



FM3000 Policy Engine

Embedded High-bandwidth Packet Processing

White Paper

June, 2008

Introduction

The FM3000 family of devices contain a Policy Engine that can process frame headers at an aggregate line rate of 240Gb/s or up to 360M packets per second. For telecom applications, this Policy Engine can be used for packet forwarding, service differentiation, VLAN encapsulation, header modification, load distribution and security. This white paper will describe these functions in more detail.

Policy Engine Overview

Incoming frames to the FM3000 are stored in shared memory while the frame headers are sent to the Policy Engine. The Policy Engine performs several functions as shown in the header processing pipeline below. This section gives a very brief description of each block in this pipeline. More details about these blocks can be found in the FM3000 data sheet.

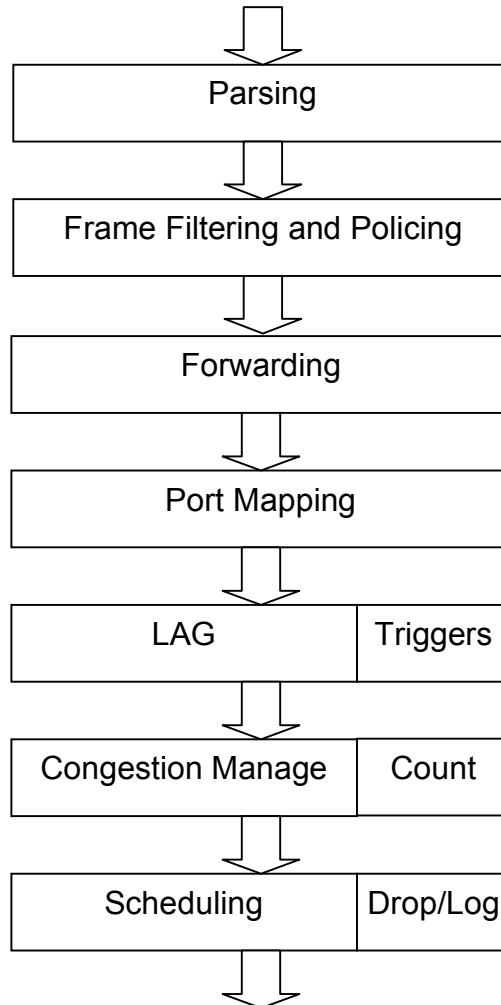


Figure 1: Policy Engine processing flow

Parsing

The operation of the parsing block depends on the port configuration. The port can be configured to receive standard Ethernet L2 frames or a variety of ISL tagged frames. The frame header received is parsed and a switch priority is associated with the frame using the 802.1p priority field or the switch priority field in the ISL tag. The switch priority can be modified by ACL rules in the Frame Filtering block.

Frame Filtering and Policing

The parsed frame header is then presented to a frame Filtering and Forwarding Unit (FFU), which produces a set of orthogonal actions on the frame using different ACL matching rules. The most common ACL actions are permit, deny, modify, count, mirror, trap, and log. Even though egress ACL actions are determined in this block, the egress action does not take place until the end of the header processing flow. These ACL egress actions include counting, dropping or logging.

The policing function monitors the incoming traffic rate and can change the priority if needed. The policers are implemented using dual token buckets while rate-limiting function can be achieved with the generation of pause frames at the ingress.

Forwarding

The forwarding block includes a VLAN table, a spanning-tree state table and a MAC address table. The forwarding function will use these tables to determine forwarding information. If the destination address is unknown and learning is enabled, the packet is flooded. The address can also be automatically learned or raise security violations depending on configuration of the switch.

Port Mapping

The Port Mapping block uses the forwarding information to compute an index into the GloRT table, and retrieve a final destination mask, which contains one bit per output port where the frame will be delivered. This block also implements link aggregation pruning to load balance traffic across multiple links.

LAG

This block creates a link aggregation mask based on the result of hashing using various frame header fields which is used to determine which port of a link aggregation group will be used to transmit the frame. This block can also perform load distribution across members of the LAG group.

Triggers

The triggers are low-level elements that can be used to modify traffic flows. There are up to 64 triggers. The output of a trigger is a destination mask and an updated destination port number. The destination mask is ANDed with the link aggregation destination mask.

Congestion Management

The congestion management maintains information about the size of the different queues and can mark, discard and/or send congestion management frames such as VCN frames back to the originator when the size of queues exceeds certain limits.

Counting

FocalPoint devices maintain numerous per-port and switch-wide frame counters that provide management with statistical information about the state of the switch and of the network in general. ACLs rules can also be used to count various frame statistics.

Scheduling

The egress scheduler uses various information about the ports and the queues to determine when to schedule a packet for transmission. The egress scheduler includes traffic shaping and pacing algorithms. If an ACL action is 'Deny', the packet will be dropped in the scheduling block. Egress ACL logging can also occur in this block.

FFU Function

The Frame Filtering and Forwarding Unit (FFU) is the heart of the Policy Engine. Up to 128-bytes of a frame header can be examined and up to 78 of these bytes can be used to make policy decisions. When the frame header is presented to the Frame Filtering and Forwarding Unit, it associates one or more actions with the frame. This unit contains 32 consecutive slices containing both TCAM and SRAM. The last slice(s) can optionally be used for egress ACLs that apply in parallel to multiple egress ports.

Each slice has the following elements:

- Configuration to select which frame header fields are selected for the TCAM comparison
- 512-entry x 36-bit TCAM block used for key comparison
- Hit detection circuitry which can cascade across consecutive slices to create keys larger than 36 bits
- Priority hit detection to determine the hit in each slice with the highest index
- 512-entry x 40-bit SRAM block to store an ingress action(s) associated with the highest priority hit
- The last slice can optionally feed its final 512 hits into the egress ACL unit
- Configuration registers to control the slice behavior for various frame scenarios and key configurations

The following sections will describe how the FFU can be used for various Policy Engine functions. More information can be found in the FM3000 data sheet and the FocalPoint ACL Application Note.

Packet Forwarding

The Policy Engine can be configured to forward packets to the proper unicast or multicast destination ports. It can also be used to drop or redirect traffic. A block diagram showing how the Policy Engine can be used to forward traffic is shown in the figure below.

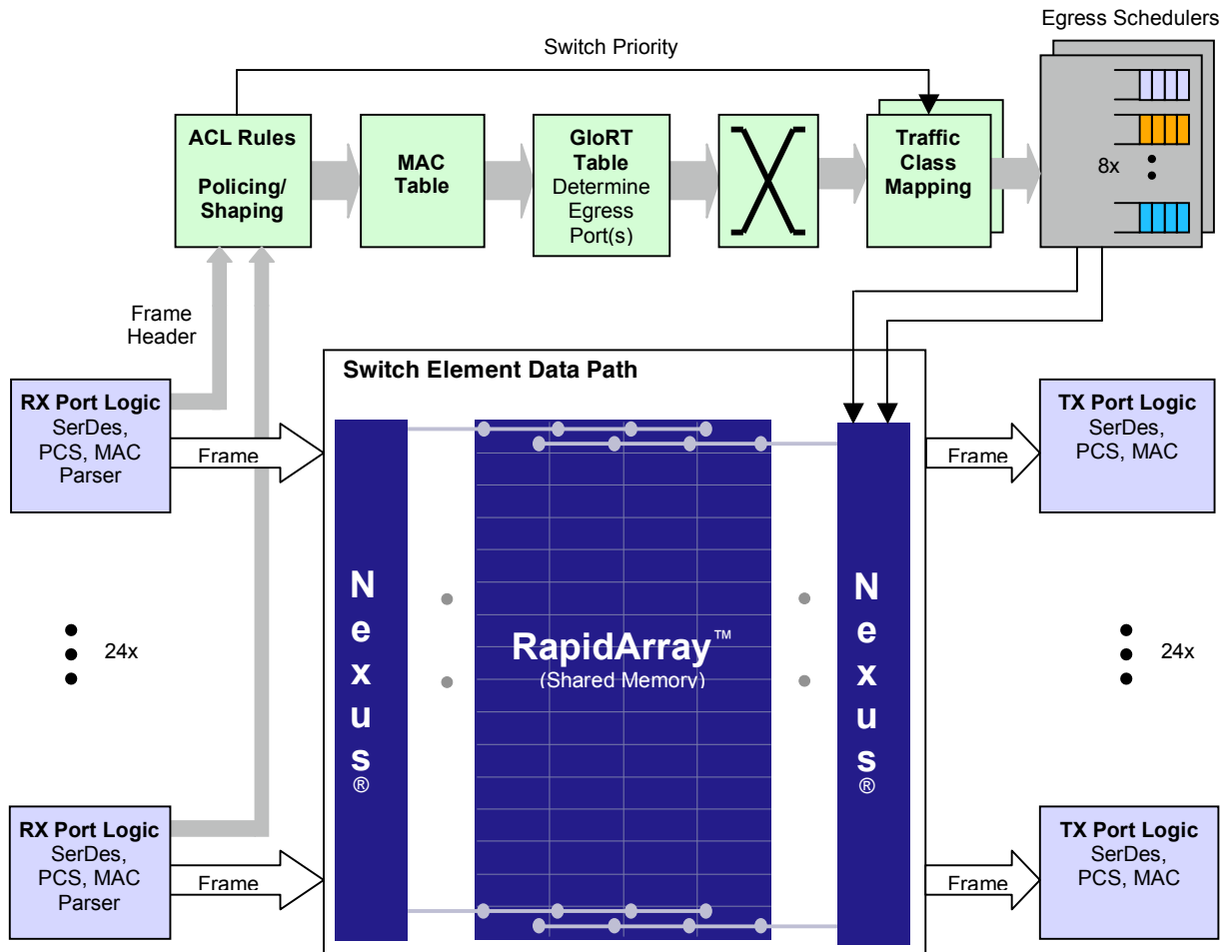


Figure 2: FM3000 forwarding and service differentiation.

For standard L2 frames, a 16K MAC table is used to match the following fields in the frame header to alter the course of the packet.

- Ingress VLAN association / Source MAC Address / Source GloRT / Source Port which are used for learning (if enabled), security checking and ingress VLAN/FID (Forwarding Information Database) filtering.
- Egress VLAN association / Destination MAC Address / Destination GloRT which is used for forwarding and egress VLAN/FID filtering.

The result of the MAC table lookup is an index into the GloRT table which contains a set of 4K egress port masks that can be used for either unicast or multicast forwarding. When using F64 ISL tags for the frame header, the GloRT index is included in the header and the MAC table can be bypassed. The frame headers are then sent to the egress scheduler(s) where the frames are de-queued from shared memory for transmission.

Service Differentiation

Various blocks can be used in the Policy Engine pipeline for service differentiation as shown in figure 2. The FFU can apply ACL rules to police and rate limit incoming traffic based on various header fields. The policers themselves are implemented using dual token buckets named "committed" and "excess". Tokens are used from the "committed" bucket first. When the "committed" TB is depleted, packets are marked as yellow and will use the "excess" TB.

Ingress rate limiting can be achieved using special ACL rules called triggers. The triggers implement sixteen drop-based rate limiters. A given frame may be mapped to one or more of these rate limiters. If the rate limiter is out of profile, then the frame's destination mask will be filtered by a configured drop mask. Example applications of this feature include the following:

- Limit chip-wide proportion of flooding traffic to protect against denial-of-service attacks.
- Limit low-priority traffic sent or trapped to the CPU in a rate-controlled manner rather than in a watermark-controlled manner.
- L2 policing: Limit bandwidth directed to a particular DMAC, VLAN, or egress port.

ACL rules are also used to determine the frame priority based on various frame header fields. This switch priority is carried along with the header where it is mapped into a traffic class, which is used by the egress scheduler. The egress scheduler supports up to 8 traffic classes, which can be scheduled using strict priority or deficit round robin for minimum bandwidth guarantees. The scheduler can also apply traffic shaping. See the FocalPoint Congestion Management Applications Note for more information.

VLAN Encapsulation

VLAN encapsulation is useful for applications such as Carrier Ethernet Provider Bridging or Ethernet based DSL. The FM3000 Policy Engine allows VLAN tag insertion, removal and/or modification. The figure below shows the VLAN encapsulation functional blocks.

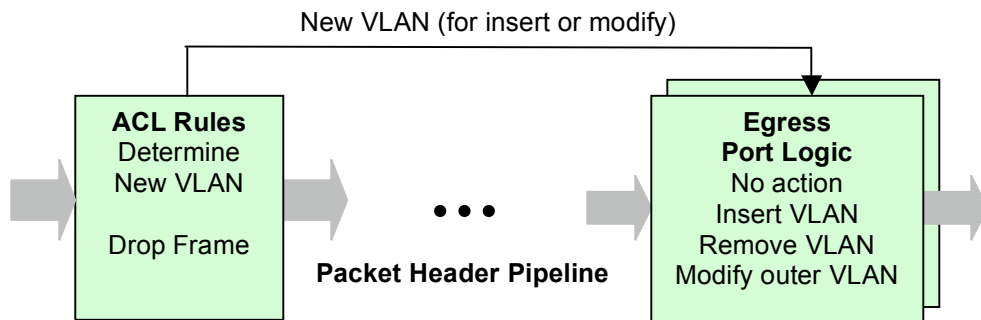


Figure 3: FM3000 VLAN encapsulation functional blocks.

The FFU can apply ACL rules to determine if a tagged or untagged frame should be dropped or if a new VLAN-ID or new VLAN priority should be applied at the egress when a tag is inserted or modified. At the egress, an outer VLAN tag can be inserted or

modified. After this, the outer VLAN tag can be modified based on the result of the ACL rules applied in the FFU. See the VLAN Encapsulation Applications Note for more information.

Header Modification

Various ACL rules can be applied in the FFU to modify fields in the frame header. The table below shows the fields that can be modified.

ACL Action	Description
Change Switch Priority	This can be used to change or set the priority of an incoming frame. The switch priority is used at the egress to determine traffic class for CoS scheduling.
Change User Bits	These 8 user bits are included in the F64 ISL tag header. These user bits can be passed through to a downstream device such as an NPU or CPU. For example, the FFU could be used to pre-classify a packet based on header information and pass along the classification results in the F64 user bits.
Change VLAN ID	The VLAN ID can be changed based on ACL rules. This can be used in applications such as Carrier Ethernet or Ethernet based DSL.
Change VLAN Priority	The VLAN priority can be modified to map the frame into a new switch priority value before the egress scheduler. This new priority will also be carried in the VLAN tag which can be useful in applications such as Carrier Ethernet or Ethernet based DSL.

More information can be found in the FocalPoint ACL Applications Note or FM3000 data sheet.

Load Distribution

The FM3000 load distribution function uses a hash algorithm to distribute traffic uniformly among several ports on a per-flow basis. Load distribution has several applications such as distributing traffic across multiple switches in a telecom backplane or distributing the packet processing load across multiple application blades in a network security system.

FM3000 Series devices use a modified Pearson's hash that is highly effective in load distribution while incurring a modest implementation cost. Various parts of the L2/3/4 header can be used as a source for the hashing function. The source is hashed to a 12-bit value (giving 4096 intermediate bins) and that result is distributed among the output links in the hash group using modulo division. For purposes of load distribution, two hash values can be calculated from the header fields of each frame:

- Layer 3/4 Hash (36 bits)
- Layer 2/3/4 Hash (48 bits)

The keys to these hash functions are constructed in a configurable manner in order to provide the following features:

- Symmetry -- Hash value remains the same when source and destination fields are swapped.
- Static field dependence -- Support for including a specific set of header fields in the hash function.
- Dynamic field dependence, based on frame type -- Certain fields can be omitted or included when a frame is IPv4/IPv6.

For more information on load distribution, see the white paper entitled Load Distribution in Telecom Servers using FocalPoint.

Security

The FM3000 Series devices provide both MAC table and VLAN security mechanisms, which can drop and trap frames based on various security violations. But additional security levels can be provide when using the Policy Engine. FocalPoint has the ability to use Access Control Lists (ACLs) to prevent security violations such as denial of service attacks. A set of rules and a set of actions for each rule can be defined as described in the FFU section above. Counters can help identify misbehaving flows or they can be trapped to the local CPU. The local CPU can then adjust the ACL rules to deny access to these flows. If FocalPoint is between the network and the application blades, it can offer a first line of defense and also off-load some of these security functions from the NPUs or CPUs on these blades.

Triggers

In addition to the ACL rules described above, the switch also contains a general set of rules for modifying frames at the last stage of the Policy Engine pipeline. These rules are user programmable and are referred to as triggers. A total of 64 trigger rules are supported.

A trigger rule is defined by two parts: a match condition and an action specification. The match condition is a programmable Boolean expression. If all of the conditions defined in the expression are true, then the trigger “fires” and the action specification is applied to the packet. For a given packet, any number of triggers may fire. In the case of conflicting action specifications, an action resolution step determines exactly how the packet will be handled.

Unlike ACL rules that use bit masks to match various programmable fields in the header, trigger rules match on fixed fields or conditions such as source MAC address, Ethertype or whether the MAC destination address was found in the MAC table. There are 18 match conditions available. The trigger actions are listed in the table below.

Trigger Action	Description
Forwarding	This action can be used to change or set the forwarding of a frame. Options include leave-as-is, forward, redirect or drop.
Trapping	This action can be used to trap or log the frame. Options include leave-as-is, trap, log or do not trap or log.

Mirroring	This action can be used to mirror a frame. Options include mirror to port, mirror to GloRT or mirror with truncate.
Change Switch Priority	This action can change the switch priority of the frame.
Change VLAN	This action can change the VLAN-ID of the frame.
Disable Learning	This action can disable learning on the L2 source MAC address into the MAC address table.
Rate Limiting	This action assigns one of 16 Trigger Rate Limiters to the frame. A drop mask will be applied to the frame if the specified rate limiter's bandwidth limit is exceeded.

More information can be found in the FocalPoint ACL Applications Note and FM3000 data sheet.

Loopback Processing

In some cases, it may be necessary to process a frame header multiple times through the Policy Engine. To do this, one or multiple ports on the FocalPoint switch can be configured in loopback mode. During the first pass through the Policy Engine the frame can forward to a loopback port. Since the FFU can use the ingress port number as part of the forwarding decision, after the second pass through the Policy Engine, the frame can be forwarded to a standard egress port.

For example, Ethernet based DLS requires the insertion of an outer provider VLAN tag, while also modifying the inner and outer VLAN tag. To do this, the first pass through the Policy Engine can be used to modify the inner tag using ACL rules while also forwarding it to the loopback port. The second pass through the Policy Engine will then add an outer VLAN tag, which is modified using a different set of ACL rules. Since the ACL rules can look at which port the frame is received on, the first and second pass can use different sets of ACL actions.

Conclusion

The FM3000 Policy Engine provides a high bandwidth header processing pipeline that can be used for a variety of applications. A sophisticated set of ACL rules along with other functional blocks support operations such as service differentiation, VLAN encapsulation, load distribution and security. By offloading these functions from other NPUs or CPUs in the system, improved cost-performance trade-offs can be made by the system designer.

Fulcrum Microsystems, Inc.
26630 Agoura Road
Calabasas Hills, CA 91301
818.871.8100
www.fulcrummicro.com