A New Way to Implement High Performance Pattern Recognition Associative Memory in Modern FPGAs

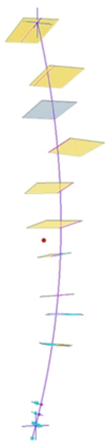
Jamieson Olsen[1], Jim Hoff[1], Tiehui Ted Liu[1], Jin-Yuan Wu[1], Zijun Xu[2]

[1] Fermi National Accelerator Laboratory, Batavia, Illinois, USA
[2] Peking University, Beijing, CHINA

Abstract

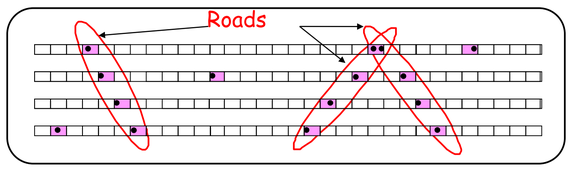
Pattern recognition associative memory (PRAM) devices are parallel processing engines which are used to tackle the complex combinatorics of track finding algorithms. Typically PRAMs have been implemented as an ASIC due to the high pattern density and performance requirements. FPGA-based PRAM designs usually cannot achieve high pattern density, however, they would allow for quick iterations, making an ideal hardware platform for designing and evaluating new PRAM features before committing to silicon. For example, modeling in FPGAs can bring the system interface to maturity much sooner and minimize the ASIC design cycles. We present our FPGA-based PRAM design that is optimized for modern FPGA architectures, and introduce a new mezzanine card which supports both the latest ASIC and FPGA PRAM designs as part of our tracking trigger R&D program.

Pattern Recognition Associative Memory (PRAM)



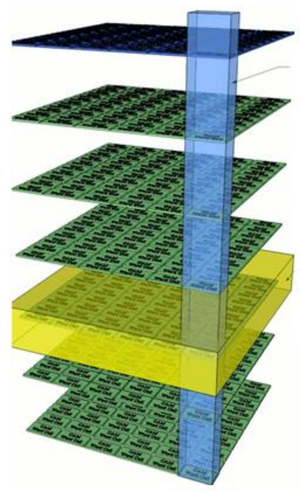
Left: a particle passes through multiple layers of silicon detector modules. These silicon modules transmit the coordinates of each hit to downstream trigger electronics.

Below: a pattern is pre-defined set of coarse resolution hits which span multiple detector layers. This pattern recognition approach has been used successfully at CDF/SVT.



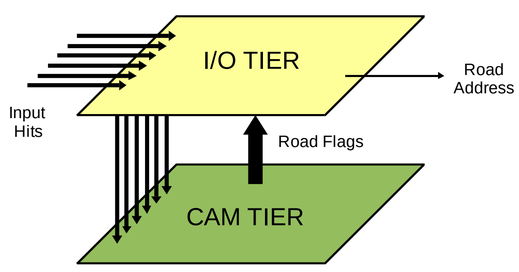
A PRAM is based on Content-Addressable Memory (CAM) which has been optimized to function as a massively parallel pattern recognition engine. PRAM allows for tertiary bits ('X') and flexible majority logic to support missing layer hits.

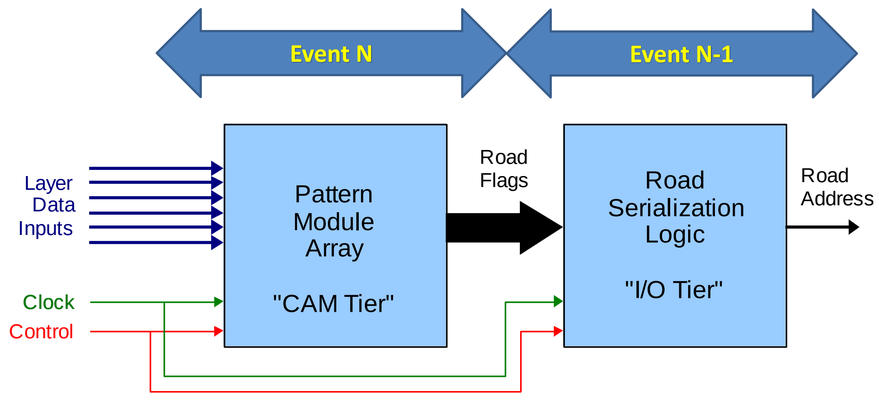
Vertically Integrated PRAM (VIPRAM) is a 3D ASIC being developed at Fermilab. In VIPRAM the detector layer pattern matching logic is partitioned into silicon tiers. In this configuration a pattern is a vertical tube (shown in blue). The entire top tier silicon is dedicated for data input, majority logic and serialization/readout logic. A simpler two tier design is also in development, in which the top tier is for data IO and the L1 tracking trigger system interface, while the bottom tier contains all the patterns. It is this two tier ASIC design that we have implemented in FPGA.



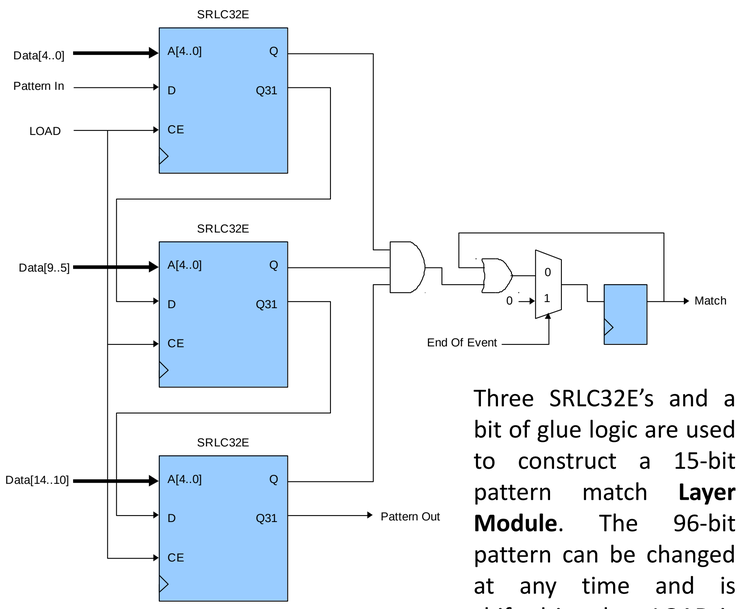
PRAM Architecture and FPGA Implementation

In our two tier PRAM ASIC design the pattern matching logic (e.g. the "CAM Tier") is separated from the road serialization logic ("I/O Tier") as shown to the right. These two tiers are fully pipelined for high-throughput, minimal dead-time operation. In our FPGA PRAM implementation these two tiers correspond directly to the top level firmware blocks, shown below:

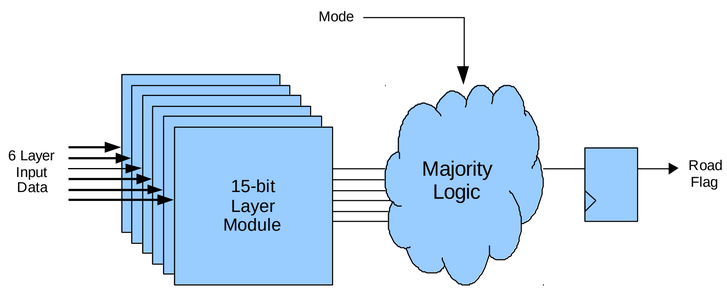


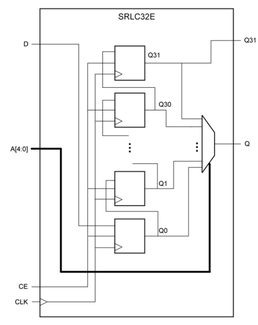


A **Pattern Module** (right) consists of several **Layer Modules** (below) and majority logic. A fired pattern is called a road and the Road Flag output is set by majority logic and controlled by global mode bits. Currently defined global modes are: "Require All Layers", "Miss 1 Layer", and "Miss 2 layers".



Three SRLC32E's and a bit of glue logic are used to construct a 15-bit pattern match **Layer Module**. The 96-bit pattern can be changed at any time and is shifted in when LOAD is high.

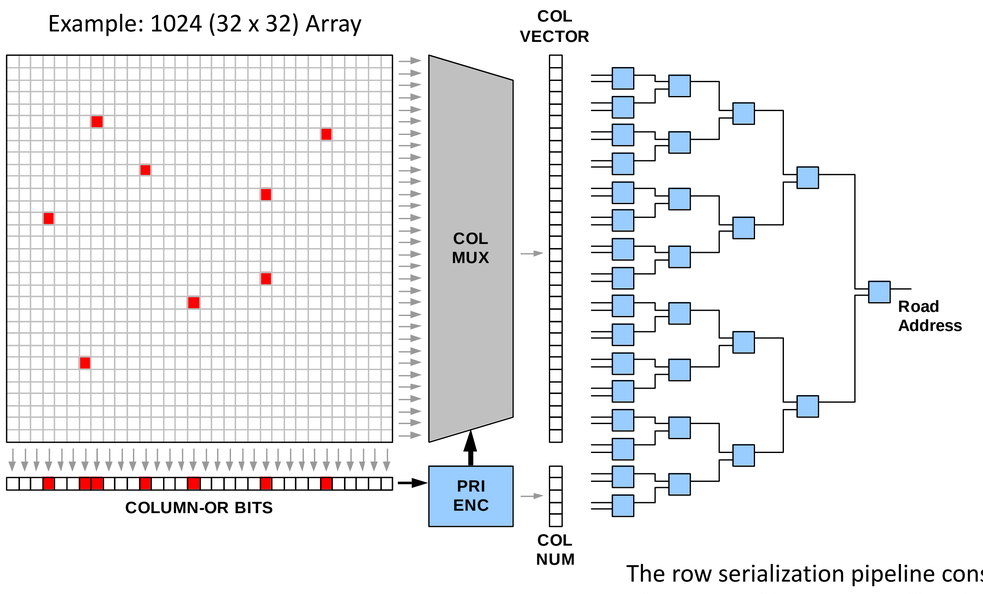




The SRLC32E primitive is most often used to implement a variable depth pipeline in one SLICE-M configurable logic block (CLB) in Xilinx 7-Series and UltraScale FPGAs. The 5 bit value on input A determines the shift register depth, up to 32 stages.

To implement our PRAM cell in the Pattern Module Array we utilize this SRLC32E primitive in a different way. A 32 bit pattern is shifted in and stored in the shift register, and the incoming data is driven on input A. This enables us to create a small fully programmable 5-bit pattern checker which supports ternary or "don't care" bits in any bit position.

Road Serialization Logic



When the End-of-Event signal is observed the array of road flags is loaded into registers and the serialization process begins. First, the logical OR of the road flags in each column are stored in registers. These column bits are used by the synchronous "peel away" priority encoder and MUX to select populated columns which are sent down the row serialization pipeline.

The row serialization pipeline consists of 5 stages of internally buffered sort nodes, one of which is shown below:

Each sort node is controlled by a finite state machine (FSM). Based on the inputs A and B and FIFO status the FSM determines which address is sent to output C. The FIFO, which is constructed from distributed RAM elements, is deep enough to insure that no road addresses are lost.

The latency from EOE to first road output is always 7 clock cycles. All found roads are output in a contiguous block with no gaps or nulls in between.

http://www-ppd.fnal.gov/ATCA/

Performance and Device Utilization

The FPGA PRAM design has been successfully implemented in Kintex UltraScale KU040 devices. Both 1k and 4k pattern designs achieve 250MHz operation in the slowest (-1) speed grade. (The ASIC design goal is to ultimately achieve a few 100 k patterns per chip.)

FPGAs support many different types of I/O interfaces; each type has bandwidth and latency trade-offs which can affect overall system performance. The FPGA PRAM design will be used to test and evaluate different types of I/O interfaces before they are incorporated into future PRAM ASIC designs.

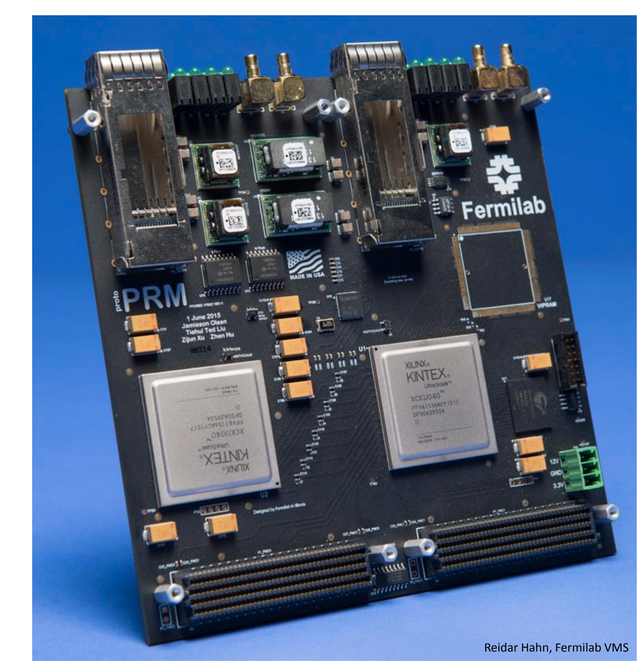
1k (32x32) Pattern Array, KU040	
FF	2%
LUT	17%
MEMORY-LUT	22%

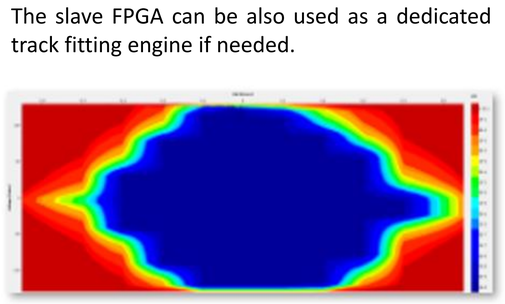
4k (64x64) Pattern Array, KU040	
FF	9%
LUT	70%
MEMORY-LUT	89%

ProtoPRM Mezzanine

A prototype Pattern Recognition Mezzanine (ProtoPRM) has been designed as part of the L1 Tracking Trigger demonstration system for CMS. This mezzanine consists of two Kintex UltraScale FPGAs and supports both the VIPRAM ASIC and PRAM FPGA designs. The mezzanine can operate stand-alone or as part of the Pulsar IIb custom ATCA processing board. This mezzanine card provides a powerful and flexible R&D platform in which:

- The slave FPGA can be configured as the PRAM chip for initial performance studies
- The latest two-tier VIPRAM ASIC is supported and can be compared side-by-side with the FPGA PRAM for performance optimization studies for future generation ASIC designs.
- The slave FPGA can be also used as a dedicated track fitting engine if needed.





An IBERT "eye diagram" showing a typical GTH receiver margin at **15.625 Gb/s** on the ProtoPRM inter-FPGA local bus.