A Pipeline of Associative Memory Boards for Track Finding

Alberto Annovi, Maria Grazia Bagliesi, Antonio Bardi, Roberto Carosi, Mauro Dell'Orso, Paola Giannetti, Giusseppe Iannaccone, Fabio Morsani, Marco Pietri, and Graziano Varotto

***Abstract*—We present a pipeline of associative memory boards for track finding, which satisfies the requirements of level two triggers of the next large hadron collider experiments. With respect to previous realizations, the pipelined architecture warrants full scalability of the memory bank, increased bandwidth (by one order of magnitude), and increased number of detector layers (by a factor of two). Each associative memory board consists of four smaller boards, each containing 32 programmable associative memory chips, implemented with a low-cost commercial field-programmable gate array (FPGA). FPGA programming has been optimized for maximum efficiency in terms of pattern density, while printed circuitboard design has been optimized in terms of modularity and FPGA chip density. A complete associative memory board has been successfully tested at 40 MHz; it can contain 7.2×10^3 particle trajectories.**

***Index Terms*—Associative memories, data acquisition, field-programmable gate arrays, parallel processing, particle tracking, pattern matching, pattern recognition, pipeline processing, trigger.**

I. Introduction

DEDICATED hardware systems, based on the use of an associative memory (AM) [1], [2], have been proposed for track recognition in high-energy physics experiments. The AM must be able to store all trajectories of interest and extract those compatible with a given event. A trajectory is compatible with an event if all (or most) detector channels crossed by that trajectory have fired in that event.

Such a task can be very conveniently parallelized. A modular architecture can be envisioned in which each module contains a "pattern." The pattern includes both the memory required for storing a single trajectory and the logic needed to compare the coordinates of all fired detector channels with those associated to the stored trajectory. Each module must receive as inputs the complete configuration of fired detector channels of each event. A programmable AM based on a modular architecture has already been realized using full custom ICs [1] and field-programmable gate arrays (FPGAs) [2].

Manuscript received October 31, 2000; revised January 22, 2001.

A. Annovi, M. G. Bagliesi, R. Carosi, P. Giannetti, F. Morsani, M. Pietri, and G. Varotto are with the Istituto Nazionale di Fisica Nucleare, Pisa, Italy (e-mail: alberto.annovi@pi.infn.it; mariagrazia.bagliesi@pi.infn.it; roberto.carosi@pi.infn.it; paola.giannetti@pi.infn.it; fabio.morsani@pi.infn.it; marco.pietri@pi.infn.it; graziano.varotto@pi.infn.it).

A. Bardi and M. Dell'Orso are with the Dipartimento di Fisica, Università di Pisa, Pisa 56100, Italy (e-mail: antonio.bardi@pi.infn.it; mauro.dellorso@pi.infn.it).

G. Iannaccone is with the Dipartimento di Ingegneria dell'Informazione, Università di Pisa, Pisa I-56126, Italy (e-mail: giuseppe.iannaccone@iet.unipi.it).

Publisher Item Identifier S 0018-9499(01)04898-5.

It is important to point out that in order to contain the pattern bank within an acceptable size, the AM operates at a coarser resolution than the actual detector. This is usually done by clustering single contiguous detector channels into larger "superbins." In the following, we will call "hits" the addresses of fired "superbins" of each layer and "roads" the coarse resolution trajectories (each road corresponds to an array of superbin addresses—one per detector layer).

II. Pipeline of AM Boards

We have developed an associative memory based on a pipeline of AM boards, which represents a substantial improvement with respect to previous realizations [1]–[4]. First, we have greatly increased the input bandwidth of the AM: hits from the overall detector are sent to the first AM board on six parallel 18-bit buses. Hits are handled and distributed to the different buses by a set of purposely developed boards, the data organizer [5]. The total bandwidth reaches 4 Gbit/s, i.e., it is an order of magnitude larger than that previously obtained [3], [4] with a single data bus.

The AM boards are then assembled into a pipeline: data (hits and roads) exiting the first board are fed into the second and so on until the end of the pipeline. In such a way, the AM is fully scalable: boards can be simply added to increase the pattern bank size, with the only drawback being increased data latency. On the other hand, if all boards were connected to the same input buses, the maximum number of boards would be limited by the bus fanout. Different AM boards can work on different events.

In addition, we have almost doubled the number of considered detector layers with respect to [3] and [4]. With all the improvements described above, the AM can meet the tight requirements for use in the level-two trigger of new large hadron collider (LHC) experiments [6].

III. AM Board

The AM board has a modular structure consisting of four smaller boards, the local associative memory banks (LAMB). Each LAMB contains 32 programmable associative memory (PAM) chips, 16 per face. The structure of the AM board is sketched in Fig. 1. The structure of the LAMB is also shown. The PAM chips come in PQ208 packages and contain the stored patterns with the readout logic.

A *start event* and an *end event* word separate hits and roads belonging to different events. Different data organizer boards

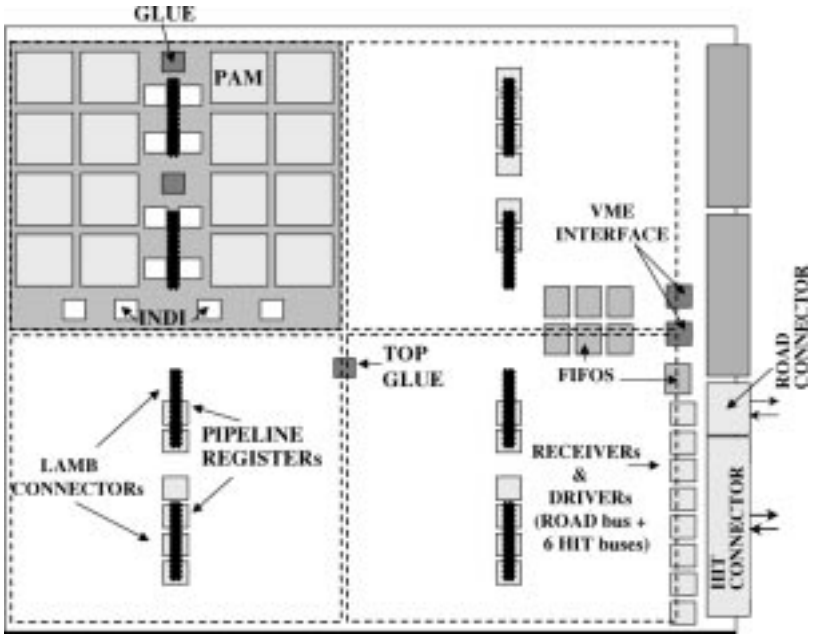

Fig. 1. AM board layout. Dashed lines show areas occupied by LAMBs. In the top left corner, the shaded area shows details of one LAMB.

process asynchronously the events that have to be synchronized at the input of each AM board. Each board input is provided with deep first in, first outs (FIFOs) for this goal. If, occasionally, an FIFO becomes "almost full," a HOLD signal is sent to the upstream board, which suspends the data flow until more FIFO locations become available. The "almost full" threshold is set to give the upstream board plenty of reaction time.

When an AM board starts to process an event, the hits are popped in parallel from the six-hit input FIFOs. Popped hits are simultaneously sent to the four LAMBs and to the output registers that drive the connections to the downstream board.

Data from different streams are checked for consistency: upon detection of different event sequences, a severe error is signaled and the whole system needs to be synchronized again. As soon as hits are downloaded into a LAMB, locally matched roads set the request to be read out from the LAMB (output ready). When the end-event word is received on a hit stream, no more words are popped from that FIFO until the end event word is received on all hit streams. Once the event is completely read out from the hit FIFOs, the LAMBs make the last matched roads available within a few clock cycles. Roads must cross the whole AM pipeline to go back to the data organizer. In addition to the LAMBs, another source of matched roads is the road FIFO, which receives roads on the road input bus from the upstream board. All matched roads belonging to the same event are multiplexed through the road output register on the same output bus, with priority to the LAMB roads.

Hits and roads flow from one board to the next on a custom backplane. For the six-hit buses and the road bus, 132 and 30 lines are necessary, respectively. Since each line enters and exits the board, these links require 324 connector pins per board. We use eight-column hard metric connectors. A small horizontal section of a pair of adjacent connectors is shown in Fig. 2 with some sample connections: inputs and outputs pins are organized into separate rows. The input pin and the output pin corresponding to the same signal bit are on adjacent rows at the same column. This configuration allows short and regular circuit paths on the custom backplane.

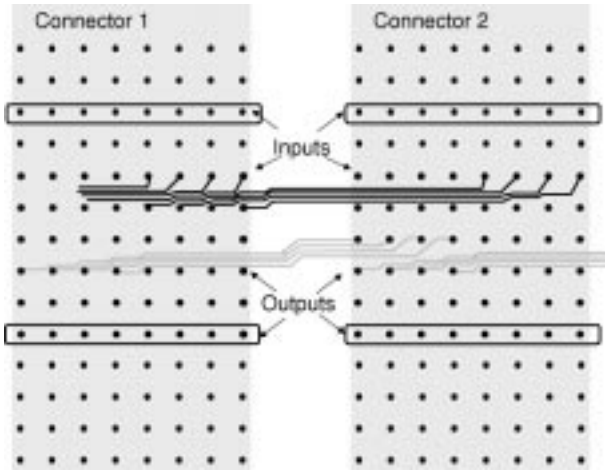

Fig. 2. Horizontal section of the custom backplane for interconnecting adjacent AM boards. For clarity, only a few metal lines are shown between a pair of adjacent eight-column connectors. Black and gray lines represent, respectively, top and bottom side connections. Actually, line patterns on both sides are replicated every two rows.

IV. LAMB

The hit buses are fed in parallel to the four LAMBs and distributed to the 32 PAM chips on the LAMB through 12 fanout chips called input distributors (INDIs); four PAM chips are driven by each INDI. With regard to reading out the matched roads, the PAM chips on each LAMB are connected into four eight-chip pipelines. This pipeline, consisting of 4 chips on the top side and 4 chips on the bottom side, as shown in Fig. 3, has the same structure as the whole AM board pipeline. Each PAM chip receives an input road bus from the upstream chip and multiplexes the input roads with the internally matched roads on a single output bus to the downstream chip. Signals

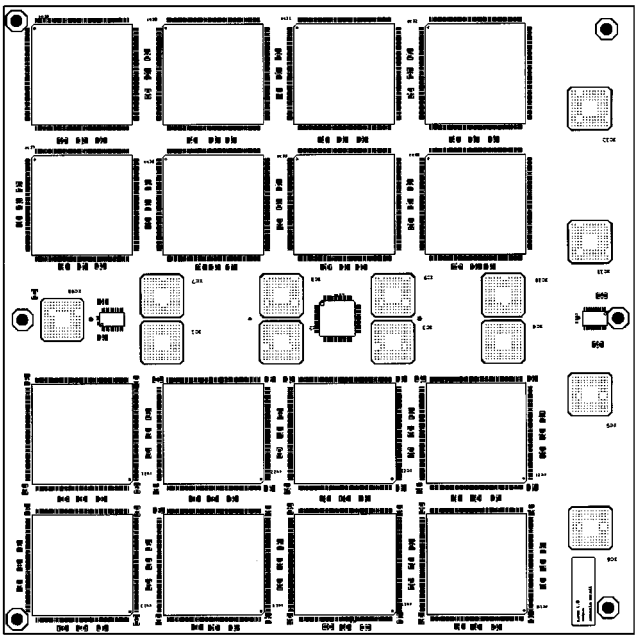

Fig. 3. Top view of a LAMB board. Sixteen PAM chip locations are arranged into a 4 by 4 array. On the bottom side, not visible in the figure, 16 more chips are arranged in the same positions. Each array column, with the corresponding column on the other board side, constitutes an eight-stage pipeline for reading out matching roads. The small squares full of tiny pads are the locations for chip-scale packages.

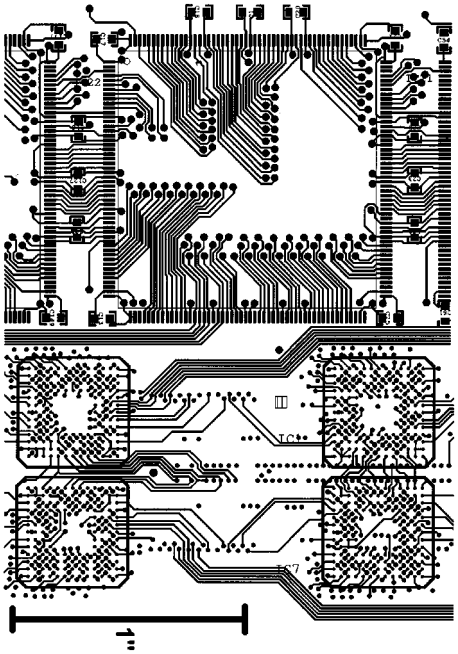

Fig. 4. A closer view of the LAMB board PCB.

propagate from chip to chip on board surfaces through very short pin-to-pin lines using no via. The outputs from the four PAM pipelines are then multiplexed into a single road bus by a purposely designed GLUE chip on each LAMB.

The LAMB board has been designed, optimized, simulated, placed, and routed with the Cadence software. Its realization has represented a significant technological challenge, due to the high density of chips allocated on both sides and to the use of the advanced chip-scale packages (CSPs) for the 12 INDI chips and the GLUE chip, as shown in Fig. 3.

The CSP features a ball grid array with a 0.8-mm pitch, connecting to a silicon die that is only 20% smaller than the package size. Such small pitch and large board density push the printed circuit board (PCB) geometry to the edge of available technology, with 20 mils blind vias, lines with minimum width, and spacing of 5 mils in a 1.6-mm-thick board with eight routing layers.

Fig. 4 shows a portion of the LAMB PCB top layer. Next to the big housing for a PAM chip on the left, four locations for INDI chips are shown. Note the small dimensions compared to the inch ruler.

Patterns are downloaded by programming the FPGAs through the Versa Module Eurocard (VME)-controlled JTAG port. Chains of four PAMs each are downloaded in parallel to reduce programming time. The VME 32-bit-wide data transfer allows us to program 32 chains in parallel for a total of four LAMBs. Downloading time is measured to be a few seconds.

V. PAM Chips

For the implementation of PAM chips, we have chosen a commercial low-cost FPGA. The FPGA approach allows a large degree of flexibility in the prototyping and testing phases of the project and an easy upgrade of the system when new generations of FPGA will be delivered to the market. Indeed, newer pin-compatible FPGAs can replace old chips in the same PCB and can be very conveniently configured via the high-level hardware description of the logic.

While the associative memory receives hits from 12 detector layers on six 18-bit buses, we have implemented—as a first prototype—a simpler PAM chip receiving hits from eight layers on four 18-bit buses. However, the architecture is rigorously modular and therefore can be easily scaled up to a larger number of layers, when a denser FPGA generation will be available.

Each PAM chip is dedicated to tracks from a single detector sector. It is convenient to order patterns and divide them into classes that correspond to detector sectors (adjacent sectors are overlapping, so that tracks that cross sector boundaries are not lost). The most significant bits are common to all patterns belonging to the same class and can be checked only once per chip. Therefore, a masking logic shall enable input registers only for hits with the appropriate sector address.

Let us call the four buses HITi, ($i = 1, \ldots, 4$). The five most significant bits (HITi[17 : 13]) represent the detector sector. HITi[12] is the channel select (CS) bit: if HITi[12] $= 0$, the hit belongs to layer i, while if HITi[12] $= 1$, the hit belongs to layer $i+4$. The 12 least significant bits (HITi[11 : 0]) represent the hit address within the detector sector.

The FPGA chip used (Xilinx Spartan 0.35 μm process) contains an array of 28×28 configurable logic blocks (CLBs), each containing two 4-bit lookup tables (LUTs), a 3-bit LUT, two flip-flops with three-state buffers, and carry logic. The details of the Spartan family can be found in documentation from Xilinx [7]. Each PAM chip contains several identical pattern modules, each consisting of the memory for storing a single road, the matching logic, the road address, and the readout logic.

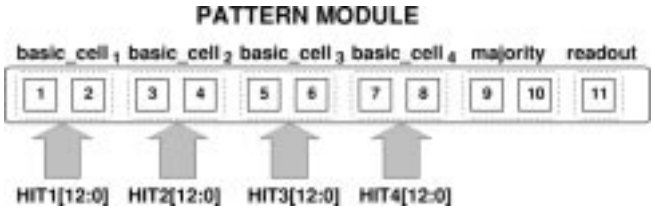

Fig. 5. Structure of the pattern module consisting of 11 CLBs.

A pattern module consists of 11 CLBs, as shown in Fig. 5. There are four basic_cell_i modules ($i = 1, \ldots, 4$), each verifying the hits on the ith bus and occupying two CLBs; a majority module (two CLBs) checking how many hits of the road have been matched; and a readout module (one CLB) with the logic required for sending the matched road address onto the address bus.

The basic_cell_1 module is shown in Fig. 6. Each of the four 4-bit LUTs receives as inputs three bits of the hit address and the CS signal. If CS $= 0(1)$, the output is 1 only if the three bits match the hit address of layer 1 (5). The 3-bit LUTs AND0 and AND1 implement a controlled four-input AND: the output CLKOUT is high if the four partial comparisons $c0$–$c3$ are high (i.e., if the whole hit address has been matched) and CLKIN (the global clock signal) goes low. CLKIN enables the CLKOUT signal with a delay of about half a cycle, preventing the occurrence of glitches on CLKOUT that could induce errors, since CLKOUT is used to trigger the two flip-flops MATCH1 and MATCH0. MATCH1 is enabled by CS (HIT1[12]) and is therefore set to one if the hit on layer 5 is matched; MATCH0 is enabled by not (CS) and is set to one if the hit on layer 1 is matched.

The four basic_cell modules have as output an 8-bit signal MATCH[7 : 0] where each bit MATCH[i] $= 1$ if the hit on the ith layer has been matched. Since the efficiency of each detector channel is smaller than one, we can assume that a pattern is matched if at least six out of eight hits have been matched. The majority module sketched in Fig. 7 has this precise function: it sets to one GENMATCH if at least six out of the eight bits of MATCH[7 : 0] are one.

When a number of patterns are matched, the PAM chip must send onto the output bus a sequence of all pattern addresses. The priority among different patterns can be handled by using the carry logic available on the Spartan chip to realize a chain of OR gates, shown in Fig. 8, which we call the carry chain. It receives as inputs all the GENMATCH_p bits (where p refers to the pth pattern). As can be seen, when all the GENMATCH bits are zero, COUT is zero. When pattern p is matched, all OR gates from the pth to the end switch to one. If afterwards pattern j is matched, nothing happens if $j > p$, while if $j < p$ all OR gates from the jth to the pth switch to one. The matched pattern with the smallest index p, i.e., closest to the start of the carry chain, has the highest priority: it can be easily identified since it is the only one for which CIN_$p = 0$ and GENMATCH_$p = 1$. As soon as its address is sent onto the output bus, GENMATCH_p is reset to zero, and the next pattern with the smallest index will be sent to output.

In an FPGA chip containing 28×28 CLBs, 56 pattern modules can be stored. With this implementation, each AM board contains $56 \times 32 \times 4 = 7168$ patterns. Pattern modules are organized in the FPGA chip, as shown in Fig. 9. In such a way, vertical long-lines traversing the whole chip are used for the hit buses. Input registers will be positioned in the ring of input/output blocks. The carry chain is visible in Fig. 9 in the narrow region between the two arrays of pattern modules. The HIT5 and HIT6 buses are not used in this specific implementation of the chip.

The road addresses are stored in the flip-flops available in the CLBs and not used in the pattern module logic (each CLB contains two flip-flops, but only one is used in the basic_cell modules). Such flip-flops are set to one or zero when the FPGA is programmed. The address bus is implemented with horizontal long-lines, the only ones that in the chip can be connected to the outputs of the three-state buffers.

The readout module, shown in Fig. 10, has the following functions: it checks whether the pattern has the highest priority, and, if this is the case, sends its address onto the output bus and resets the matching logic. In particular, the MATCH-FF register updates the GENMATCH bit with a delay of one clock cycle. The enable (EN) LUT sets BUFCON to one if SEL $= 1$, i.e., the PAM chip has been selected for sending the addresses, if LAM $= 1$ and CIN $= 0$, i.e., the pattern has the highest priority. When BUFCON $= 1$, the three-state buffers are enabled to send the pattern address onto the address bus, and the flip-flops in each basic_cell module are reset. The RESET signal in Fig. 10 is the global chip reset.

Chips belonging to the same pipeline of the LAMB board are connected in a single chain as shown in Fig. 11. Each chip collects patterns from upstream chips raising the SEL_OUT signal. It multiplexes the received roads with those internally matched, and generates the OR_OUT to signal the presence of patterns to be read out.

The PAM chips have been logically designed with the VERILOG hardware description language and have been mapped onto FPGAs with Synopsis. The logical structure of the PAM chip has been optimized for the FPGA family considered, and particular care has been taken in order to obtain an efficient use on FPGA logic. In such a way, we have been able to use 95% of the total number of CLBs available on the chip, a percentage much larger than that usually obtained with FPGAs.

VI. Conclusions

We have designed a pipeline of associative memory boards for track finding. It works at a clock frequency of 40 MHz. Our design is rigorously modular and can be easily scaled up to a very large number of roads. It can therefore satisfy the requirements of level-two triggers of the next experiments at the LHC.

PCB design has been optimized in terms of modularity and number of deployed FPGA chips. In addition, FPGA programming has been carefully tailored to the architecture of the FPGA family considered, in order to optimize the use of the available logic blocks and achieve the maximum allowed pattern density. As a result, the proposed PAM board can handle 7168 roads.

As an additional advantage, the use of commercial FPGA chips will allow us to substitute the currently used FPGA chips with pin-compatible denser chips from the same family, as soon

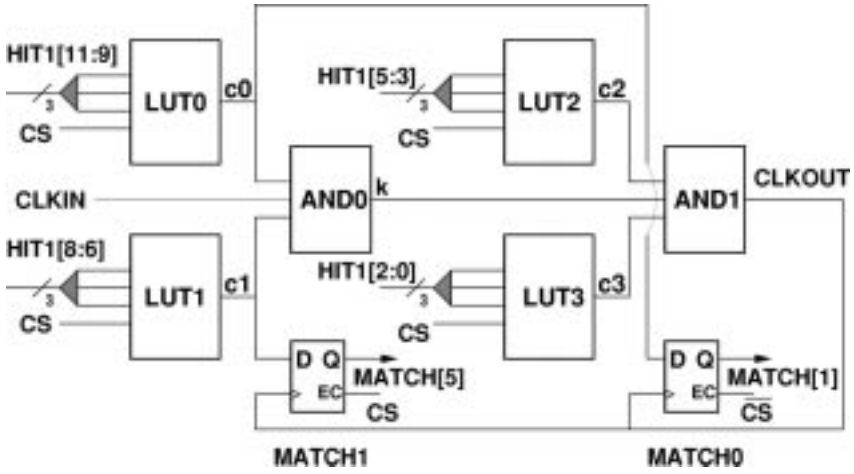

Fig. 6. Structure of the module basic_cell_1, consisting of two CLBs. MATCH0 is set to one if the hit on layer 1 is matched; MATCH1 is set to one if the hit on layer 5 is matched.

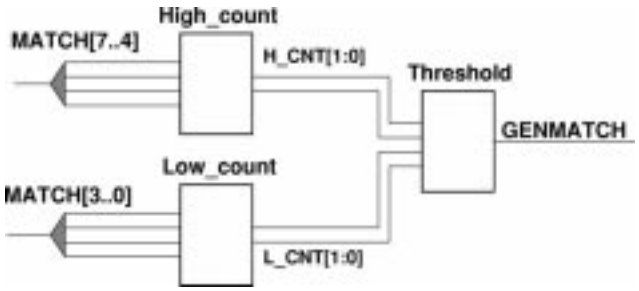

Fig. 7. Majority module. GENMATCH is set to one if at least six out of eight bits of MATCH[7 : 0] are one.

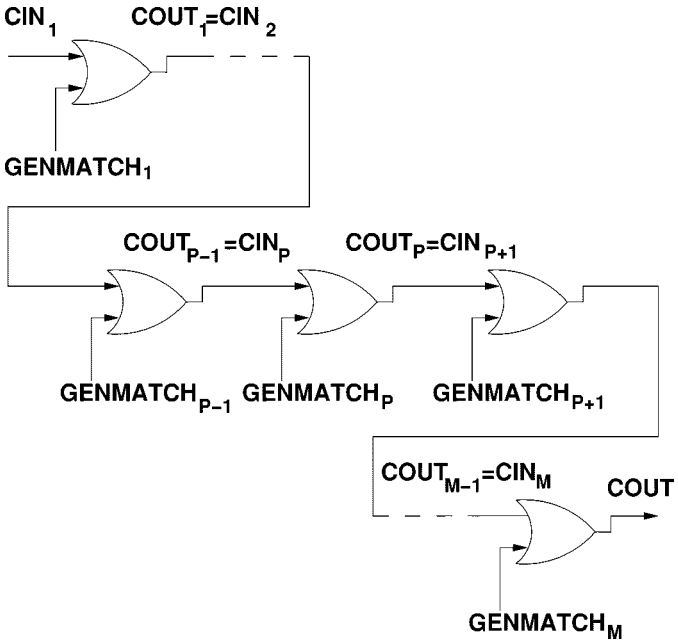

Fig. 8. Carry chain. The highest priority address is the one corresponding to pattern p if GENMATCH_$p = 1$ and CIN_$p = 0$.

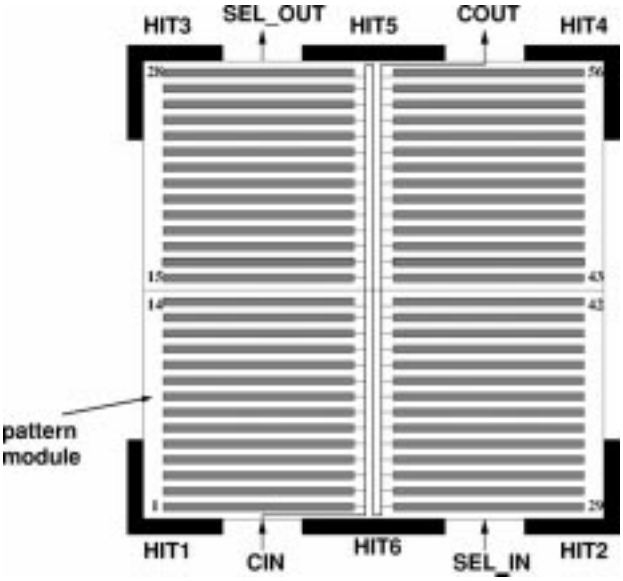

Fig. 9. 56 pattern modules can be stored in a single chip. The carry chain runs vertically in the middle of the chip.

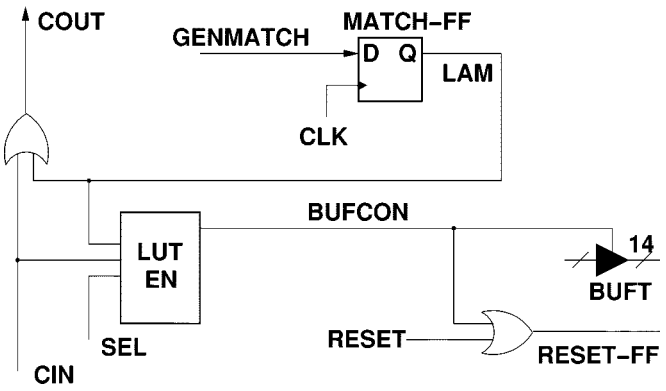

Fig. 10. Readout module. If CIN $= 0$, LAM $= 1$, and SEL $= 1$, the three-state buffers are enabled to send the road address onto the address bus.

as they will be delivered to the market. In such a way, little design upgrade is needed to keep the pace of improvements in semiconductor technology.

With no upgrade of PCB design, an improvement of about one order of magnitude in pattern density can be obtained by simply replacing the FPGA PAM chips with purposely designed pin-compatible application-specific ICs. Estimates and projections of the number of AM boards required, both with the FPGA and

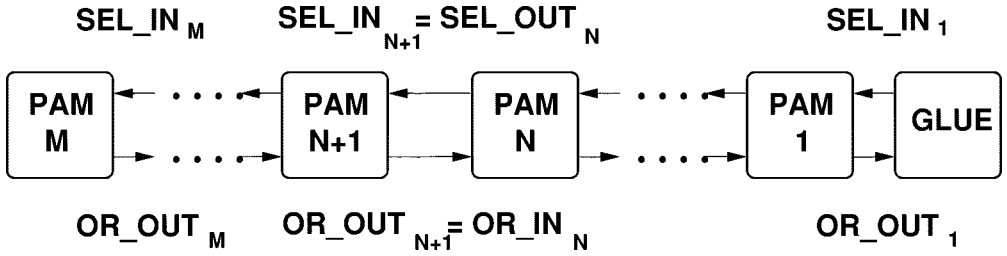

Fig. 11. Pipeline of PAM chips.

with the ASIC approach, are discussed in [6] in relation to the expected progress of CMOS technology.

REFERENCES

[1] S. R. Amendolia, S. Galeotti, F. Morsani, D. Passuello, L. Ristori, and G. Sciacca *et al.*, "The AM chip: A full-custom CMOS VLSI associative memory for pattern recognition," *IEEE Trans. Nucl. Sci.*, vol. 39, pp. 795–797, Aug. 1992.

[2] A. Bardi, S. Belfort, A. Cerri, M. Dell'Orso, S. Donati, and S. Galeotti *et al.*, "A programmable associative memory for track finding," *Nucl. Instrum. Meth.*, vol. A413, no. 2–3, pp. 367–373, Aug. 1998.

[3] A. Bardi, S. Belforte, J. Berryhill, A. Cerri, A. G. Clark, and R. Culbertson *et al.*, "SVT: An online silicon vertex tracker for the CDF upgrade," *Nucl. Instrum. Meth.*, vol. A409, pp. 658–661, May 1998.

[4] A. Bardi, S. Belforte, M. Dell'Orso, S. Galeotti, P. Giannetti, and E. Meschi *et al.*, "A prototype of programmable associative memory for track finding," *IEEE Trans. Nucl. Sci.*, vol. 46, pp. 940–946, Aug. 1999.

[5] A. Annovi, M. G. Bagliesi, A. Bardi, R. Carosi, M. Dell'Orso, and P. Giannetti *et al.*, "The data organizer: A high traffic node for tracking detector data," IEEE Trans. Nucl. Sci., Aug. 2001, to be published.

[6] A. Annovi, M. G. Bagliesi, A. Bardi, R. Carosi, M. Dell'Orso, and P. Giannetti *et al.*, "The fast tracker processor for hadronic collider triggers," *IEEE Trans. Nucl. Sci.*, pp. 575–580, June 2001.

[7] Zilinx, The Programmable Logic Company. (2001, Feb.) Xilinx Data Book 2000. Xilinx, San Jose, CA. [Online]. Available: http://www.xilinx.com/partinfo/databook.htm