for example RPCs.

In fully combinatorial operation the variation in the delay through the device as a function of input pattern is an important factor in maintaining the ability to uniquely identify the bunch crossing.

The specification for the device is that its propagation delay from input to output be less than 10ns, and the variation in this delay be less than a few nanoseconds.

3.1 Low Momentum Trigger

Low momentum muons are bent through large angles. To simplify the implementation this trigger therefore uses the planes of station one only.

A cone is extended from SS1 to SS2 whose width at SS2 depends on the momentum cut-off desired. Because of the smaller separation between the planes the cone will be smaller for the same momentum cut-off than between stations one and two.

The algorithm uses three of four majority logic at each X/Y intersection, as shown in Figure 4.

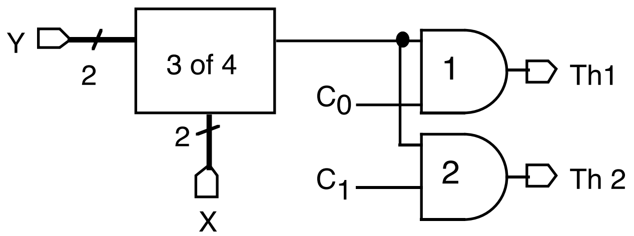

Figure 4 — Low momentum threshold logic per intersection

The X input is connected to SS1, the two corresponding strips providing two inputs per column. Similarly the Y input is connected to the two strips in SS2. C_0 and C_1 are co-efficients that specify whether the first or second threshold has been satisfied, respectively.

This algorithm requires that at least three out of the four strips be hit. This serves two functions:

- It takes into account chamber inefficiency, in order not to suppress muons, and
- It lowers the probability that a combination of random hits, for example induced by low-energy neutrons or photons, will pass the trigger.

3.2 High Momentum Trigger

High momentum muons are bent through smaller angles, therefore it is necessary to use Station 1 and Station 2 to form the trigger.

In this case the algorithm is similar, except that the X input is the output of the low momentum trigger and the Y input is a two of three majority of the planes in station 2.

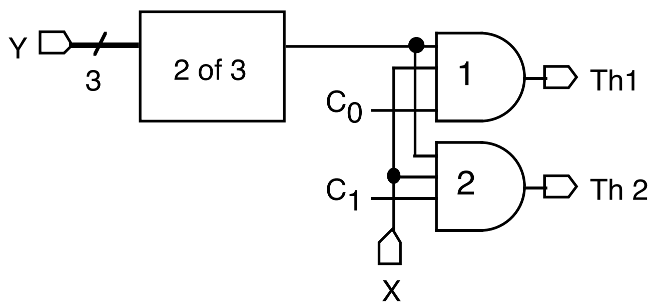

Figure 5 — High momentum threshold logic per intersection

3.3 Input Micro-Pipelining

Clearly, it is essential that the inputs to the co-incidence array arrive at or near the same time, despite differing cable lengths and time of flight in the detector.

One possible solution is instead of running the co-incidence array in combinatorial mode and sampling the output is to pipeline the operation at 160MHz and have a 'micro-pipeline' on each set of inputs. This is a shift register whose depth can be programmed, such that different inputs can have different depths to align the inputs in time.

To test this method a four stage micro-pipeline is implemented on each input. In the demonstrator the depth can be programmed only for each register (X_A, Y_A etc.), where the final system may require the depth to be individually controllable for each input. This register can be bypassed to allow strict combinatorial operation.

Single ended CMOS I/O pads had to be used because of pin limitations so the demonstrator will only be required to run at 80MHz.

4. The Demonstrator ASIC

The ASIC is intended to be flexible enough to demonstrate the high and low momentum muon triggers of RD27 and other coincidence array applications.

The functionality of the device is shown in Figure 6.

There are logically two X inputs and three Y inputs. To save pins on the device the X_B and Y_C inputs are multiplexed,

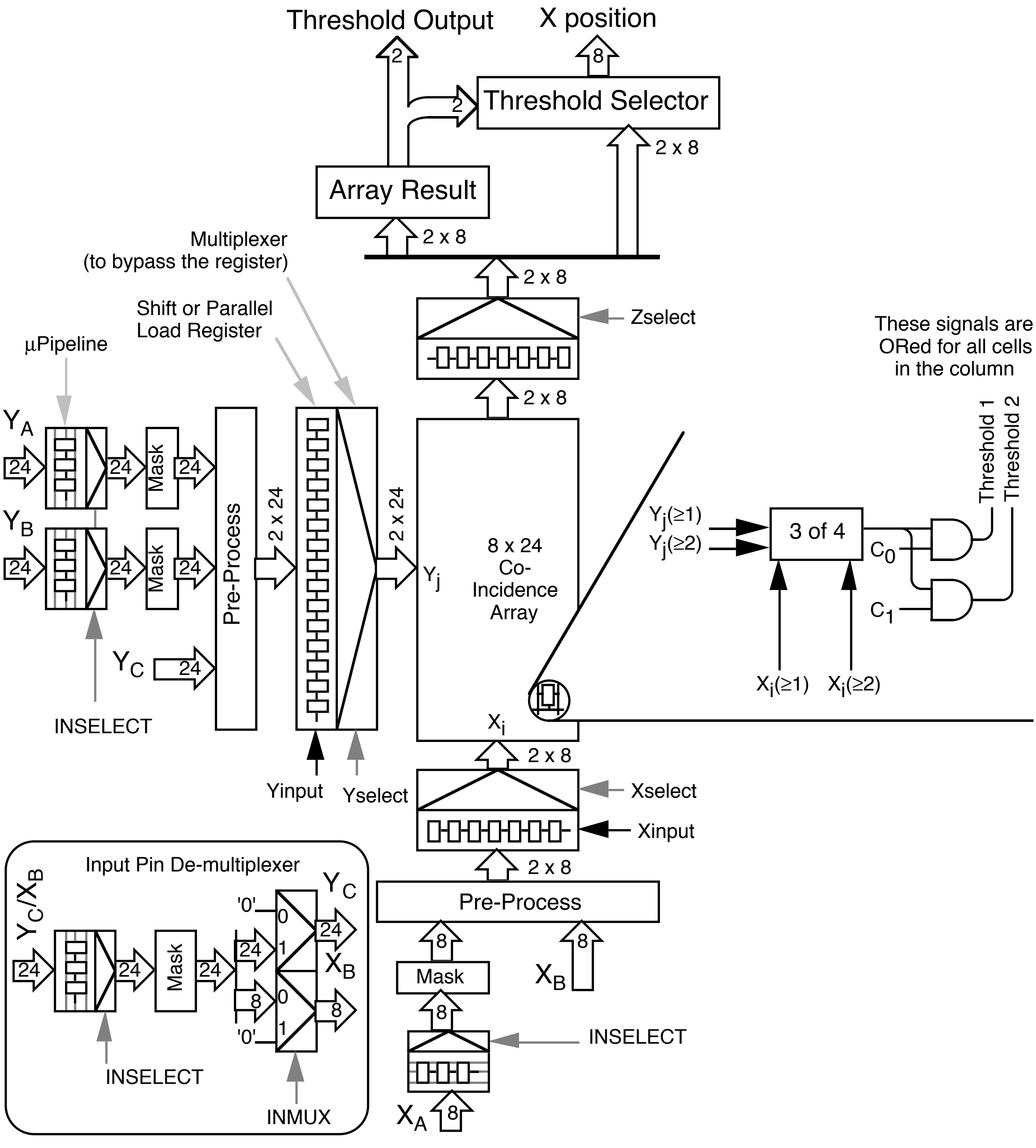

Figure 6 — The functionality of the demonstrator ASIC

only one can be used at a time — the other is set to zero.

4.1 The Input Registers

The X and Y input registers can be:

- bypassed for combinatorial operation, or
- allow the chip to operate in a programmable depth pipeline mode (µPipelining). The depth of the pipeline is controlled by another internal register.

In addition, because they can be configured as shift registers and loaded via the slow controls, they allow test vectors to be applied, in-situ, to the pre-processing block and Y_C/X_B demultiplexer.

4.2 Input Masking

Individual channels can be masked before pre-processing, for example if they are not connected, or if they fail.

This function is controlled by four mask registers, one for each of the input groups: X_A, Y_A, Y_B, Y_C/X_B. If bit 'i' in the X_A mask register is '1', for example, then that bit is masked before pre-processing, if the bit in the mask register is '0' the bit passes unchanged.

4.3 The Co-incidence Array Logic

As discussed in the section on low and high momentum triggers, the ASIC must make a 3 of 4 majority decision at each intersection of X and Y. In fact it is more efficient to split this function into two steps and perform some processing on the axis inputs (the 'pre-process' block) and therefore require less logic at each intersection.

The pre-process step takes the two individual chamber inputs (three in the case of the Y axis) and produces two signals:

≥1 Meaning that at least one high was seen on the inputs, and

≥2 Meaning that at least two highs were seen on the inputs.

The '3 of 4' logic at each intersection is then simpler than the full '3 of 4' majority function and saves logic overall. This split also has the advantage of allowing the same array logic to implement the low and high momentum trigger schemes. This will be shown later.

The dimensions of the array are eight in X by twenty-four in Y. The two different 'thresholds' are implemented using two coefficients per intersection (as shown in the top right of the figure) to encode:

Co-efficients (C_1C_0)	Meaning for the corresponding combination of X_i and Y_j
00	No threshold satisfied
01	Threshold 1 satisfied
10	Threshold 2 satisfied
11	Thresholds 1 & 2 satisfied

The two AND gates perform the two different threshold functions and then both are individually combined by ORing with all corresponding AND results in the column. The output of the array is therefore two sets of eight column results.

4.4 The Axis Registers

The X and Y inputs to the array can be driven from the pre-processing either directly or via axis registers. Similarly the output can be driven from the array or the Z axis register.

Note that it is possible to configure the device such that it is completely combinatorial in operation, completely registered, or a mixture of the two.

If the axis register is selected to drive the array, then a T-Zero function can be performed [2] by configuring the axis registers as shift registers and connecting their inputs to chamber outputs.

To implement the RD27 low and high momentum trigger schemes, the axes are configured as combinatorial input, through the parallel inputs. The parallel inputs are then driven as shown below:

Axis Input	Low momentum trigger	High momentum trigger
$X_A(7:0)$	SS1 plane 1	Low momentum output
$X_B(7:0)$	SS1 plane 2	'0'
$Y_A(23:0)$	SS1 plane 3	S2 plane 1
$Y_B(23:0)$	SS1 plane 4	S2 plane 2
$Y_C(23:0)$	'0'	S2 plane 3

4.5 The Array Result

The column results are combined into two single bit outputs, one flags whether threshold 1 is satisfied and the other whether threshold 2 is satisfied. It is possible to have both flags asserted simultaneously.

4.6 The X Position

The eight bit column result for the most significant satisfied threshold is then passed out of the chip by the threshold selector. This can be used by external logic to determine the X input(s) that fired the co-incidence array, or used as the input to cascaded co-incidence arrays.

In the case where both thresholds are satisfied simultaneously, threshold 2 is considered more ‘significant’ than threshold 1 and only the position of the threshold 2 co-incidence(s) is produced.

This allows the trigger to be tuned to ‘prefer’ either lower or higher momentum muons, depending on which threshold is programmed into threshold 2.

4.7 Slow Control

The slow control interface is used to:

- set and check the coincidence co-efficients of the array,
- test the functionality of the chip, both in the lab and in the system,
- capture the input to and output from the coincidence array in the axis registers and later read-out to monitor the operation of the system, and
- provide a source of test vectors from the axis or input registers to the coincidence array in the test lab and in-system.

5. Results

The design shown in Figure 6 was described in Verilog and checked by simulation. This Verilog description was used in system simulations, primarily to check the slow controls.

Schematic design of different parts of the ASIC was undertaken in parallel at INFN and RAL, combined and checked against the functional description.

The design was targeted at a 34k gate, 0.5μm CMOS Fujitsu gate array, of which it required 15k gates. The layout was performed by Fujitsu.

Timing simulations indicated that it would meet the specification. Results from fabricated devices, which were in agreement with the simulation, are:

Maximum clock frequency:	greater than 80MHz
Combinatorial delay from input to output:	7ns
Variation of delay as a function of input pattern	2ns

Two VME boards have been constructed to use the ASIC.

INFN have produced a board to implement the full trigger scheme, which will be tested in beam tests this year.

RAL have produced a general purpose test board, which can control the ASIC from VME and either:

- apply test vectors to the ASIC from memories downloaded via VME and capture the result in further memories, which can be inspected over VME, or
- inputs and outputs can be through front-panel connectors with the VME memories used to spy on the front-panel inputs and outputs.

This second module has been used to fully check all valid modes of operation of the device.

6. Conclusions

We have implemented a demonstrator ASIC with most of the functionality required to implement a muon trigger for the ATLAS experiment at LHC. Tests on fabricated devices have confirmed they perform within specification.

For a final muon trigger the following improvements would have to be made:

- the number of inputs to a single device needs to be increased to perhaps 32 in X by 64 in Y.
- the number of co-incidence arrays inside the device needs to be at least three.
- if pipelined operation is chosen over fully combinatorial operation then the device must work at 160MHz.
- the device could be implemented in the current technology because total doses in the muon detectors do not require rad-hard electronics.

Current sub-micron CMOS processes can already satisfy all final specifications. If fully pipelined operation is chosen pseudo-differential ECL I/O buffers can be used up to 320MHz frequencies.

7. References

[1] “A Fast Processor for Muon Triggering Using Resistive Plate Chambers”; E. Gennari, E. Petrolo and S. Veneziano; Conference Record of the 1991 IEEE Nuclear Science Symposium, Santa Fe, New Mexico, November 2-9, 1991; pp 579-583

[2] “Fast Pipelined Trigger Processors for the Forward Muon Chambers of the H1 Experiment on the Hadron Electron Ring Accelerator (HERA) at DESY, Hamburg”; J. Dowdell, M. French, S. Jaroslawski and M. Prydderch; Conference Record of the 1991 IEEE Nuclear Science Symposium, Santa Fe, New Mexico, November 2-9, 1991; pp 898-902

[3] “Use of GaAs circuits for first level muon triggering at LHC”; E. Petrolo and S. Veneziano; Nuclear Instruments and Methods for Physics Research A344 (1994) 194–198.

[4] “A coincidence array demonstrator ASIC for the RD27 muon trigger”; RD27 Note 30; J. Dowdell et. al.; 13th July 1994