

Implementation of the Sorting Schemes in a Programmable Logic

Matveev, M.

Rice University, Houston, TX 77005 USA
matveev@physics.rice.edu

Abstract

Experiments at the Large Hadron Collider (LHC) at CERN require a sophisticated multi-level trigger systems for data selection. At the Compact Muon Solenoid (CMS) experiment, the main task of Level 1 Trigger is to reduce the frequency of events from 40MHz down to 100kHz [1]. There are three major muon subsystems at the CMS: Cathode Strip Chambers (CSC), Drift Tubes (DT), and Resistive Plate Chambers (RPC). The Level 1 decision should be made about 3 μ s after an interaction in the collision area, so only simple algorithms implemented in hardware can be used at this level.

The current design requires that only the four best trigger candidates from each muon subsystem be passed to the Global Muon Trigger (GMT) System. It is considered that there is a sorter at the upper level of each muon subsystem trigger chain, which selects the four best candidates out of several tens of incoming and transmits them to the GMT. We propose a fast and flexible solution which would allow the implementation of such sorters for the various CMS muon subsystems. Four sorting schemes “3 objects out of 18”, “4 objects out of 8”, “4 objects out of 24” and “4 objects out of 36” are discussed. The first one is intended for the Muon Port Card [2] being designed for the CSC Trigger Electronics. The second scheme can be used for the RPC Sorting Processor [3]. The third scheme can be used for the DT Muon Sorter [4]. The fourth scheme is targeted to CSC Muon Sorter [2]. Designs based on Altera 20KE [5] Programmable Logic Devices (PLD) and results of simulation are presented.

I. DESIGN ASSUMPTIONS

We assume that sorting is based on the value of input patterns: higher ranks correspond to “better” patterns for the purpose of sorting. All

schemes are targeted to a single chip implementation in order to reduce the overall sorting time. Each sorter chip receives 8, 18, 24 or 36 input patterns and outputs an addresses of three or four best patterns. The addresses of incoming patterns are assigned inside the sorter chip for the first, third and fourth schemes and obtained from the outer logic for the second scheme. If sorting objects contain more bits than the patterns that are used for the sorting, they should be temporarily pipelined in the external registers, and the addresses of the best patterns can be used for their further multiplexing onto outputs of the sorter board. This pipelining as well as a sorting logic can be done in a single chip if it has a sufficient number of input and output pins. We have implemented our sorting schemes for the 7- and 8-bit patterns, which is typical for the CMS muon trigger systems.

We assume that all patterns come to the sorter device in parallel being synchronized with its master clock. The input and output latches for all patterns and their addresses provide a reliable synchronous operation and predictable timing. All our designs are targeted to applications at the LHC experiments, where the main operating clock frequency is 40.08Mhz, so our goal was to achieve a registered performance of 40+ MHz for all sorting schemes with the minimal latency.

Finally we assume that all inputs are not pre-selected (or ranked), but all the output patterns should be ranked, or present on the outputs of the sorter chip in descending order.

II. TARGETED DEVICES AND DESIGN SOFTWARE

The main requirements for the targeted devices are: (1) they should be fast and flexible enough to be able to implement sorting schemes in a few clock cycles of the 40MHz clock and (2) provide an adequate number of input and output pins for a single-chip implementation. One approach is to develop and use an ASICs.

Another approach assumes utilization of PLD, or Field Programmable Gate Arrays (FPGA). So far ASICs have an inherent speed advantage over a PLD or FPGA due to their dedicated metal routing. Soft routing of PLD/FPGA makes them run slower than an ASIC. But the gap between ASIC and PLD/FPGA performance is decreasing due to advances in technology and chip architecture. The important advantages of using PLD/FPGA are the low nonrecurring engineering fees, reprogrammable features as well as a shorter design cycle. Due to these factors we focus on using of the PLD/FPGA devices for our designs. The Xilinx Virtex-E [6] and Altera 20KE [5] families are the most attractive for our needs due to their high performance and great flexibility. Our designs are targeted to Altera 20KE devices. We have used an Altera Quartus version 2000.05 PLD design software running on a PC.

III. SORTING METHOD

We perform all necessary comparisons in parallel at the beginning of the sorting process. In case of (n) input patterns the total number of comparisons between all patterns is $N=n(n-1)/2$. These numbers N are 28, 153, 276 and 630 for $n=8, 18, 24$ and 36 respectively. N results of all comparisons would allow us to obtain (4n) combinatorial signals for the address multiplexing onto chip outputs. In case of “4 objects out of 8” scheme the four best patterns along with their addresses are transmitted to the chip outputs. Simplified block diagram of the sorting scheme for $n=18$ is shown on Figure 1. For all our designs, if a few equal patterns are selected, the pattern with the largest address receives the highest rank.

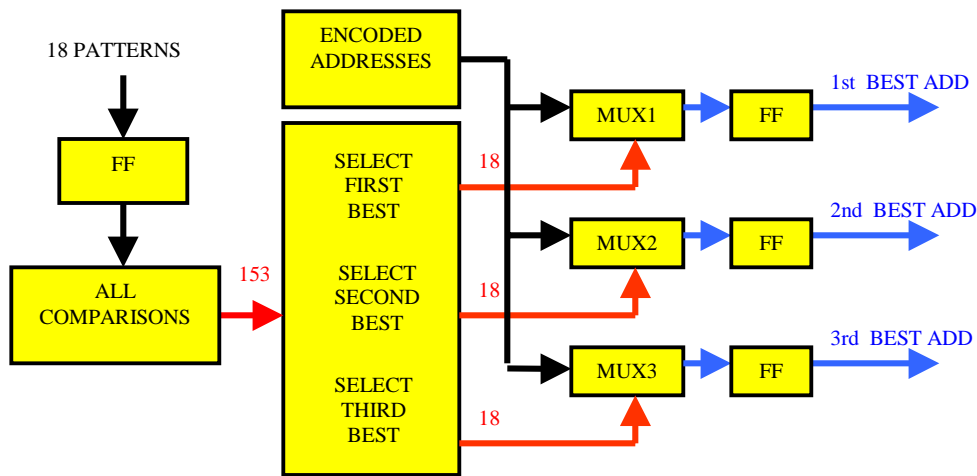


Figure 1: Sorting Scheme “3 objects out of 18”

IV. RESULTS OF SIMULATION

The results of simulation are shown in Table 1. All timing parameters are obtained for the fastest (-1 speed grade) devices. The sorting schemes “4 out of 24” and “4 out of 36” are implemented in two and four clock cycles respectively and both require an intermediate flip-flops (FF).

We assume that the actual sorting latency (in clock cycles or nanoseconds) means the time

interval between the latching of the input patterns into sorter chip and moment when the selected addresses and in some cases also the best patterns are available for latching in the external register outside the sorter chip.

Additional time is needed for the output data latching in the external registers to assure reliable synchronous operation. This time should be added to the internal latency parameter in Table 1 in order to obtain an actual latency of each sorting scheme. According to the timing model of the fastest devices in 20KE family [5], the maximum clock-to-output delay T_{outco} with

the global clock at the input/output register is less than 5 ns. The propagation delay between the sorter PLD and destination device as well as setup time for the destination device should also be taken into account. ClockShift circuitry [5] available in 20KE devices can provide an

output clock signal shifted with respect to the input clock. Particularly, the output clock with phase shift of 180 degrees (half of period of the 40MHz clock, or 12.5 ns) can be used for the output data latching into the external register.

Table 1: Results of Simulation

Sorting Scheme	3 out of 18	4 out of 8	4 out of 24	4 out of 36
Number of inputs (except clock)	$8*18=144$	$(8+8)*8=128$	$7*24=168$	$7*36=252$
Number of outputs	$5*3=15$	$(8+8)*4=64$	$5*4=20$	$6*4=24$
Altera PLD Device	EP20K100 EFC324-1	EP20K60 EFC324-1	EP20K200 EFC484-1	EP20K400 EFC672-1
Internal latency, clock cycles of 40MHz	1	1	2	4
Maximum registered performance, MHz	43.62	68.25	41.94	52.95
Internal latency, nanoseconds	24	15	48	76
Number of logic cells used and total available inside the PLD	3616/4160 (86%)	1422/2560 (55%)	7100/8320 (85%)	11383/16640 (68%)

V. CONCLUSION

We have proposed a design methodology for the implementation of the fast sorting schemes. Four schemes “3 objects out of 18”, “4 objects out of 8”, “4 objects out of 24” and “4 objects out of 36” are implemented and simulated. The internal sorting latency is as low as 24, 15, 48 and 76 ns for the first, second, third and fourth schemes respectively. The total latency should also include a delay needed for the latching of the sorting result in the external registers. All our designs are based on the 20KE family of Altera PLD and can be used at the various levels of trigger systems at the CMS and other experiments.

VI. REFERENCES

- [1] The Compact Muon Solenoid. Technical Proposal. CERN/LHCC 94-38. LHCC/P1. 15 December 1994. Available at: <http://cmsinfo.cern.ch/TP/TP.html>
- [2] D.Acosta et al. The Track-Finding Processor for the Level-1 Trigger of the CMS Muon System. CMS Note 1999/060. December 8, 1999. Available: ftp://cmsdoc.cern.ch/documents/99/note99_060.pdf
- [3] G.De Robertis, A.Ranieri, I.M.Kudla, G.Wrochna. The Sorting Processor Project. CMS TN/95-028. 6 March, 1995. Available at: http://cmsdoc.cern.ch/tn/95/tn95_028.pdf
- [4] http://cmsdoc.cern.ch/~marcodv/tridas_990519.pdf

- [5] Altera Device Data Book 1999. APEX 20K Programmable Logic Device Family.
- [6] Xilinx Programmable Logic Data Book 2000.