Realtek Semiconductor Corp.

RTL9310
Architecture Overview

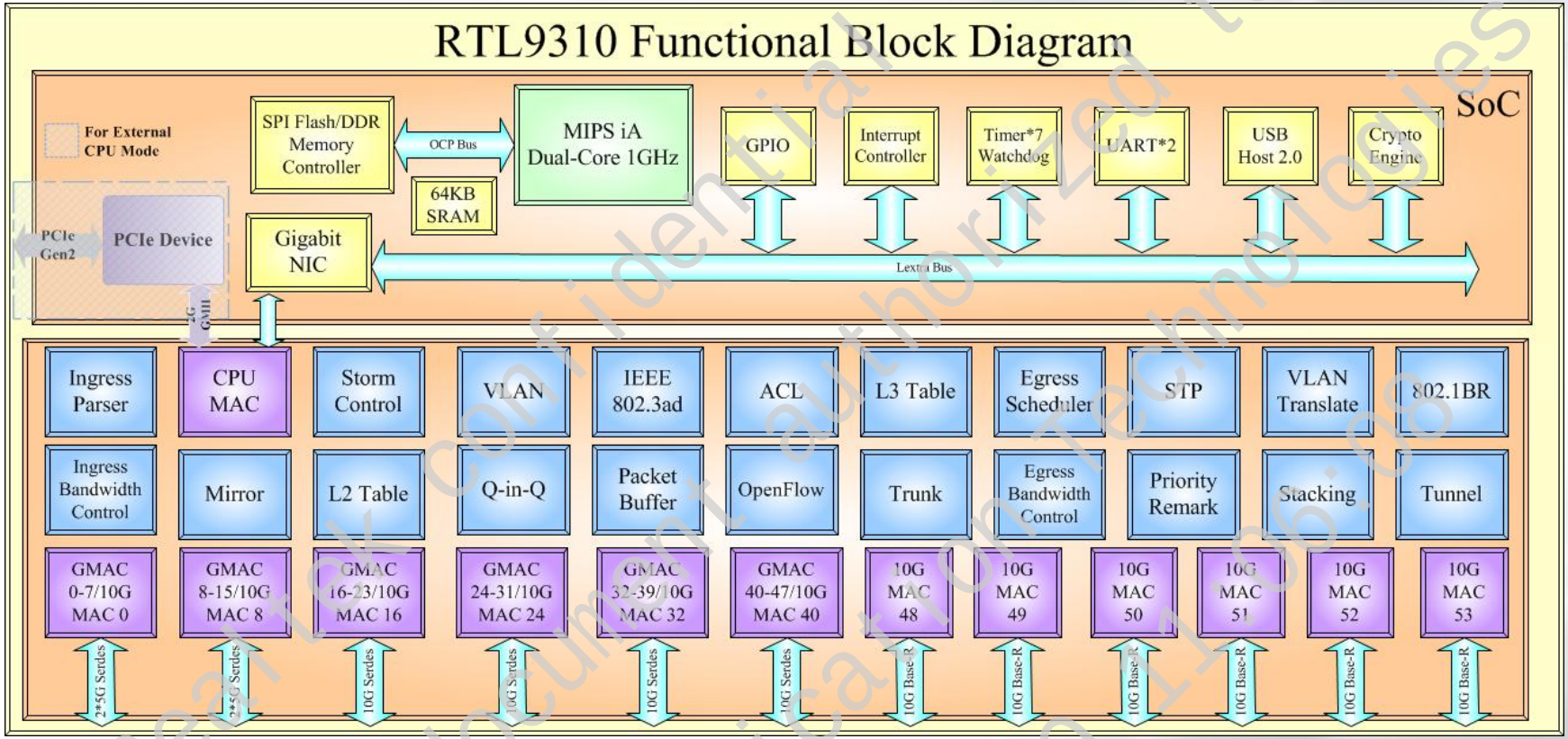
Realtek Semiconductor Corp.

System Description

Content

- System Description
- VLAN
- L2 MAC Address Lookup
- ACL
- Traffic Policer
- L3
- IP Tunnel
- Spanning Tree
- Traffic Isolation
- Link Aggregation
- Mirroring
- sFlow
- RMA
- Ingress/Egress Bandwidth Control
- IP MAC Binding
- Storm Control
- Attack Prevention
- Priority Decision
- Scheduling
- Egress Remarking
- Flow Control
- SWRED
- OAM & 802.1ag, ETH-DM
- MPLS
- 802.1BR Port Extension
- VXLAN
- OpenFlow
- NIC
- Stacking & Remote Access
- EEE
- LED
- MIB Statistics

Functional Block Diagram



- 240Gbps Switching Capacity
- 16Mbits Embedded Packet Buffer

CPU Sub-system

- MIPS iA(interAptive)
  - Dual-Core 1GHz (1.7 DMPIS/MHz)
  - 32KB/256KB L1/L2 cache
- SPI Flash
  - SPI NOR(Max. 64MB) and NAND(Max. 512MB) with ECC
  - Up to 100MHz
- DDR Memory
  - DDR3/DDR4/LPDDR3, Max. 2GB.
  - 16/32-bit I/F, up to 800MHz.
  - 2*16-bit chip supported (combined as 32-bit)
- UART*2
  - UART2 is shared with EJTAG
- USB Host 2.0

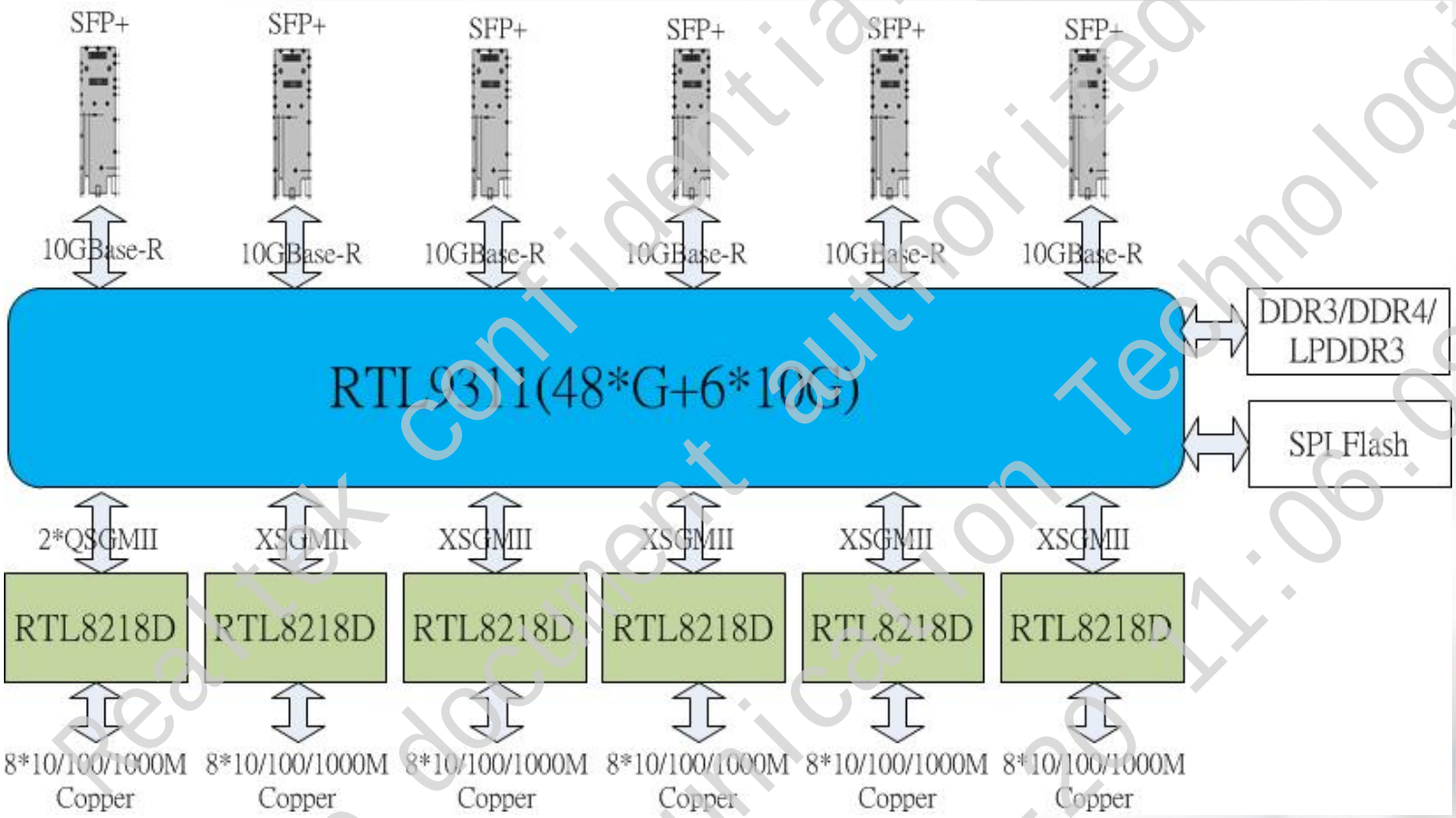
CPU Sub-system (Cont.)

- Timer*7 + Watchdog
  - Each timer can be set to timer or counter mode
  - Watchdog can reset CPU directly or trigger an interrupt before resetting CPU
- GPIO
  - 32 internal GPIO
  - 74 external GPIO (use RTL8231 as GPIO extender)
- I2C/SPI Master
  - I2C master supports 12 set with 2 clock
  - SPI master supports 2 set with 2 chip select

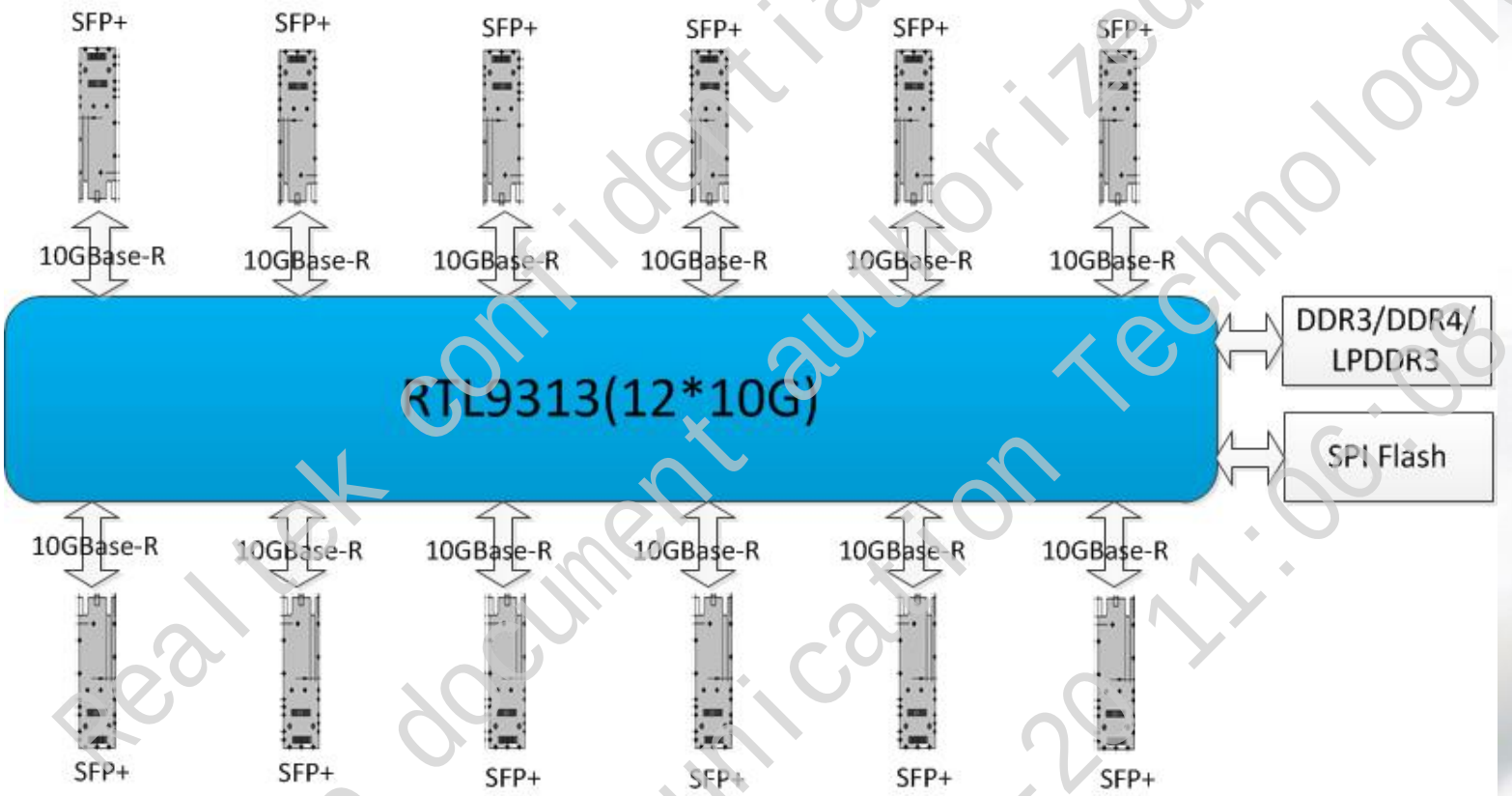
Application Model

- RTL9311
  - 24*GE + 6*10G
  - 48*GE + 6*10G
- RTL9312
  - 24*2.5G + 6*10G
- RTL9313
  - 12*10G

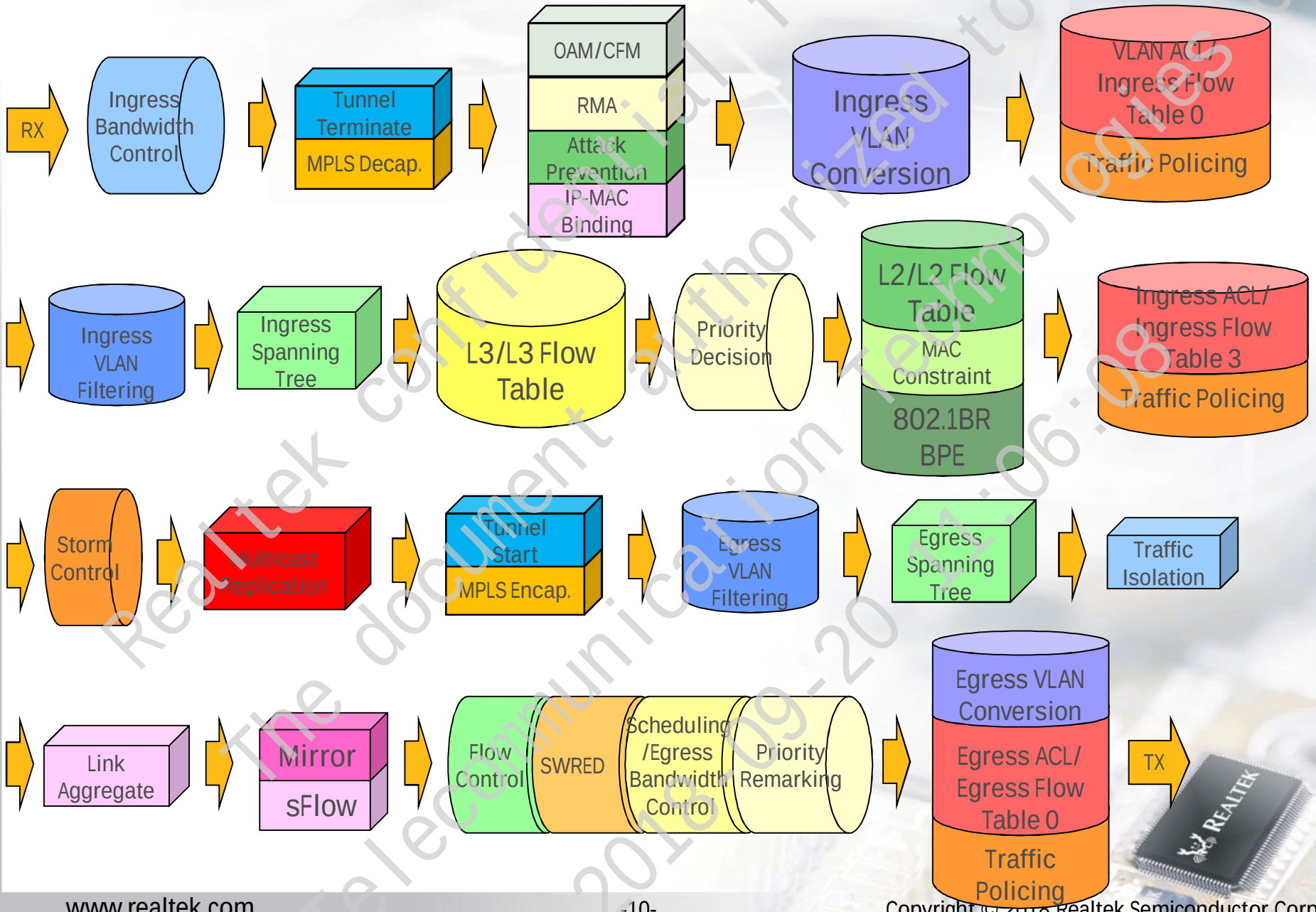
Application Model - RTL9311



Application Model - RTL9313



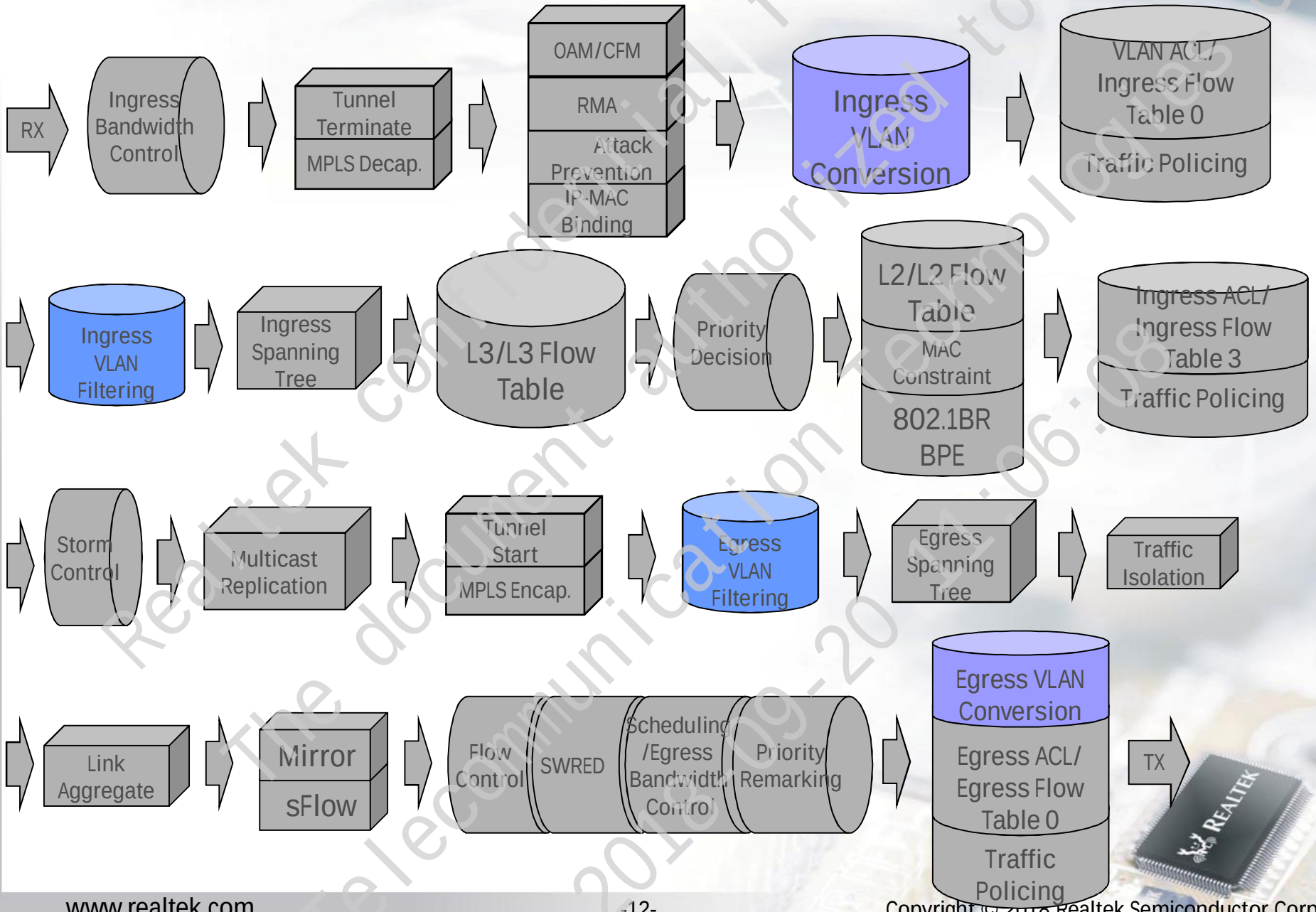
Packet Processing Pipeline



Realtek Semiconductor Corp.

VLAN

Packet Processing Pipeline



VLAN Overview (1/3)

- Inner, Outer, Extra VLAN tag parsing
- Global inner/outer/extra TPID pools
  - Per port can specify the recognized TPIDs from each pool
- Per-port and per-tag-status selects forwarding VID: inner or outer
- 4K VLAN table
  - Per VLAN map to one of 16 VLAN profile for:
    - New SA learning option (H/W auto, learn-as-suspend, disable)
    - New SA forwarding option
    - IPv4/IPv6 multicast bridge mode (MAC-based or IP-based)
    - Unknown L2/IPv4/IPv6 multicast flooding mask
  - 128 MST Instance/Filtering Database
  - 16-bit group mask (ACL's match key, predefined application packets)
- Port-based VLAN
- Protocol-and-Port-based VLAN

VLAN Overview (2/3)

- MAC-based/IP-subnet-based VLAN
- Ingress/Egress VLAN Filter
- VLAN Translation / Q-in-Q
  - 2048 Ingress VLAN Conversion (IVC) entries
    - VLAN translation (C->C',C->S,C->S+C)
    - 4*32 VLAN range entries
  - 1024 Egress VLAN Conversion (EVC) entries
    - VLAN translation (S+C->C,C'->C,S'->S,S->C)
    - 4*32 VLAN range entries
  - MAC-based N:1 VLAN translation for known unicast packet
    - TR101 (CTC 2.0)
- VLAN transparent mode
  - Keep the original tag status and content

VLAN Overview (3/3)

Table	Description	Size
VLAN table	Includes VLAN member, untagged member and multiple spanning tree instance etc.	4096
VLAN profile table	VLAN additional configuration.	16
Protocol-and-Port-based VLAN table	Specifies IEEE 802.1v (Protocol-and-Port-based VLAN) configuration. Global eight protocol values are supported and each port can have different PPB VLAN value for each global protocol value.	8
Ingress VLAN conversion table/MAC-based VLAN table/IP-subnet-based VLAN table	Adds or removes C-TAG and S-TAG, shared with MAC-based/IP-subnet-based VLAN.	2048
Egress VLAN conversion table	Egress VLAN conversion.	1024

Agenda

- Basic VLAN Configuration
- Ingress VID Decision
- VLAN Forward and Filter
- Egress VID Decision
- Egress VLAN Tagging

VLAN TPID Configuration (1/2)

- Global configuration
  - 4 inner TPIDs can be defined
  - 4 outer TPIDs can be defined
  - 1 extra TPID can be defined
- Per port configuration
  - Each inner TPID has an enable bit
  - Each outer TPID has an enable bit
  - Extra tag also has an enable bit

VLAN TPID Configuration (2/2)

Inner TPID list

Index 1	8100
Index 2	8200
Index 3	8300
Index 4	8400

	Idx 4	Idx 3	Idx 2	Idx 1
Port 1	1	0	0	1
Port 2	0	1	0	1
.	. . . .			
Port N	0	0	1	1

outer TPID list

Index 1	88A8
Index 2	88A9
Index 3	88AA
Index 4	88AB

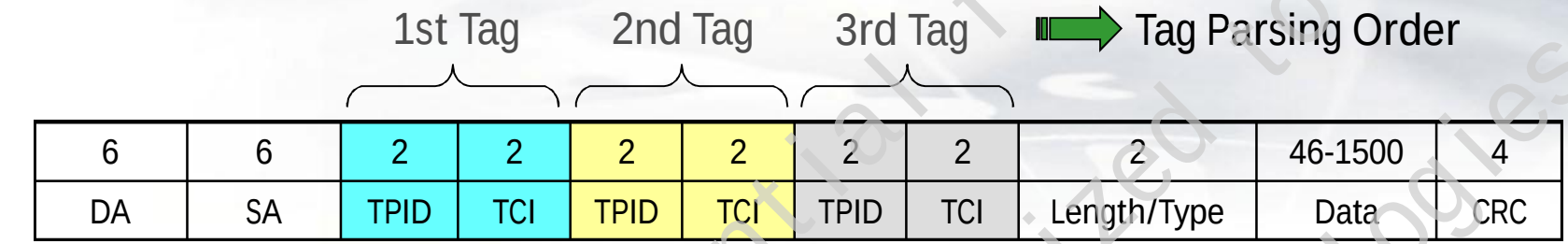
	Idx 4	Idx 3	Idx 2	Idx 1
Port 1	0	1	0	1
Port 2	1	1	0	0
.	. . . .			
Port N	1	0	1	0

extra TPID list

Index 1	8100

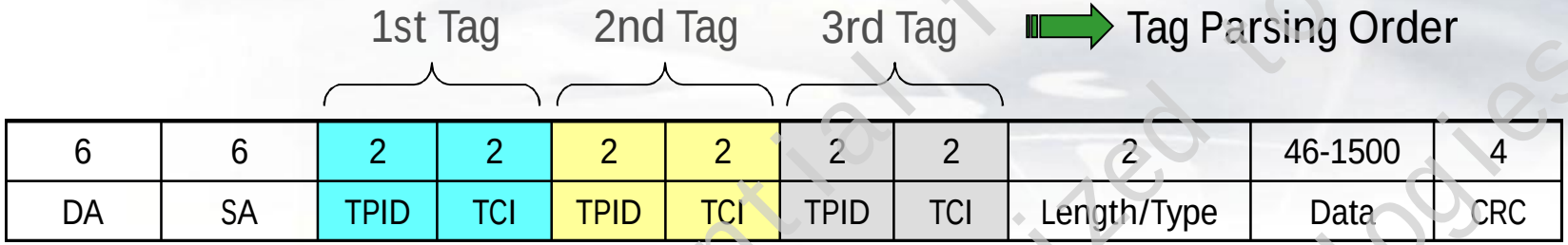
	Idx 1
Port 1	0
Port 2	1
.	. . . .
Port N	1

VLAN Tag Parsing (1/2)



Packet tag status	Outer	Inner	Extra

VLAN Tag Parsing (2/2)



Packet tag status	Inner	Extra

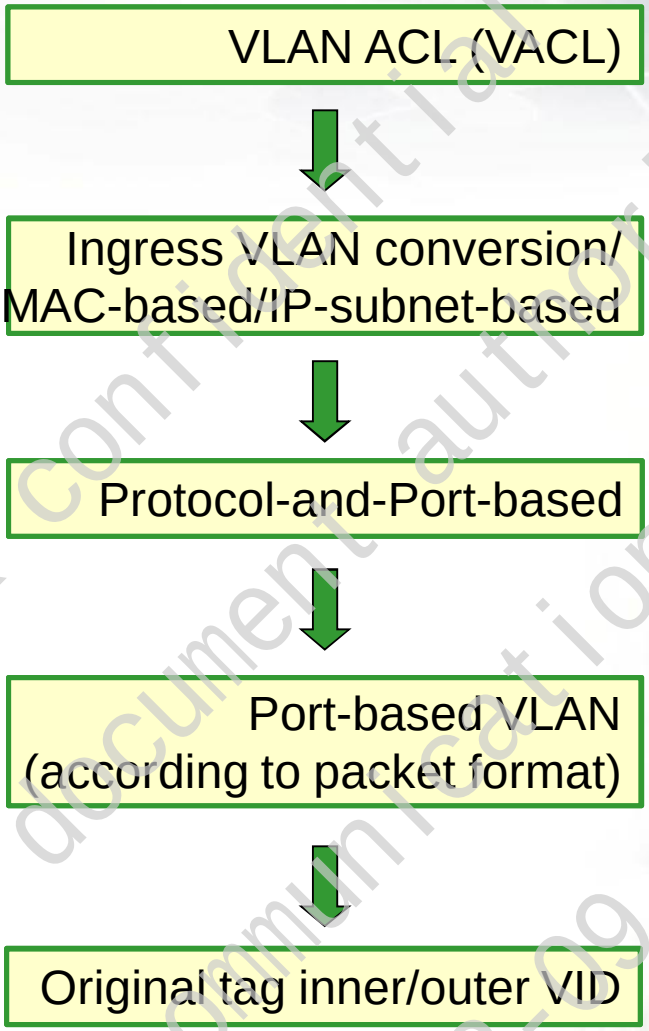
Accept Frame Type

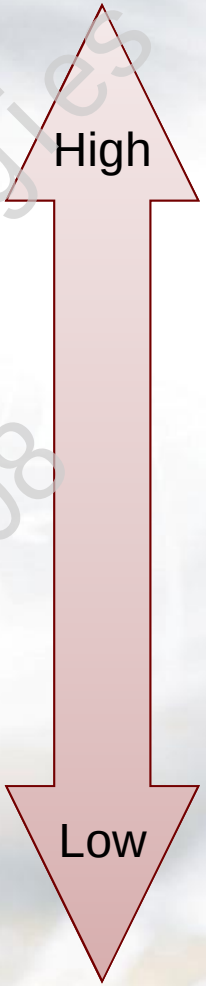
- Per port configurable
- Supports inner and outer VLAN configuration
- The permissible values for the inner parameter are
  - Only VLAN-tagged frames
  - Only untagged and priority-tagged frames
  - All frames
- The permissible values for the outer parameter are
  - Only VLAN-tagged frames
  - Only untagged and priority-tagged frames
  - All frames

CFI Operation

- Legacy function
  - CFI was used for compatibility between Ethernet and Token Ring networks
- Global action for tag with ICFI = 1
  - Forward
  - Drop
  - Trap
- Global action for tag with OCFI = 1
  - Forward
  - Drop
  - Trap

Ingress Inner/Outer VID Decision



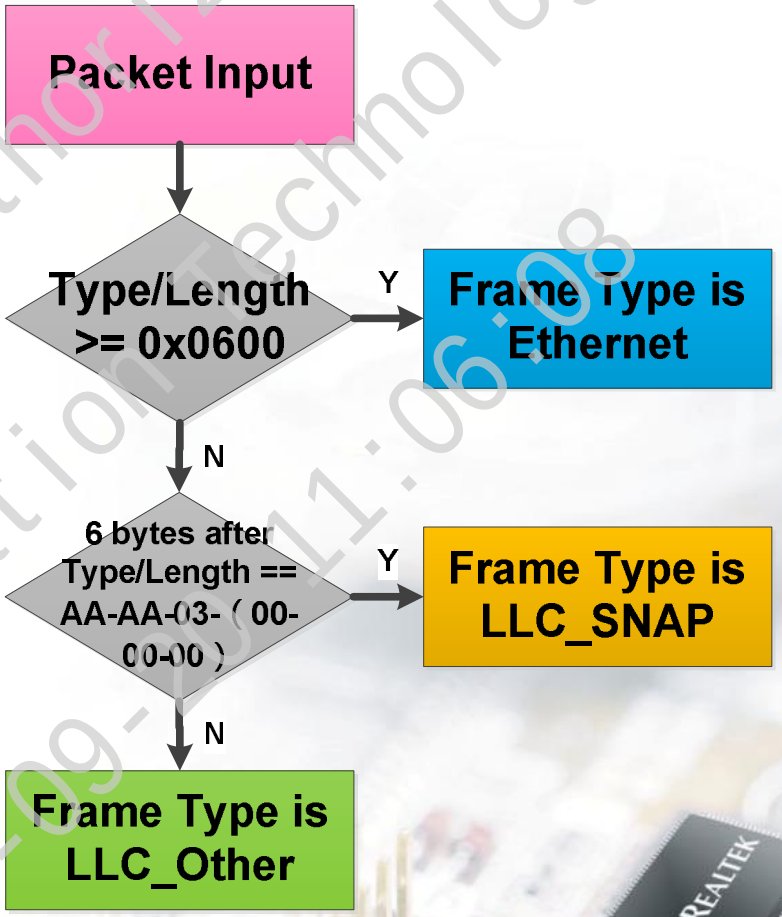


Port-based VLAN

- Per port configuration
  - Apply inner PVID according to packet format:
    - Inner untagged and inner priority tagged packet
    - Inner untagged packet
    - All packet
  - Apply outer PVID according to packet format:
    - Outer untagged and outer priority tagged packet
    - Outer untagged packet
    - All packet

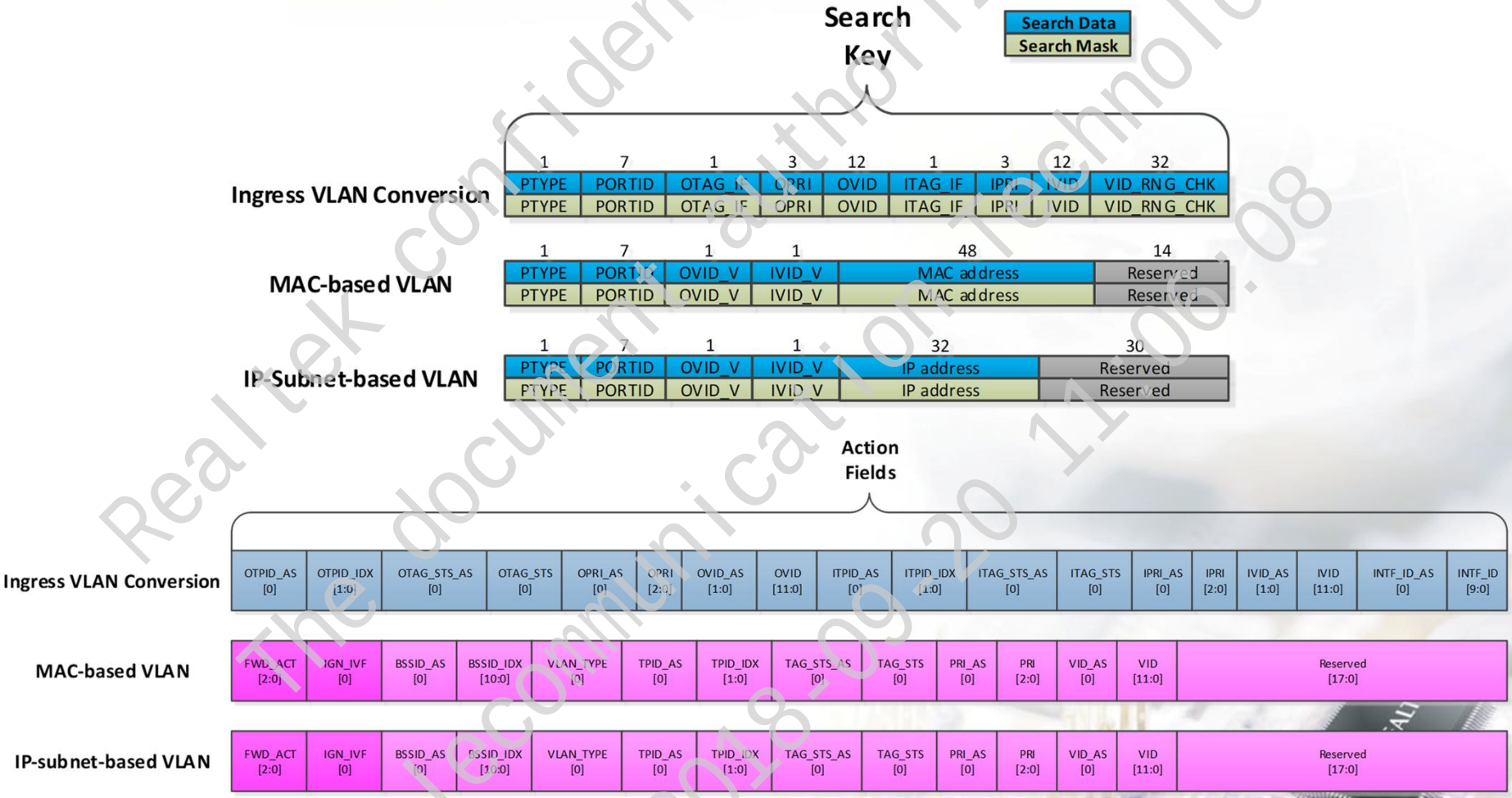
802.1v Protocol-and-Port-based VLAN

- Protocol-and-Port-based VLAN
  - VID applies to untagged and priority tagged packet
  - Priority applies to untagged packet
- Global Configuration (8 Groups)
  - Frame type:
    - Ethernet (IEEE 802.3)
    - LLC_Other
    - LLC_SNAP
  - Ethertype/(DSAP-SSAP) value
- Per Port and Per Group Action
  - Apply to inner VLAN or outer VLAN
    - Assign VID
    - Assign priority



Ingress VLAN Conversion (1/4)

- IVC (Ingress VLAN Conversion) table includes 16 blocks, 128 entries per-block, total 2048 entries
- Each block can be used for IVC or MAC-based or IP-subnet-based VLAN



Ingress VLAN Conversion (2/4)

- Configurable VID(or VID range) of search key from inner or outer
- Per entry has a hit indication flag
- Choose lowest index entry if multiple entries hit

Search key

1 bit	12 bits	32 bits	6 bits	3 bits
Valid	Original VID	Original VID Range	Port/Trunk ID	Original Tag Priority
	Original VID Mask	Original VID Range Mask	Port/Trunk ID Mask	Original Tag Priority Mask

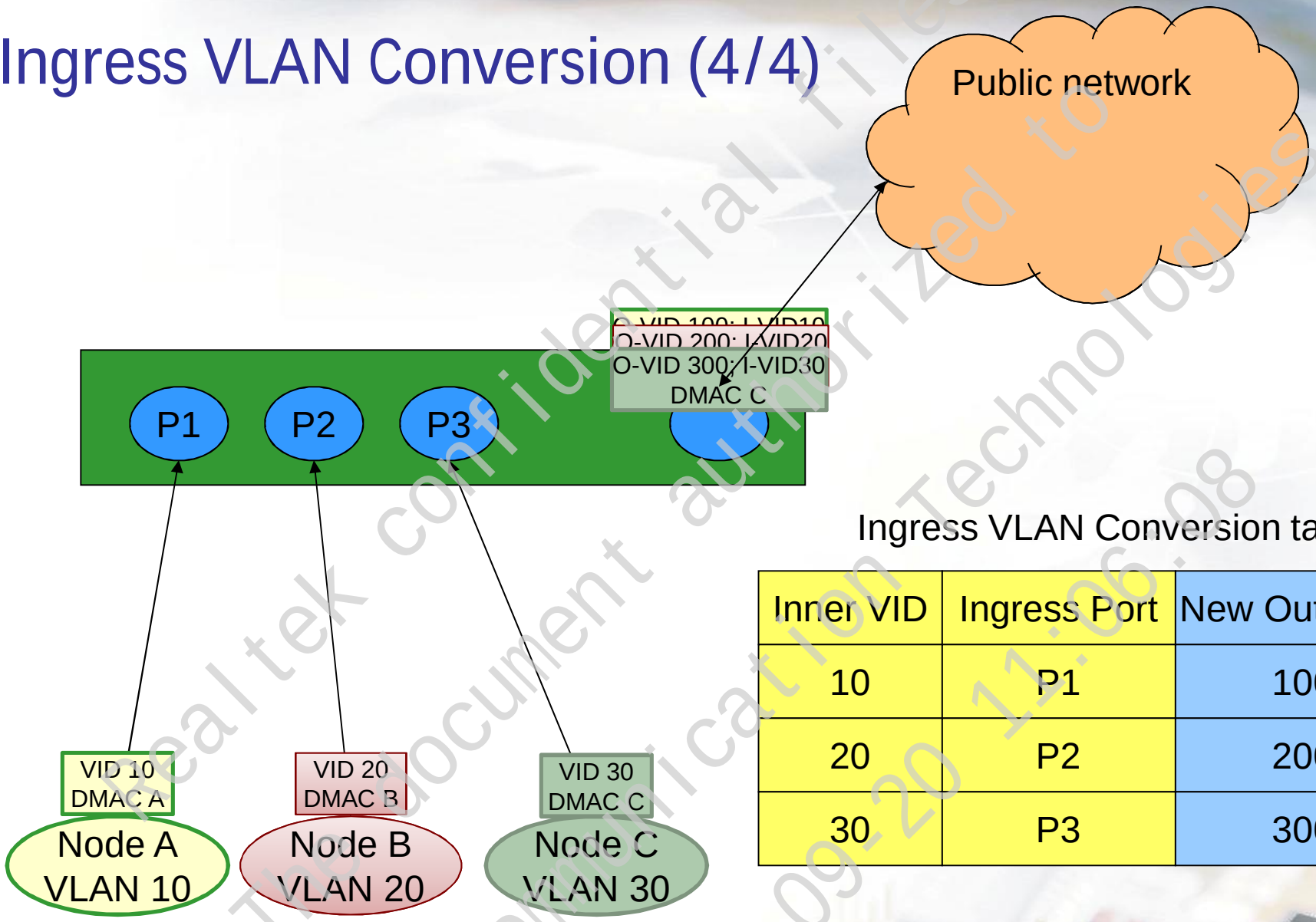
Translation field

Action		Value
VID assign	Assign a VID with New VID	New VID
	Shift VID with New VID	
	Copy from original I-VID/O-VID	
Priority assign	Assign a Priority with New Priority	New priority
	Copy from original inner/outer priority	
Tag status assign		New Tag status
TPID assign		New TPID
Interface ID assign		Assigned Interface ID

Ingress VLAN Conversion (3/4)

- Per Ingress Port VLAN Conversion Control
  - Enable lookup for specified port
  - Lookup miss action for inner/outer VLAN-tagged packet: Forward, Drop
  - Specify one of the VID range check tables
- System supports 4 VID range check tables
  - Each table has 32 dedicated range check entries
    - VID range lower bound
    - VID range upper bound
    - Range check data type
      - original I-VID range check
      - original O-VID range check

Ingress VLAN Conversion (4/4)



Ingress VLAN Conversion table

Inner VID	Ingress Port	New Outer VID
10	P1	100
20	P2	200
30	P3	300

MAC-based/IP-subnet-based VLAN (1/2)

- 2048 entries shared with IVC
- Configurable source MAC/IPv4 address of search key
- Can ignore ingress VLAN filtering
- Forward action: forward, trap, drop
- VID_VALID = 1 if packet is VLAN-tagged and VID != 0 (not 1p priority tag)

Search key

1 bit	1 bit	1 bit	48/32 bits	6 bits
Valid	OVID_VALID	IVID_VALID	MAC/IP Address	Port/Trunk ID
	OVID Valid Mask	IVID Valid Mask	MAC/IP Address Mask	Port/Trunk ID Mask

Translation field

Action	Value
VID assign - Assign VID with New VID	New VID
Priority assign - Assign to tag priority with New Priority	New Priority
Tag status assign	New Tag Status
TPID assign	New TPID

MAC-based/IP-subnet-based VLAN (2/2)

- Per Ingress port Control
  - Enable IP-Subnet-based VLAN Look up
  - Enable MAC-based VLAN Look up

VLAN Forward and Filter – VLAN table

- 4K VLAN entries

MBR 57 bits	Untag MBR 57 bits	MSTI 7 bits	L2 unicast hash key 1 bit	L2 multicast hash key 1 bit	VLAN profile index 4 bits	Group Mask 16 bits	SVL_FID 12 bits

- MBR: VLAN member port. Which is utilized by VLAN ingress /egress filter
- Untag MBR: define the egress packet tag status from the specific port
- MSTI: MSTP instance, forward/drop packet based on MSTI instance
- L2 unicast hash key: select hash key for L2 unicast and broadcast
  - FID=VID (IVL mode)
  - FID=VLAN_CTRL.UC_SVL_FID or VLAN_ENTRY.SVL_FID (SVL mode)
- L2 multicast hash key: select hash key for L2 and IP multicast
  - FID=VID (IVL mode)
  - FID=VLAN_CTRL.MC_SVL_FID or VLAN_ENTRY.SVL_FID (SVL mode)
- VLAN profile index: index to VLAN profile
- Group Mask:
  - Predefined bits for IGMP, MLD, DHCP, DHCP6, ARP_REQ, ARP_REP traffic action
  - User-defined bits for customer's application with ACL search
- SVL_FID: per VLAN specified FID of SVL mode

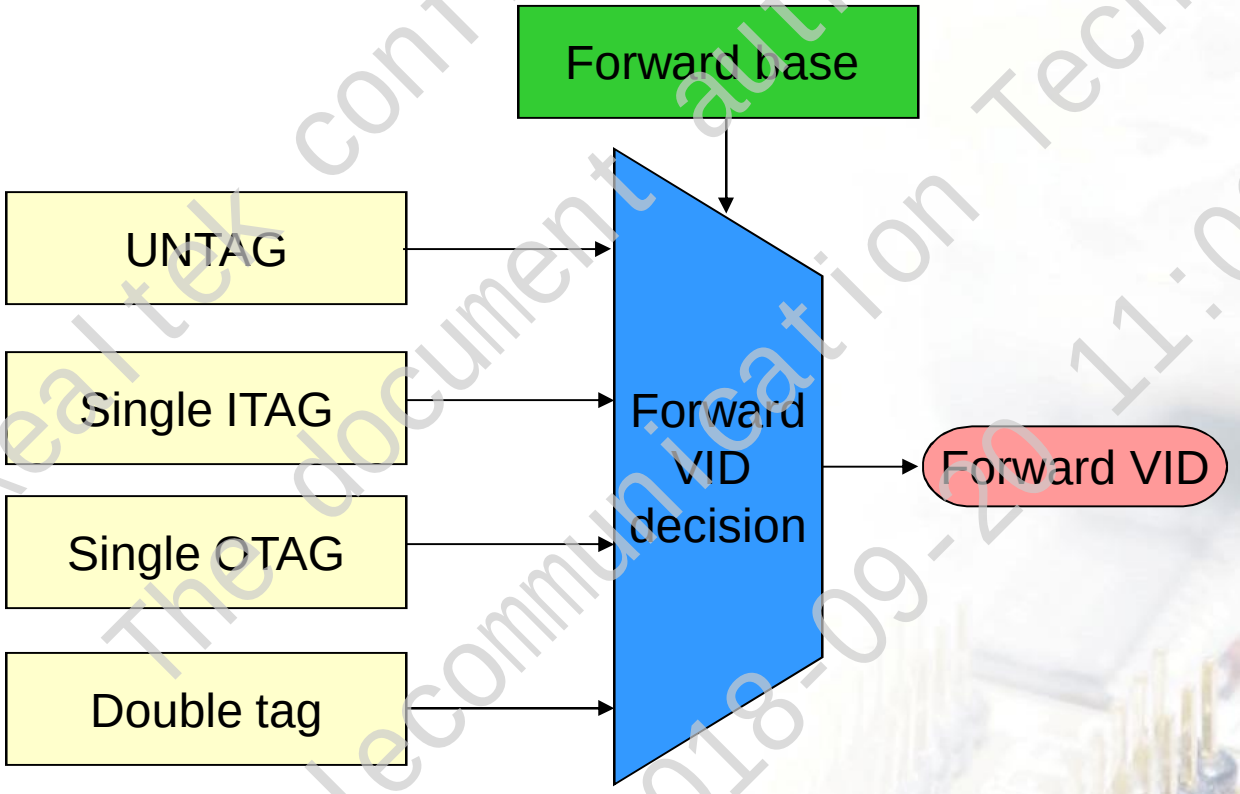
VLAN Forward and Filter – IVL/SVL Mode

Index	Member set	Untagged set	FID	VLAN Base Fwrd
VID[100]	P1	.....	100	.....
VID[200]	P2	.....	100	.....
VID[300]	P3	.....	100	.....
VID[1000]	P25	.....	100	.....

- Forwarding database using FID instead VID
  - Different VLANs can share the same filter database (L2/MAC table)
- Packet tag is still keeping same VID; FID is invisible to User

VLAN Forward and Filter – Forward VID

- Forward base :
  - Per-port and per-tag-status configurable
  - Forward VID Decision
  - Tag status can be changed by IVC or IACL



VLAN Forward and Filter – Ingress Filter

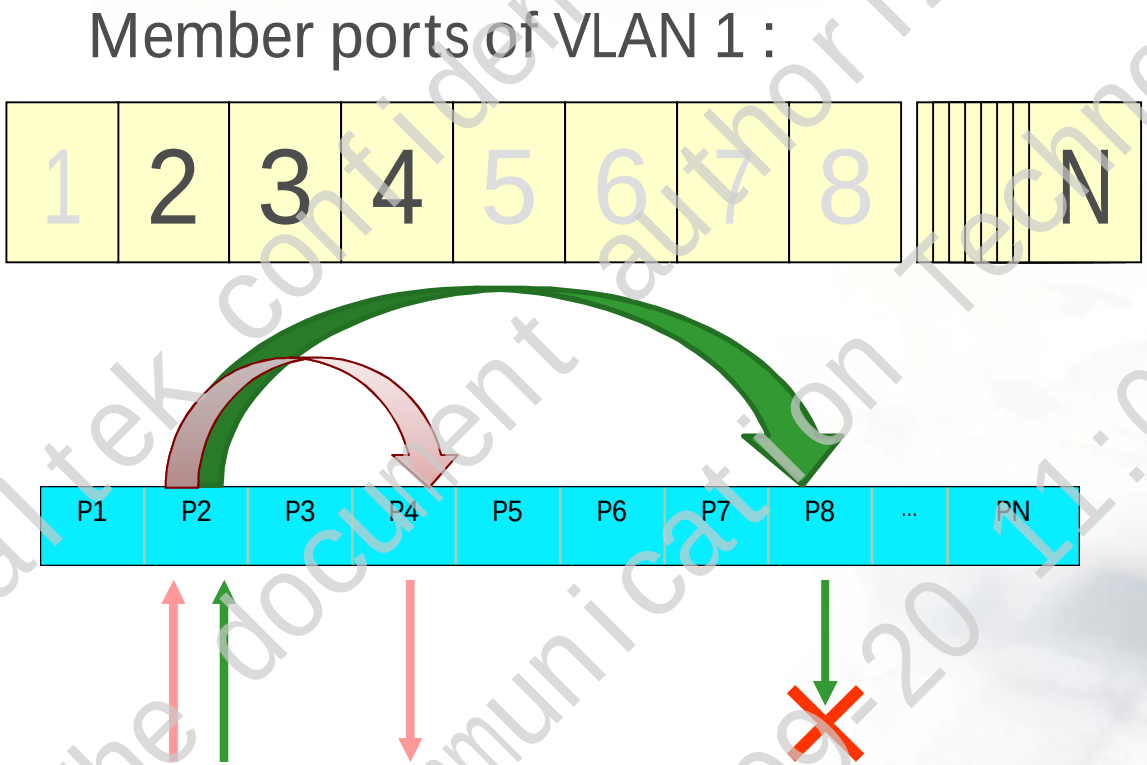
- Per Port Configuration:
  - Filter based on inner or outer VID (forwarding VID decision result)
  - Action of VLAN ingress filter violation
    - Forward (as filter disabled)
    - Drop (as filter enabled)
    - Trap (for further processing)

VLAN Forward and Filter – Egress Filter (1/3)

- Per Port Configuration
  - Check the VLAN membership on egress ports which are Egress Filter enabled
- Global Bypass Configuration for Known L2/IPv4/IPv6 Multicast Packets
  - Bypass Egress VLAN Filter
  - Bypass Spanning Tree Egress Filter

VLAN Forward and Filter – Egress Filter (2/3)

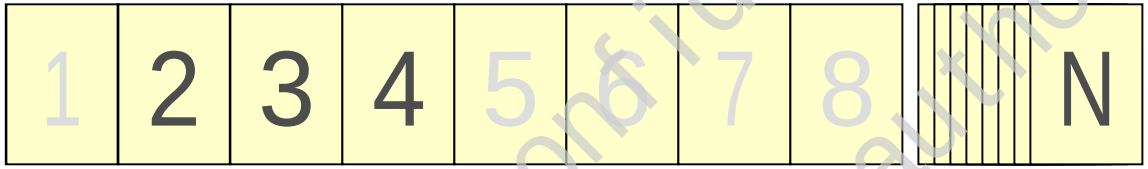
VLAN Egress Filter is enabled



VLAN Forward and Filter – Egress Filter (3/3)

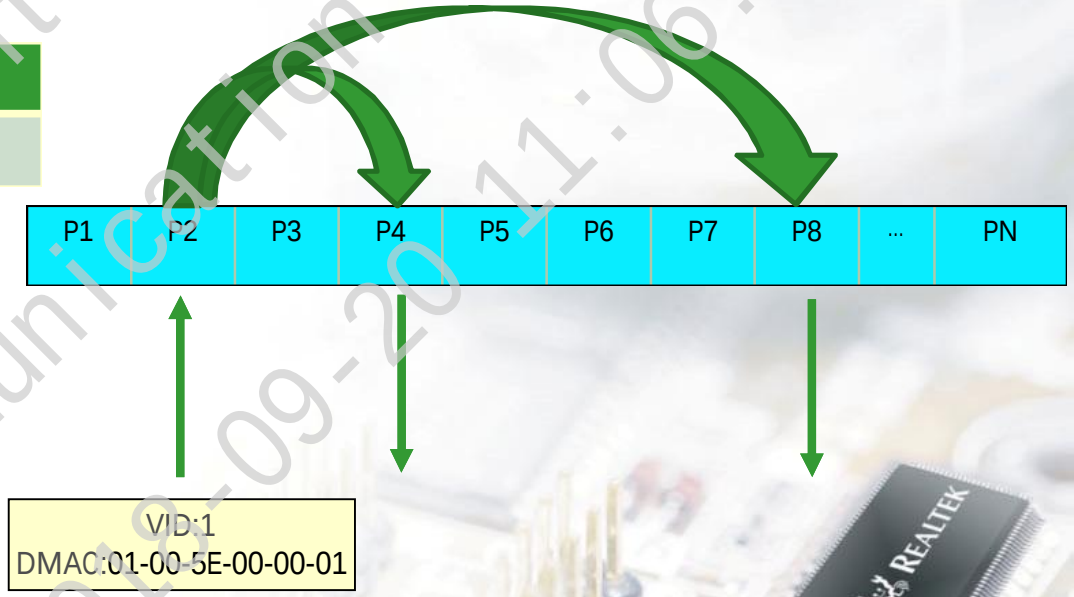
VLAN Egress Filter is enabled
Known Multicast Bypass VLAN Egress Filter is enabled

Member ports of VLAN 1 :



L2 Multicast Table:

VID	DMAC	Portmask
1	01-00-5E-00-00-01	Port4, Port8



VLAN Forward and Filter – VLAN Profile

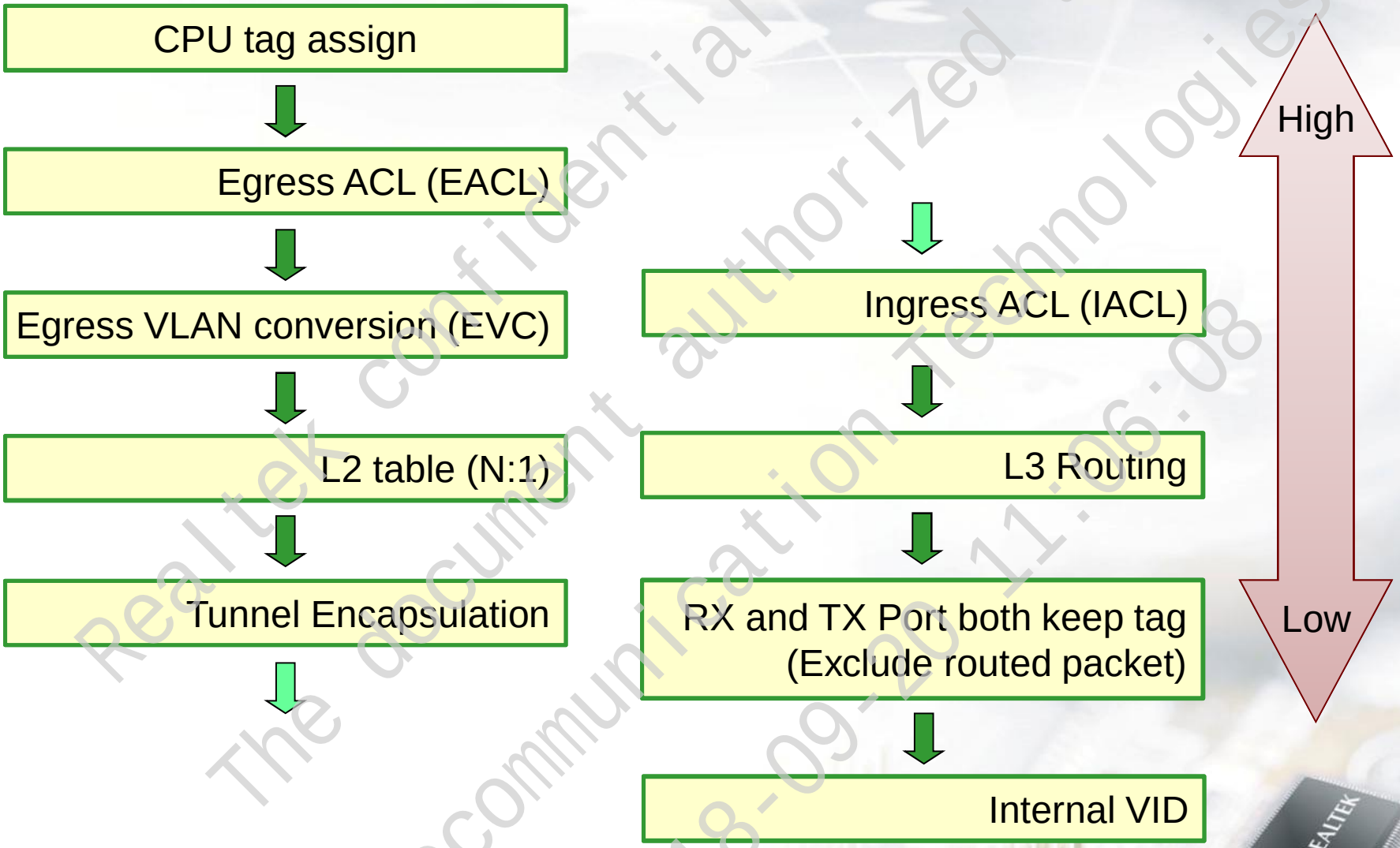
- 16 entries

L2 related	L2 table learning behavior when receives a packet with new SMAC
	Packet forwarding behavior when receives a packet with new SMAC
	L2 bridge mode for IPv4/IPv6 multicast packets
	L2 action of unknown (lookup miss) L2/IPv4/IPv6 multicast packets
	Unknown L2 multicast flooding portmask
	Unknown IPv4 multicast flooding portmask
	Unknown IPv6 multicast flooding portmask

VLAN Forward and Filter – Application Trap

- System-wised Applications Trap Control: forward/trap/copy
  - IGMP Snooping
  - MLD Snooping
  - DHCP
  - DHCPv6
  - ARP Request
  - ARP Reply
- VLAN-based Trap Control (using GROUP_MASK field)
  - 6 predefined bits (also can be redefined by user)
    - GROUP_MASK[2] : IGMP Snooping
    - GROUP_MASK[3] : MLD Snooping
    - GROUP_MASK[4] : DHCP
    - GROUP_MASK[5] : DHCPv6
    - GROUP_MASK[6] : ARP Request
    - GROUP_MASK[7] : ARP Reply

Egress VID Decision



Egress VLAN Conversion (1/4)

- 1024 entries
- Configurable internal VID (or internal VID range) of search key
- Per entry has a hit indication flag
- Choose lowest index entry if multiple entries hit

Search key

1 bit	12 bits	32 bits	6 bits	3 bits
Valid	Internal VID	Internal VID Range	Egress Port/trunk	Internal Tag Priority
	Internal VID Mask	Internal VID Range Mask	Egress Port mask	Internal Tag Priority Mask

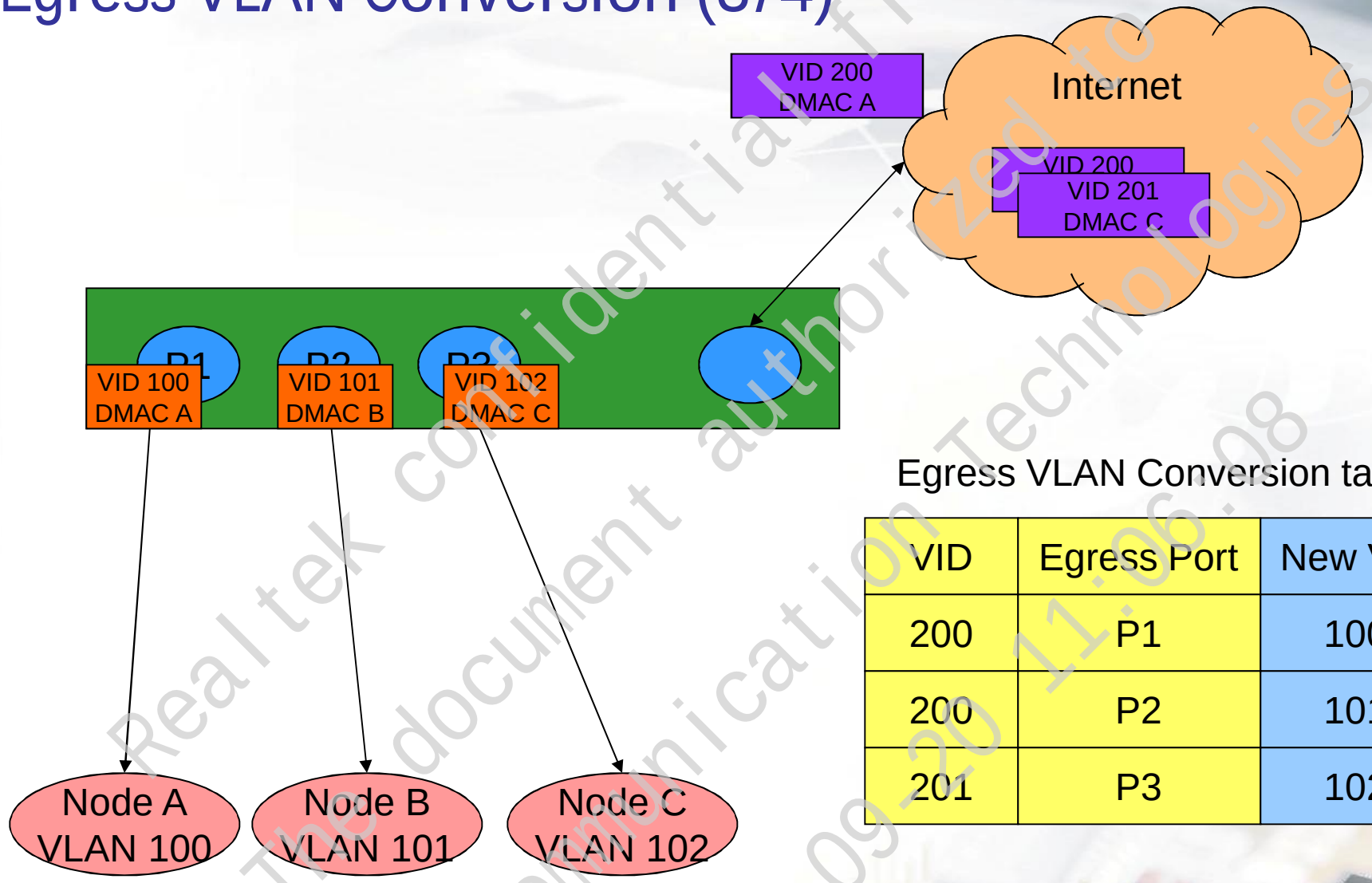
Translation field

Action		Value
VID assign	Assign VID with New VID	New VID
	Shift VID with New VID	
	Copy from internal I-VID/O-VID	
Priority assign	Assign priority with New Priority	New Priority
	Copy from internal inner/outer priority	
Tag Status assign		New Tag Status
TPID assign		New TPID

Egress VLAN Conversion (2/4)

- Per Egress Port VLAN Conversion Control
  - Enable EVC VLAN look up for specified port
  - EVC lookup miss action for internal inner/outer tagged traffic: Forward, Drop
  - Selection of VID range check set
- System supports 4 VID range check tables
  - Each table has 32 dedicated O-VID/I-VID range check entries
    - VID range lower bound
    - VID range upper bound
    - Range check data type
      - internal I-VID range check
      - internal O-VID range check

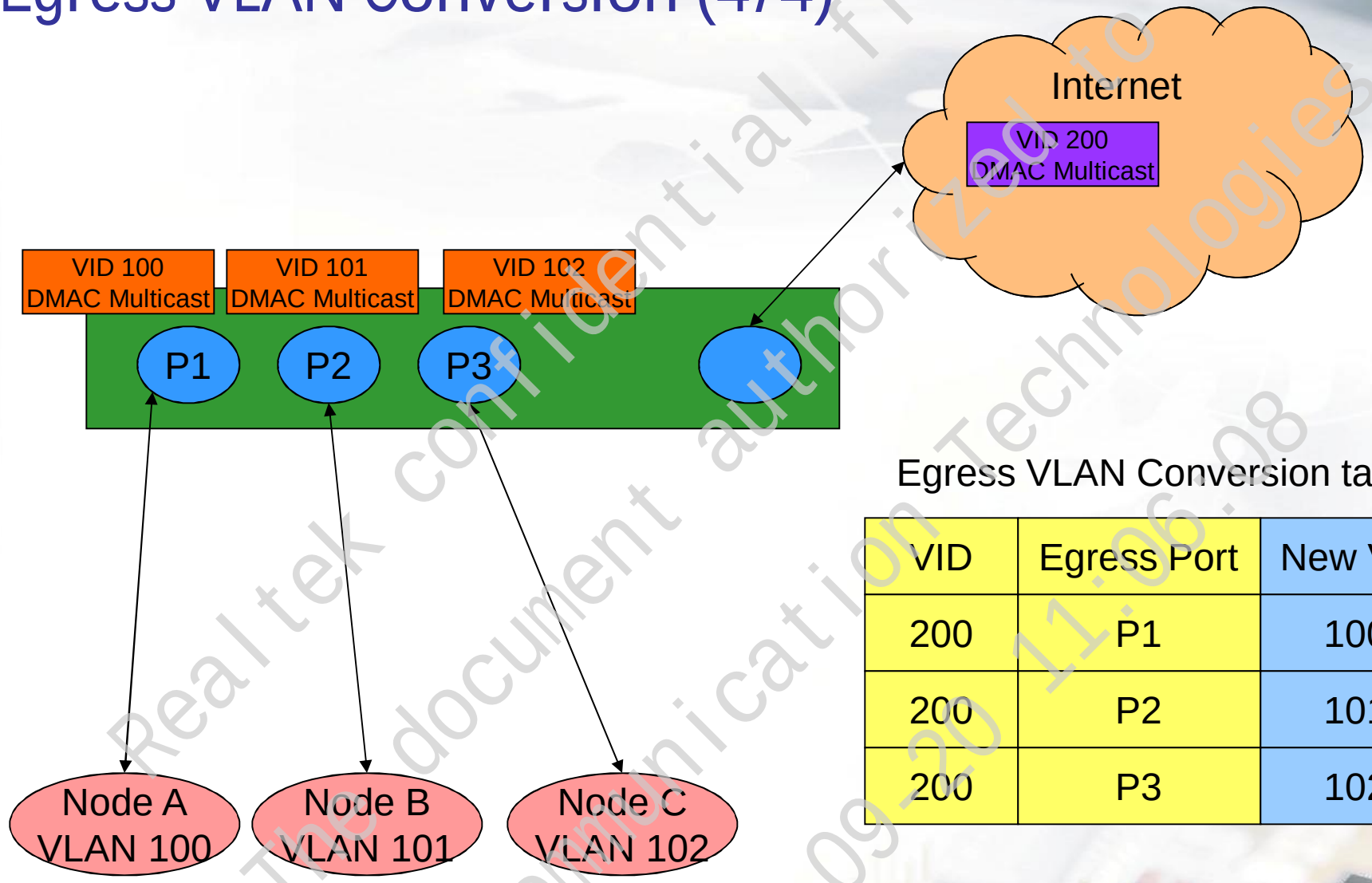
Egress VLAN Conversion (3/4)



Egress VLAN Conversion table

VID	Egress Port	New VID
200	P1	100
200	P2	101
201	P3	102

Egress VLAN Conversion (4/4)



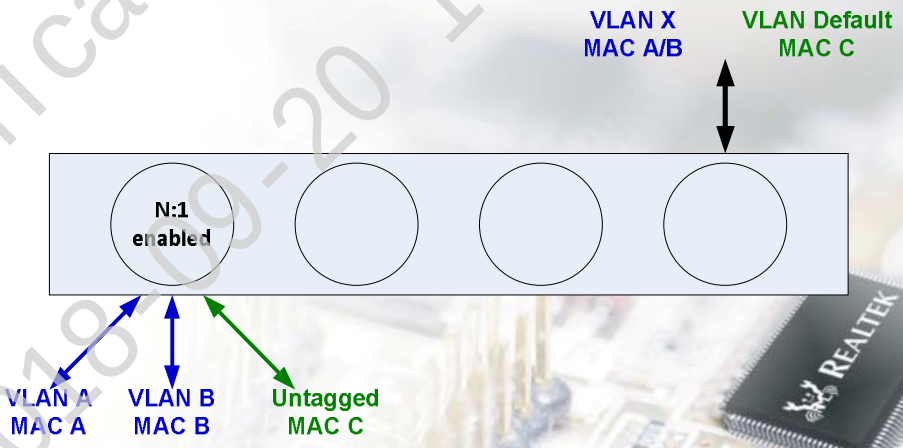
Egress VLAN Conversion table

VID	Egress Port	New VID
200	P1	100
200	P2	101
200	P3	102

N:1 VLAN Translation

- For a downstream traffic with same forwarding VLAN
- Per individual/trunk port enable
  - VID will be learnt with source MAC address on ingress
  - Replace VID retrieve from L2 table by destination MAC address on egress
  - Selection of inner or outer VID for learning/replace
- L2 Learning VID/Priority
  - VLAN-tagged packet : learning original VID
  - Priority-tagged packet : per port select 0 or Rx PVID (Port-based VID) to learn
  - Untagged packet : learning VID with 0

Learnt Value	Status	Priority	VLAN ID
VLAN-tag	1	PCP	original VID
Priority-tag	1	PCP	0 / PVID
Untagged	0	0	0



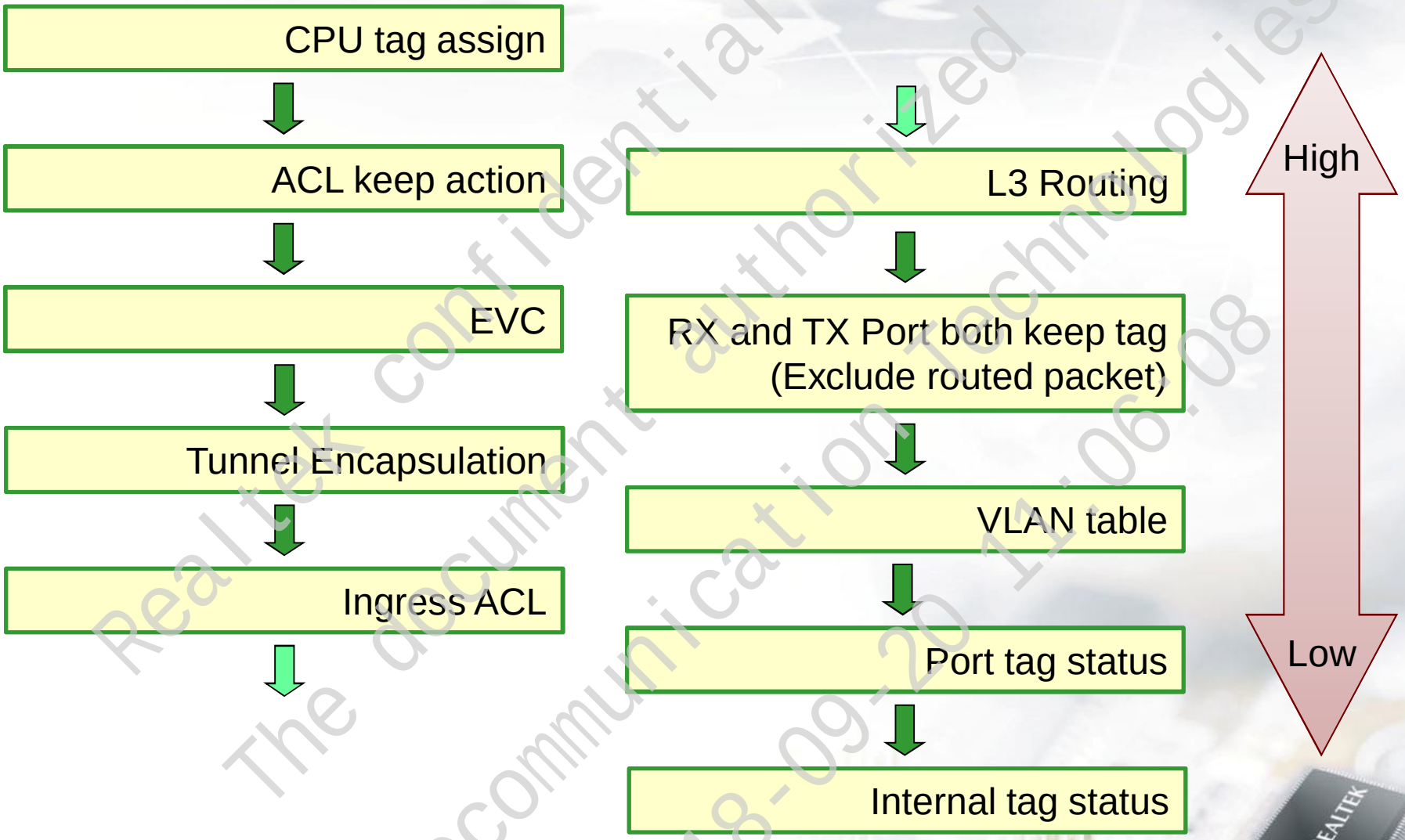
Egress VLAN Tagging – Port Tag Keep Control

- Per Ingress/Egress port configuration
  - Per Ingress Port configuration (Inner tag and Outer tag)
    - Normal mode , follow VLAN untag set and egress port tag status decision
    - Keep tag content
  - Per Egress Port configuration (Inner tag and Outer tag)
    - Normal mode , follow VLAN untag set and egress port tag status decision
    - Keep tag content
  - If both Rx/Tx ports of the traffic are enabled to keep-tag, the keep action will be taken. (unless there is another higher priority action)

Egress VLAN Tagging – Port Tag Status

- Per Port Configuration
  - Option for non-forwarding VLAN
  - Inner configuration
    - Untagged / Tagged / Priority-tag
    - Follow the internal status
  - Outer configuration
    - Untagged / Tagged / Priority-tag
    - Follow the internal status

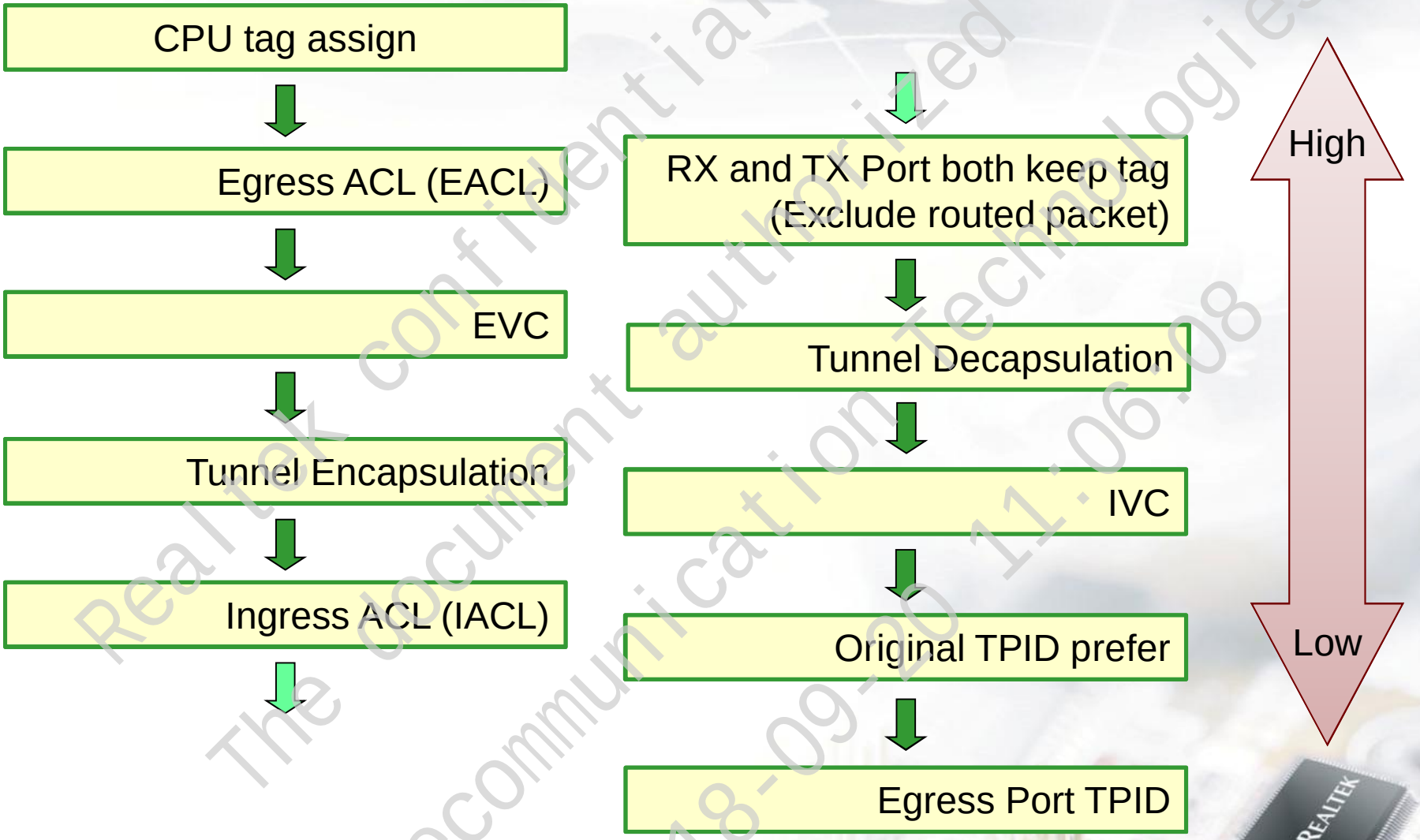
Egress VLAN Tagging – Tag Status Decision



Egress VLAN Tagging – Port Egress TPID

- Per-Port Configuration
- Inner Tag TPID Control
  - TPID Index: Select one from 4 TPID values
  - Option to prefer original TPID (if it is inner-VLAN tagged on ingress)
  - Default TPID for original inner untagged packet
- Outer Tag TPID Control
  - TPID Index: Select one from 4 TPID values
  - Option to prefer original TPID (if it is outer-VLAN tagged on ingress)
  - Default TPID for original outer untagged packet

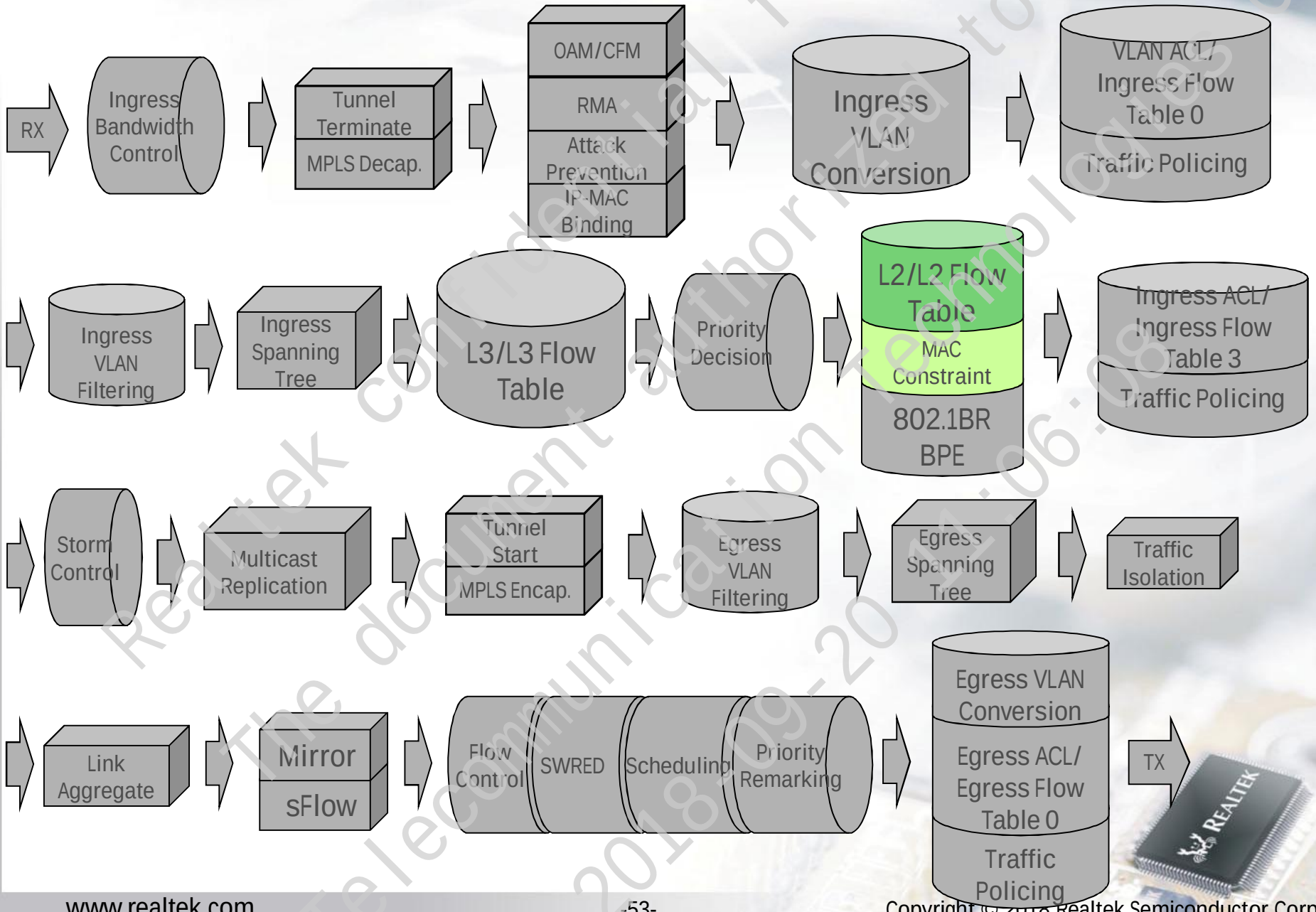
Egress VLAN Tagging – Egress TPID Decision



Realtek Semiconductor Corp.

L2 MAC Address Lookup

Packet Processing Pipeline



Feature Summary

- L2 Table
  - 32K entries with 2-left 4-way hash
  - 2 hash algorithm
  - Aging time: 0 ~ 1.67M seconds
  - SA/DA MAC Blocking
  - Suspend Learning
  - Port Move Action
  - Hash Full Action
  - MAC Limit
  - Flush MAC by port and/or VLAN
  - Table Change Notification

- Multicast Forwarding Port-mask Table
  - 4K multicast forwarding port-mask
  - Indexed by L2 multicast entry

L2 Table Entry Format

	Hash Key		Table Content										
L2 Unicast	FID/ VID (12)	DMAC (48)	Is_trk (1)	Unit (4) / Trunk (10)	Port (6)	Age (3)	SABlk (1)	DABlk (1)	Static (1)	Sus (1)	NH (1)	Agg_PRI (3)	Agg_VID (12)
L2 Multicast	FID/ VID (12)	DMAC (48)	Index (12)								NH (1)		

- 32K-entry table shared by Unicast and multicast entry
- Multicast entry index to 4K multicast port-mask table

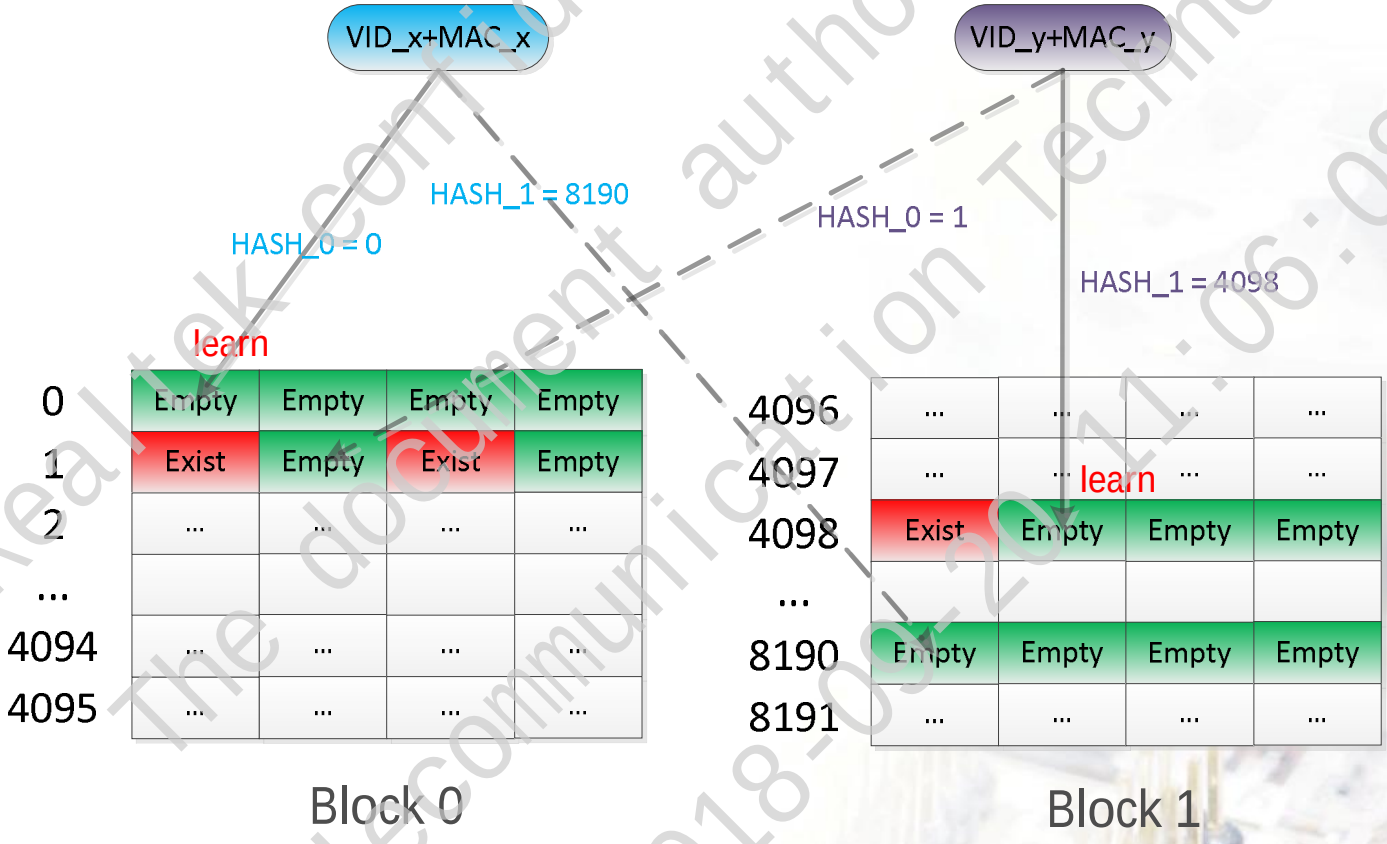
L2 Unicast Entry Format

Hash Key		Table Content										
FID/ VID (12)	DMAC (48)	Is_trk (1)	Unit (4)	Port (6)	Age (3)	SABlk (1)	DABlk (1)	Static (1)	Susp (1)	NH (1)	Agg_PRI (3)	Agg_VID (12)
			Trunk (10)									

- SA Block: Source MAC filtering
- DA Block: Destination MAC filtering
- Static: No age out & no port move
- Suspend:
  - Only trap new learn SA packet once and mark as “temporary entry” (suspend = 1)
  - DA lookup will skip suspend entry since it is considered a temporary entry
- NH (Next-hop) : for L3 routing
- Agg_PRI: for replacing priority field in N:1 VLAN translation
- Agg_VID: for replacing VID field in N:1 VLAN translation

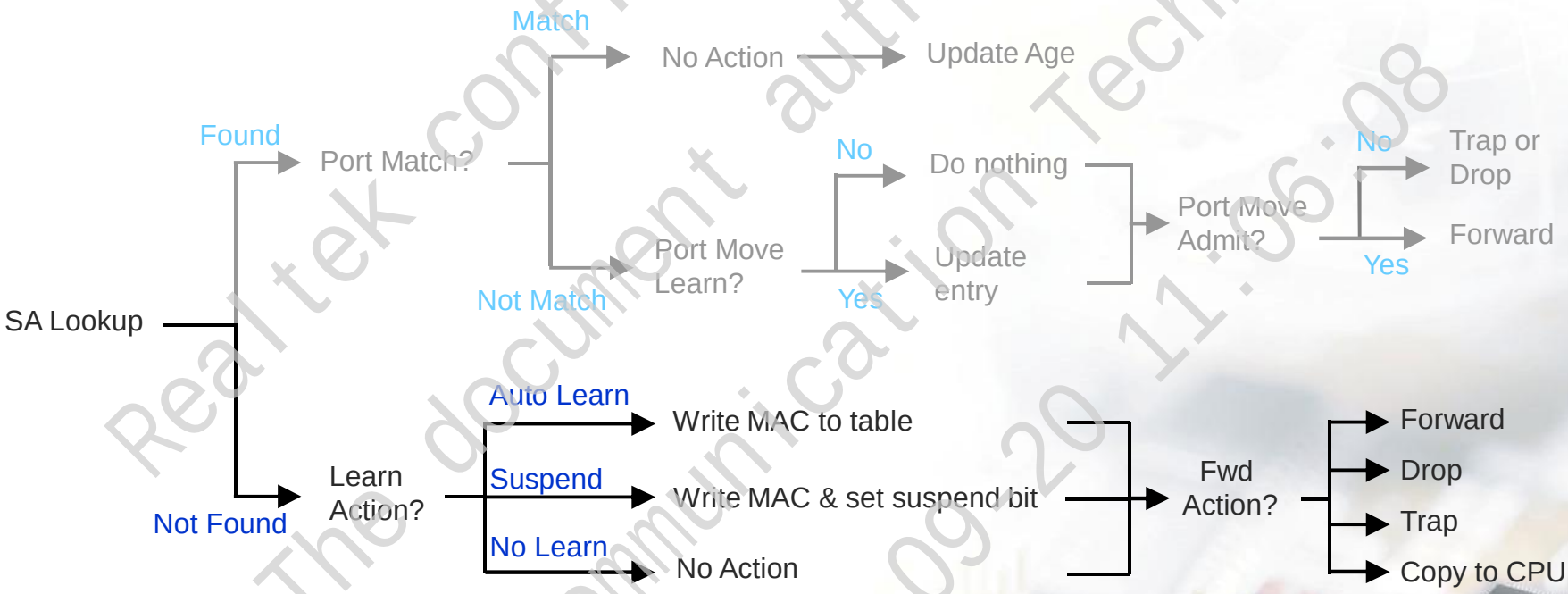
2-left 4-way Hash

- SA lookup/learning and DA lookup through 2-left 4-way Hash
- Each block can specify one of the two hash algo
- Higher learning rate than traditional single 4-way hash
- The following figure shows MAC learning position



L2 Unicast Learning

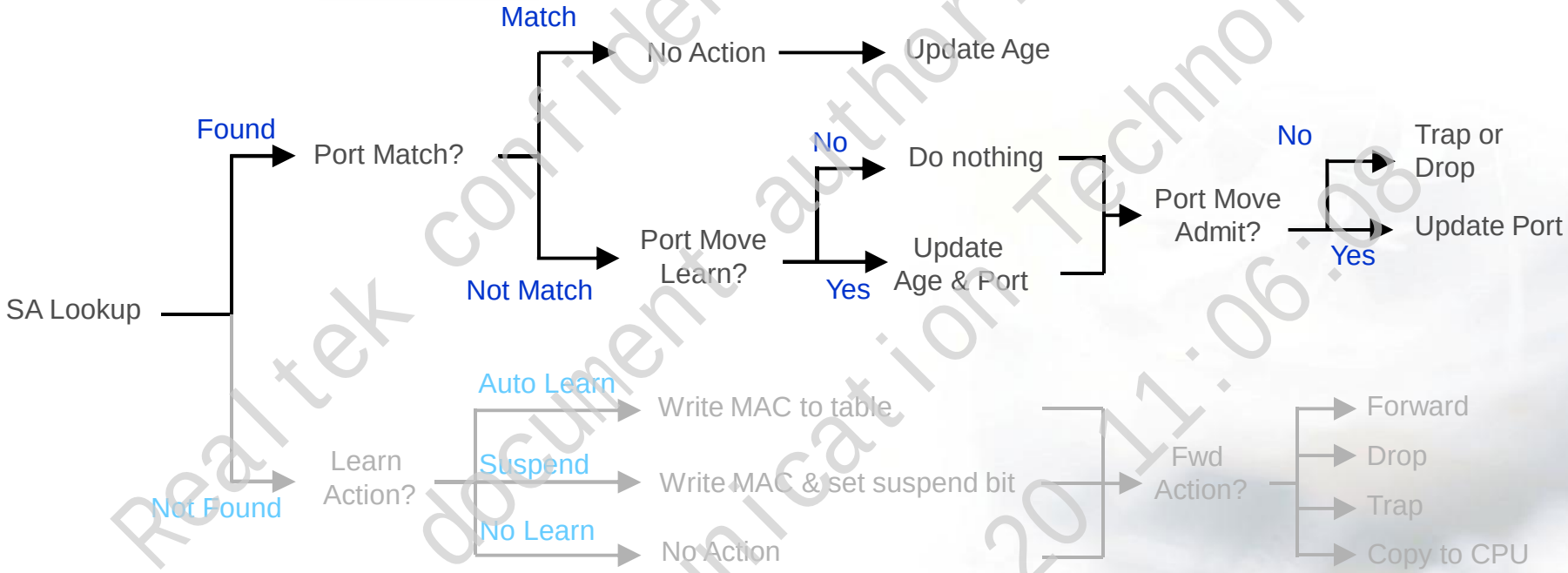
- Port-based and VLAN-based source MAC learning control
  - New SMAC Learn Action: Auto Learn / Learn as Suspend / No Learn
  - New SMAC Forward Action: Forward / Drop / Trap / Copy to CPU



L2 MAC Address Port Movement (1/2)

- Static Entry Port Move
  - Only for static entry
  - Global configuration
  - Action: Forward / Drop / Trap / Copy to CPU
  - Learn: Yes (only update AGE, AGG_PRI, AGG_VID)/ No
- Dynamic Entry Port Move
  - Only for dynamic entry
  - Per Port configuration
    - Reference ingress port configuration, not L2-entry learnt port
  - Action: Forward / Drop / Trap / Copy to CPU
  - Learn: Yes / No
- Port Move Forbidden
  - Only for dynamic entry
  - Per Port configuration
    - Reference ingress port & L2-entry learnt port configuration
  - Action: Forward / Drop / Trap / Copy to CPU
  - Learn: No

L2 MAC Address Port Movement (2/2)



Misc. Configuration

- Exception SA
  - Multicast/Broadcast SA
    - Action: Forward / Drop / Trap
    - Learn: No
  - Zero SA
    - Action: Forward / Drop / Trap
    - Learn: Yes / No

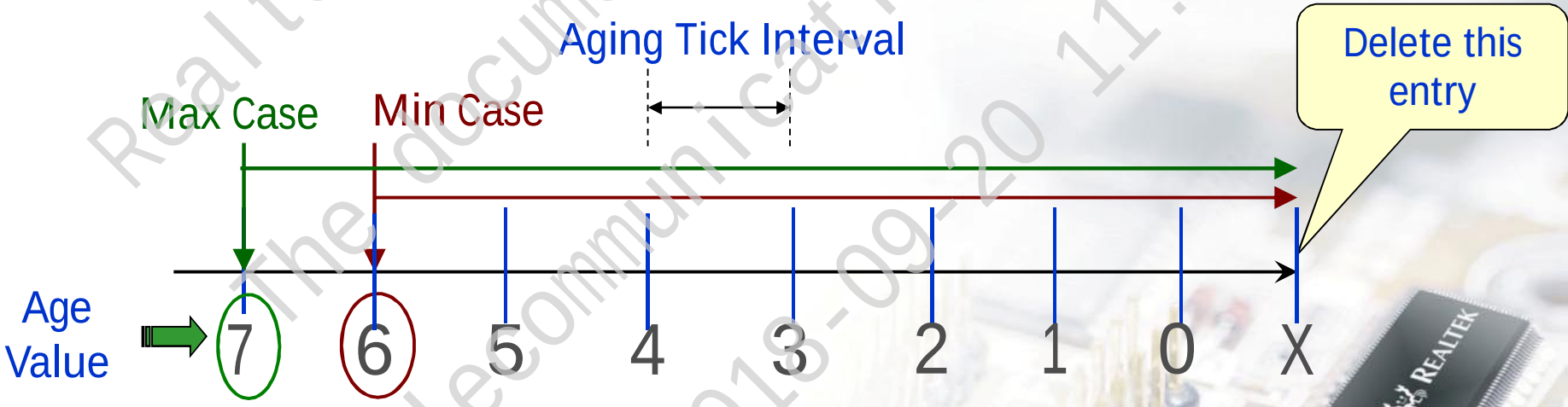
- Hash Full
  - Action: Forward / Drop / Trap
  - Learn: Yes / No

DA Lookup Miss Action

- Unicast
  - Per Port Configuration: Forward / Drop / Trap / Copy
  - Global Forward port-mask
- Broadcast
  - Action: Forward
  - Global configure forward port-mask
- L2 Multicast
  - VLAN Profile Configuration : Forward / Drop / Trap / Copy
  - Configuration: Forward port-mask

MAC Address Aging

- Aging time ranged from 0 second to 1.67M seconds
- Age: Each MAC entry 3 bits
  - Age value decrement 1 for every aging tick
  - Suspend Entry can be fast age out, by configuring suspend Max Age from 0 to 7,
- Aging Unit: 21 bits
  - Age tick interval: Aging Unit * 0.1 second (0 means no aging)
- Default Aging Unit: 375
  - 375 * 0.1 * 7 = 262.5 (seconds) → Min time to age out
  - 375 * 0.1 * 8 = 300 (seconds) → Max time to age out



MAC Address Flush

- Global setting for link down port auto flush
  - Keep static MAC, SA Block MAC, and DA Block MAC entries
  - Trunk port supported. Only takes effect when last member port link down

- S/W trigger quickly unicast entry flush/ Port replace
  - Configure port ID(optional)
  - Configure FID (optional)
  - Choose entry type
    - Dynamic unicast entry (include suspend entry)
    - All unicast entry
  - Choose action
    - Flush entry
    - Replace Port (new Port configurable)
    - Clear AGG_VID and AGG_PRI
    - Clear Nexthop bit

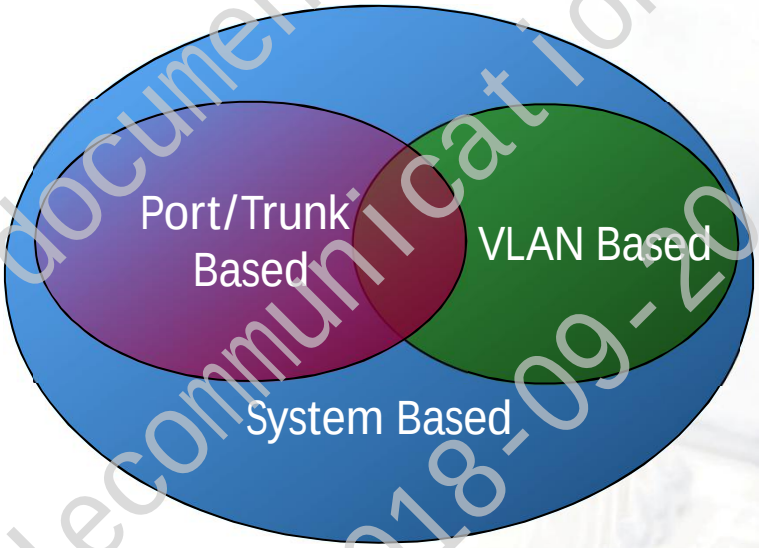
Realtek Semiconductor Corp.

MAC Constraint

Overview

- System-based MAC Constraint
- Port/Trunk-based MAC Constraint
- VLAN-based MAC Constraint (up to 32 groups)

- When L2 learnt constraint exceeds,
  - New SA wouldn't be learnt
  - Action would be applied
  - Static, SABlk, DABlk, and Suspend entry will NOT be counted in.
- System, Port/Trunk, and VLAN-based can work at the same time.



MAC Constraint Common Configuration

- Maximum MAC constraint number setting
  - Max learned 32K
  - 0xFFFF (unlimited)
  - 0x0 (never learning)

- Current Counter maintain by ASIC
- MAC Constraint action
  - Forward / Drop / Trap / Copy
- When multiple MAC Constraint action take effect at the same time:
  - Drop > Trap > Copy > Forward

VLAN-Based MAC Constraint

- 32 entries
- Match key
  - VID
  - Port/Trunk
    - If configured as 0x3f, means port/trunk is not compared

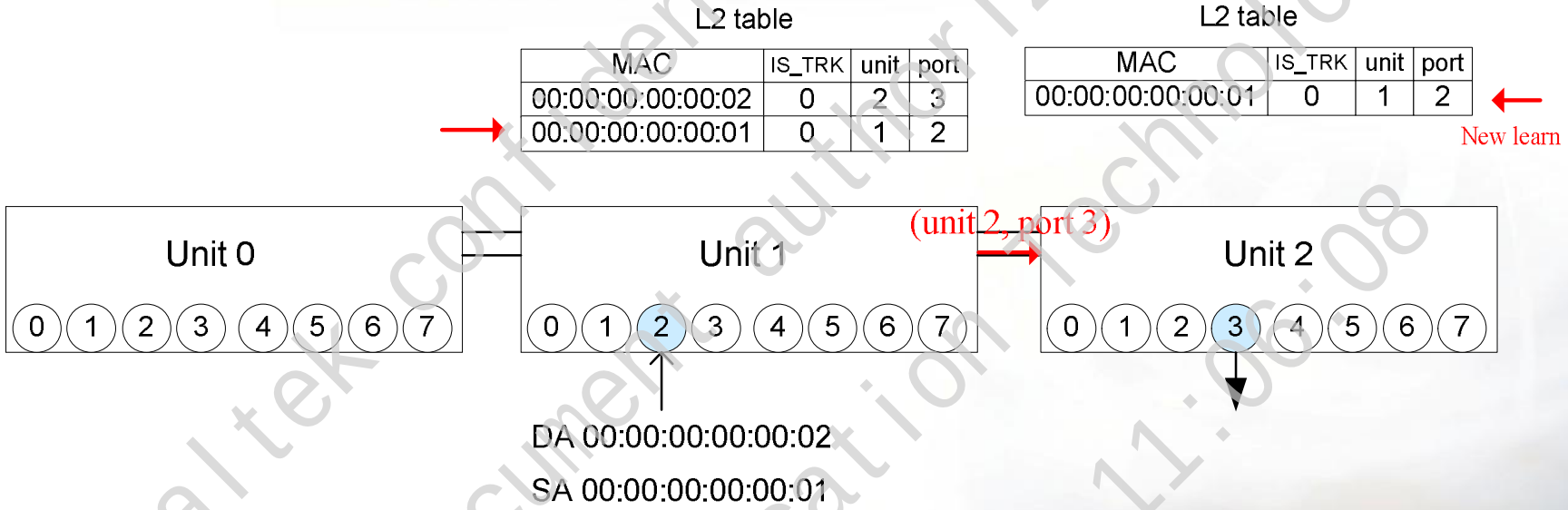
VID	Port/Trunk	MAX Number	Current Counter

Realtek Semiconductor Corp.

L2 Behavior in Stacking

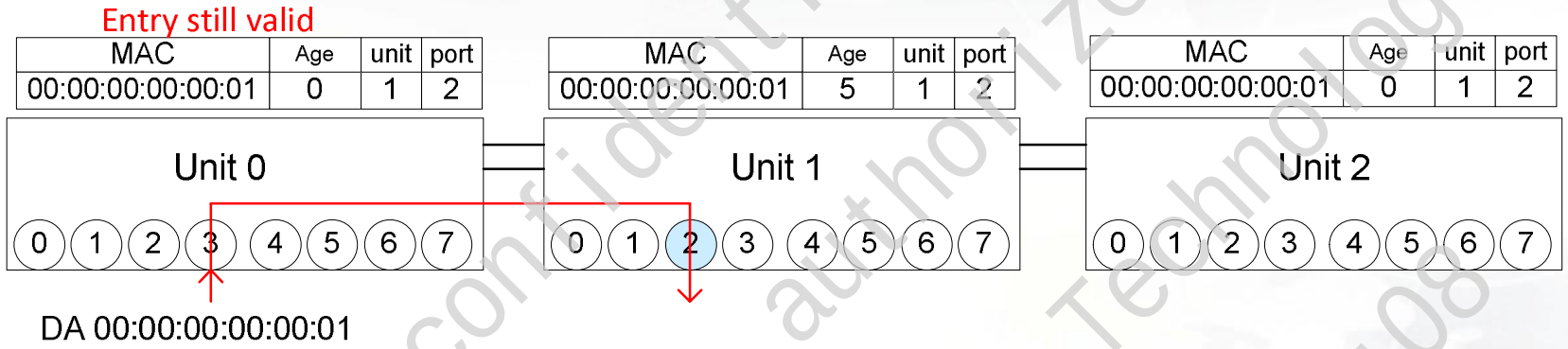
Auto Learning in Stacking Devices

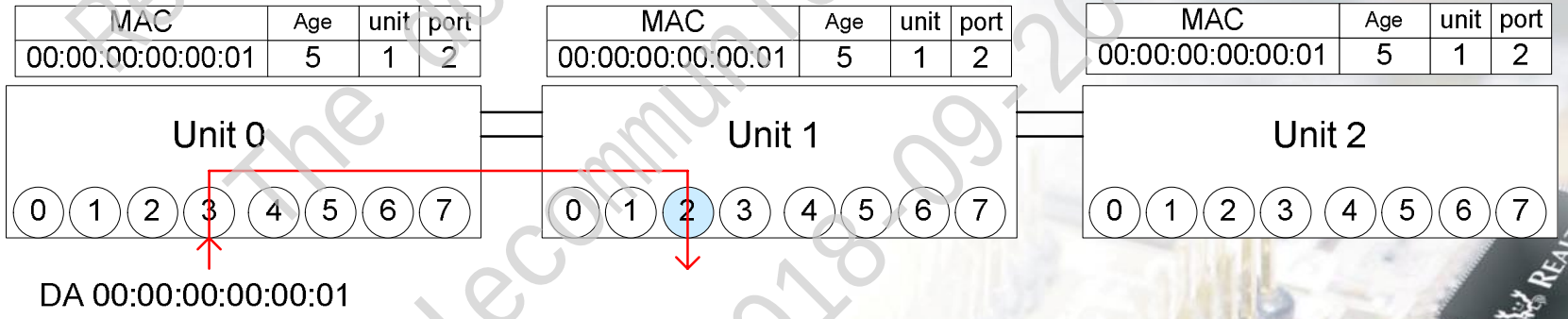
- If configured, SA would be learnt among devices that packet ran through



Aging in Stacking Devices

- For ease of table synchronization, remote L2 entry could be configured to be alive after “Age” field in L2 reaches 0.





Realtek Semiconductor Corp.

L2-Entry Notification

Overview

- To notify CPU that L2 table has been modified
- 2 modes:
  - NIC Mode (internal CPU)
  - Packet Mode (internal/external CPU, cascade, and stacking)
- 4 event types:
  - Add (New-Learn)
  - Delete (Age Out)
  - Modify (Port-Move, Agg_VID , Agg_PRI and ECID change)
  - Hash Full
- Global Enable/Disable configuration:
  - Static entry
  - SA-block & DA-block Entry
  - Dynamic Entry
  - Suspend Entry
  - Hash Full

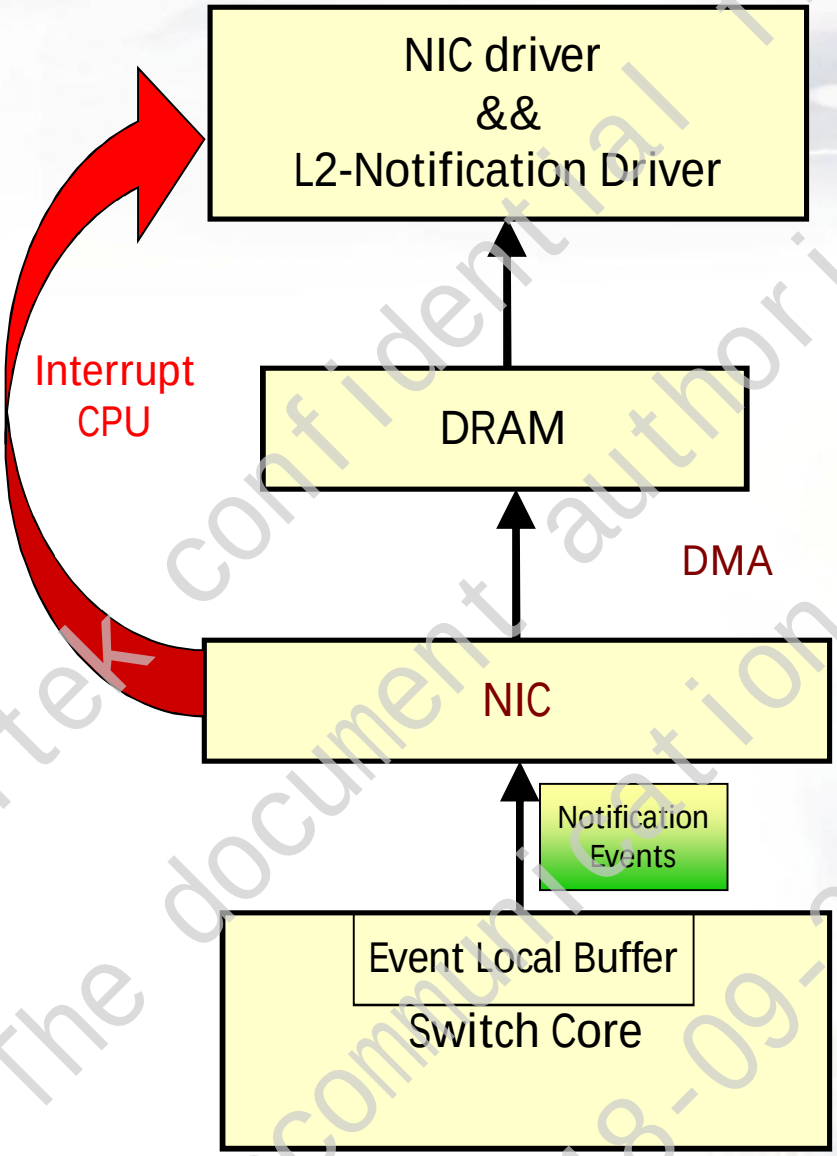
Basic Mechanism

- Each L2-Entry changed(new Learn, age...) will produce an event and stored in a local buffer

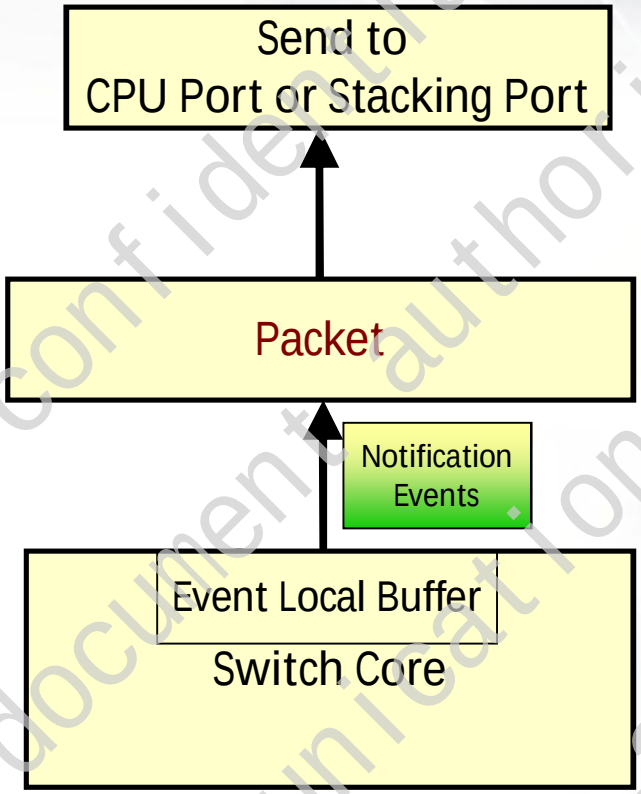
- Events will be united as a message, if
  - Stored event number >= threshold(default is 8)
  - Stored event number < threshold, but time out(default is 10us)

- Message is transferred to
  - Internal CPU, by NIC Mode
  - Internal or external CPU, by Packet Mode
  - Master device in a Stacking System, by Packet Mode

NIC Mode

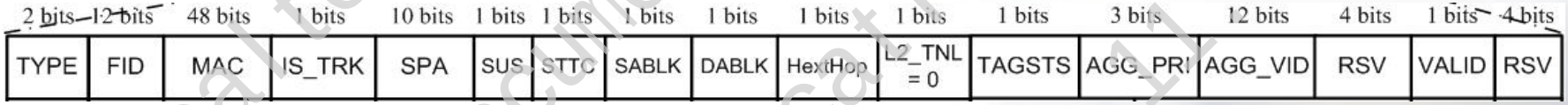


Packet Mode



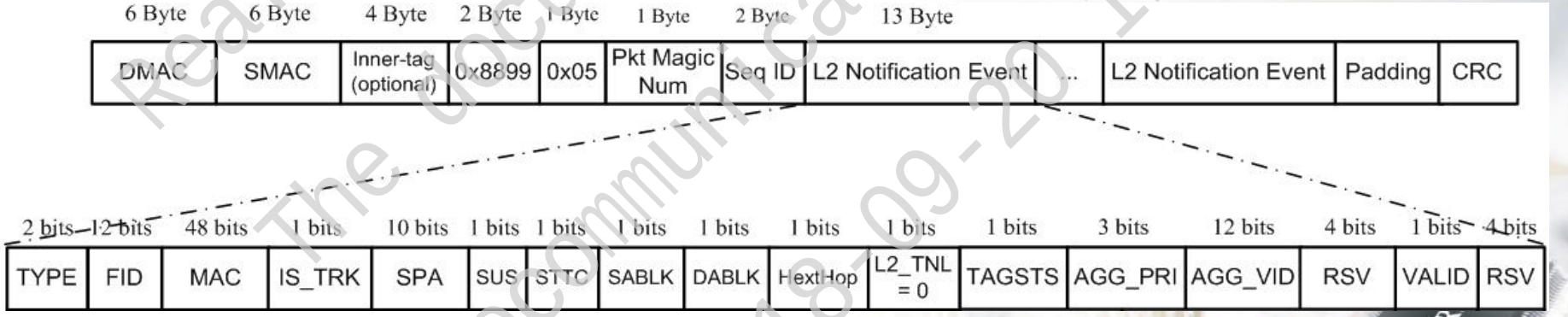
Event Format

- L2-Notification Event format
  - 13 Bytes
  - Valid -- indicate the event is valid or not
  - Type – indicate the event type
    - 0: Age or Link-down flush
    - 1: new Learning
    - 2: Port-Move or Aggregation VID/PRI or ECID changed
    - 3: Hash Full
  - Other fields (FID, MAC…) are L2-entry fields



Packet Format

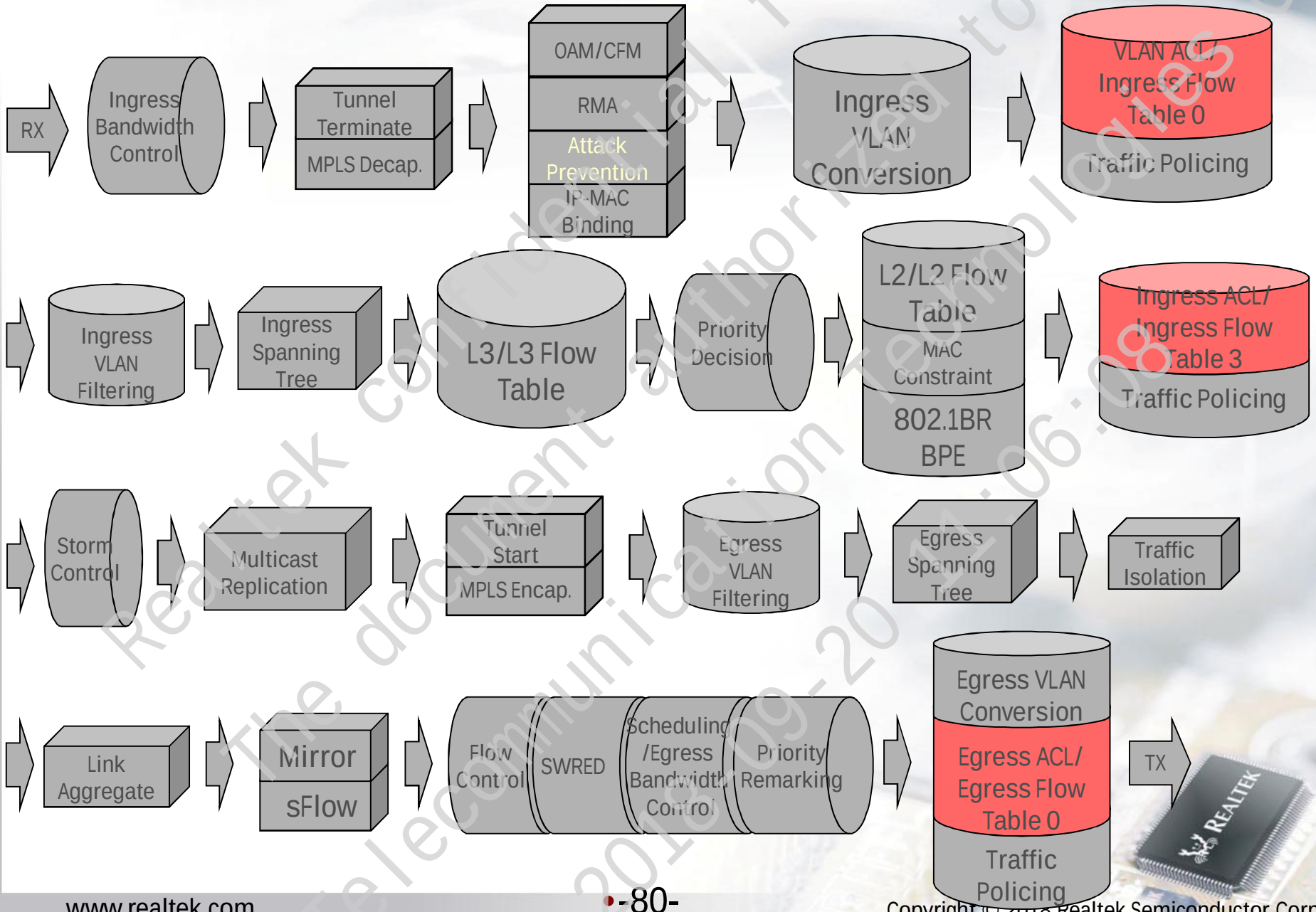
- DMAC & SMAC
  - Both are configurable
- Packet Magic Number
  - Used to identify the effectiveness of a received notification packet
  - Packet is valid only if the Magic Number in packet equals the configured Magic Number
  - When reset L2-Notification, software should configure a new Magic Number
- Sequence ID
  - Sequence ID will increase from 0 by each packet to detect any packet loss
- Padding
  - All zero padding
- Max packet length is configurable for aggregating events in a packet



Realtek Semiconductor Corp.

ACL

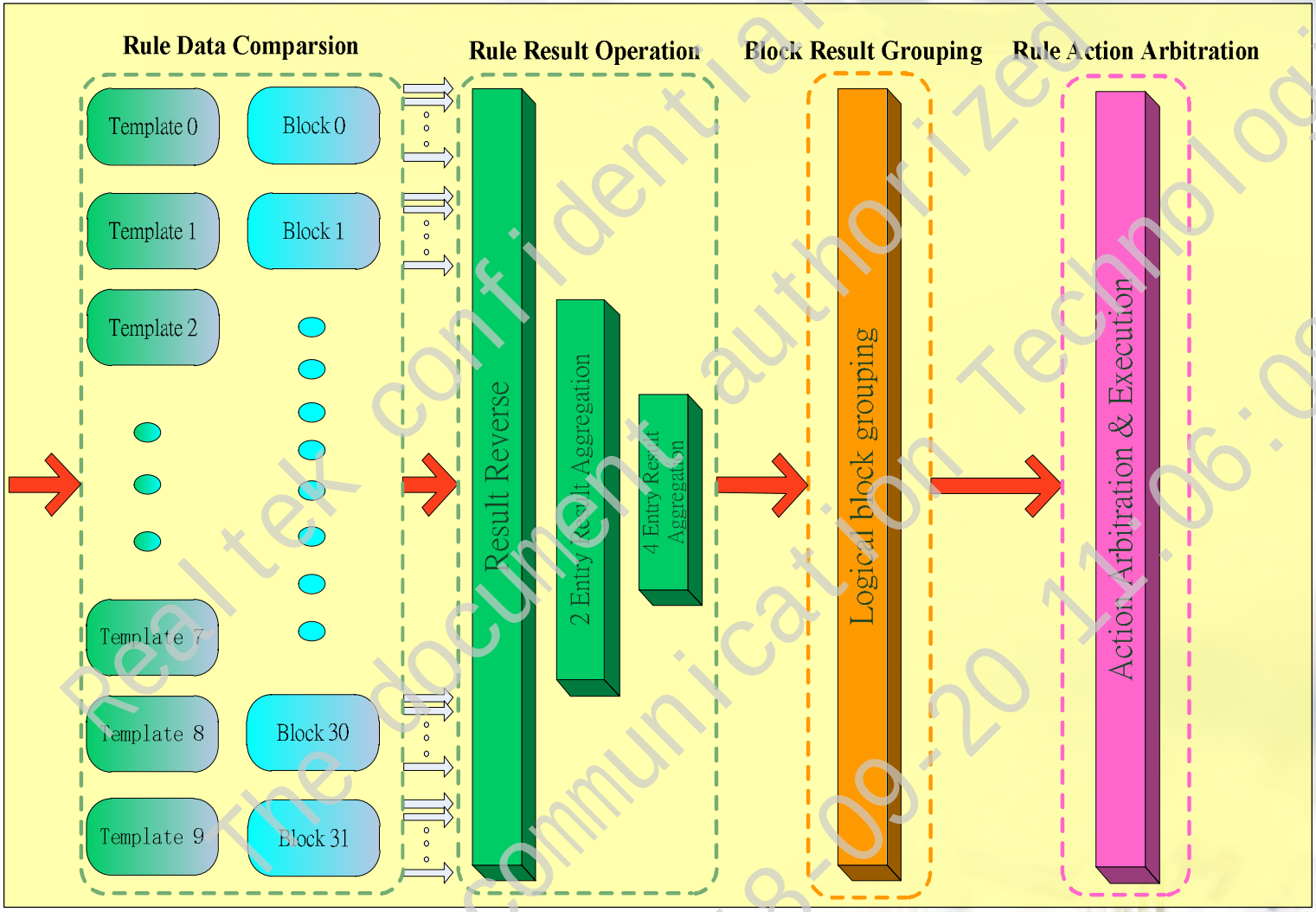
Packet Processing Pipeline



Access Control List Overview

- 4k ACL entries
  - 240 bits key (224 bits definable + 16 bits fixed) with 240 bits mask
  - up to 912 (224*4 + 16) bits through aggregate the entry results
  - Multiple actions support
  - Aggregation and result reverse operations
- 3 lookup phase
  - VLAN, Ingress and Egress ACL
- 32 ACL blocks
  - Per block specify 1 of 3 lookup phases
  - Group ID and logic ID for redefine priority of block
  - 128 entries per ACL block
- 10 templates
  - 5 pre-defined templates
  - 5 user-configurable templates
  - Per block can specify 2 different templates in VACL and IACL
  - Per block can specify 1 different templates in EACL
- Multiple entries hit in different block group
  - All non-conflict actions among hit entries are executed
- Clear and move bulk entries operation

ACL Architecture Preview

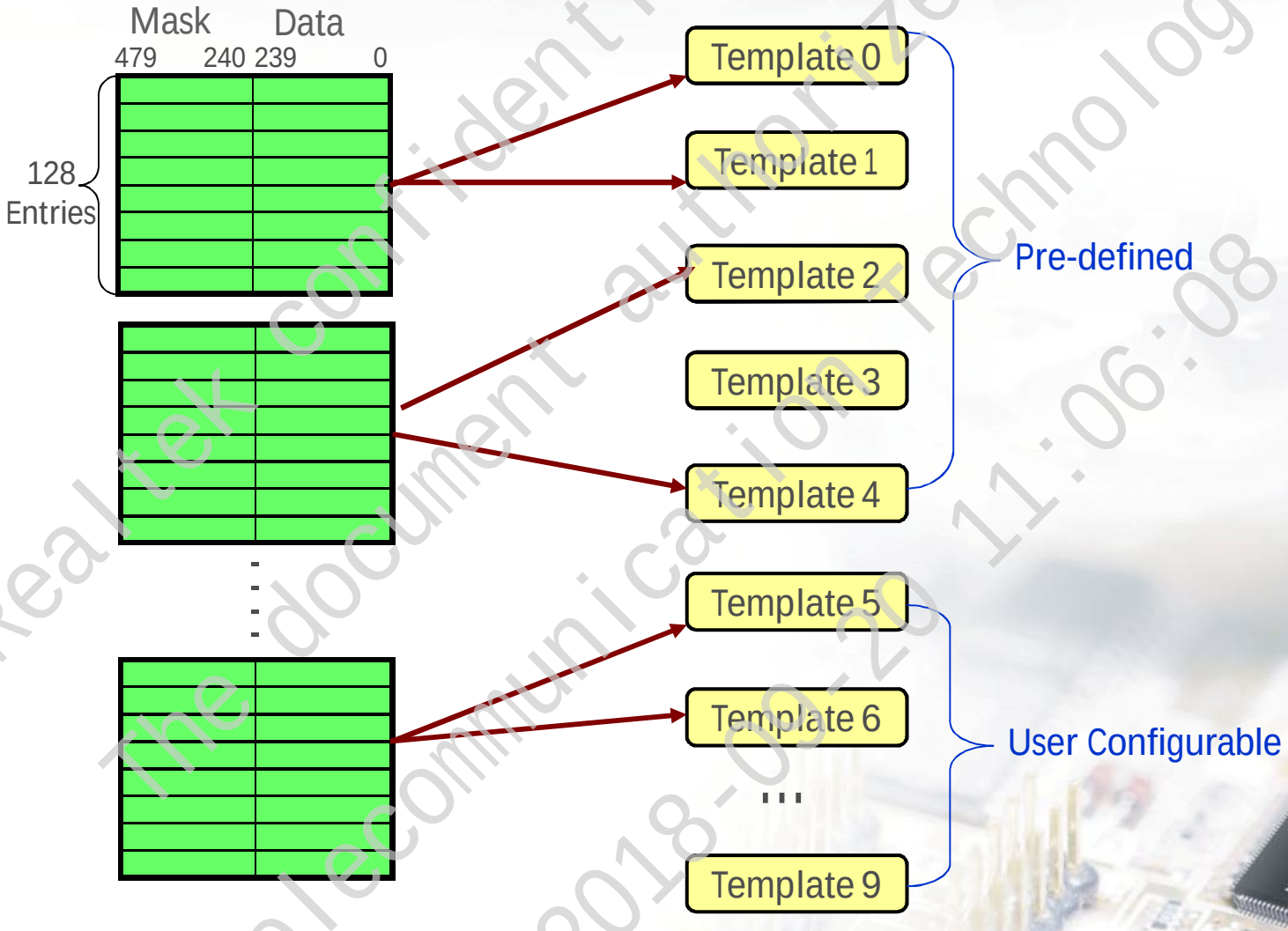


Template

- Indicate what the 224 bits represent in ACL entry
- Template select of each ACL block
  - 2 templates can be selected for VACL and IACL
  - 1 template can be selected for EACL
- 10 templates: 5 pre-defined, 5 user defined
- 14 fields of each template
  - 16 bits per field
  - Each field can be defined by user in user defined templates
- 61 different template field types
  - field type is comprehensive through L2/L3/L4,
    such as SMAC, DMAC, ITag, OTag, EtherType, SIP, DIP,
    Range Check, Field Selector, TCP/UDP Port,
    VLAN Group Mask, Metadata, Lookup Result, Forward/Drop Reason
    ... ...

Template Mapping

- Per block can specify 2 different templates in VACL and IACL
- Per block can specify 1 different templates in EACL



Template

- Template 0~4 are pre-defined, 5~9 are user configurable
- Each field is 16 bits width

L2 ACL & VLAN Translation

Template 0

Field13	Field12	Field11	Field10	Field9	Field8	Field7	Field6	Field5	Field4	Field3	Field2	Field1	Field0
SPM3/ DMP3	SPM2/ DPM2	SPM1/ DPM1	SPM0/ DPM0	ETHERTYPE	DSAP SSAP	IPTOS PROTO	VLAN	SMAC2	SMAC1	SMAC0	DMAC2	DMAC1	DMAC0

L3/L4 ACL

Template 1

Field13	Field12	Field11	Field10	Field9	Field8	Field7	Field6	Field5	Field4	Field3	Field2	Field1	Field0
SPM3/ DPM3	SPM2/ DPM2	SPM1/ DPM1	SPM0/ DPM0	RANGE CHK	VLAN	L4 DPORT	L4 SPORT	TCP INFO	IPTOS PROTO	DIP1	DIP0	SIP1	SIP0

Template (cont.)

Control Protocol

Template 2

Field13	Field12	Field11	Field10	Field9	Field8	Field7	Field6	Field5	Field4	Field3	Field2	Field1	Field0
SLP/ DLP	Meta Data	L4 DPORT	L4 SPORT	DIP1	DIP0	SIP1	SIP0	IPTOS PROTO	ETHERT YPE	VLAN	DMAC2	DMAC1	DMAC0

L3/L4 IPv6 ACL

Template 3

Field13	Field12	Field11	Field10	Field9	Field8	Field7	Field6	Field5	Field4	Field3	Field2	Field1	Field0
SLP/ DLP	RANGE CHK	L4 DPORT	L4 SPORT	TCP INFO	IPTOS PROTO	DIP7	DIP6	DIP5	DIP4	DIP3	DIP2	DIP1	DIP0

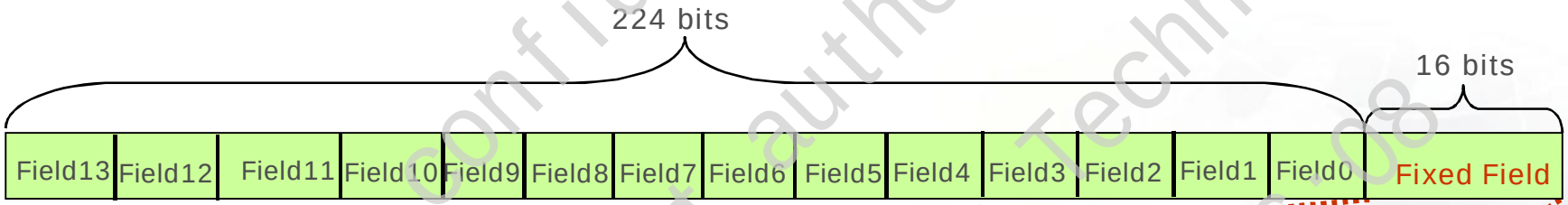
L3/L4 IPv6 ACL

Template 4

Field13	Field12	Field11	Field10	Field9	Field8	Field7	Field6	Field5	Field4	Field3	Field2	Field1	Field0
SPM3/ DPM3	SPM2/ DPM2	SPM1/ DPM1	SPM0/ DPM0	VLAN	Meta Data	SIP7	SIP6	SIP5	SIP4	SIP3	SIP2	SIP1	SIP0

ACL Entry Structure

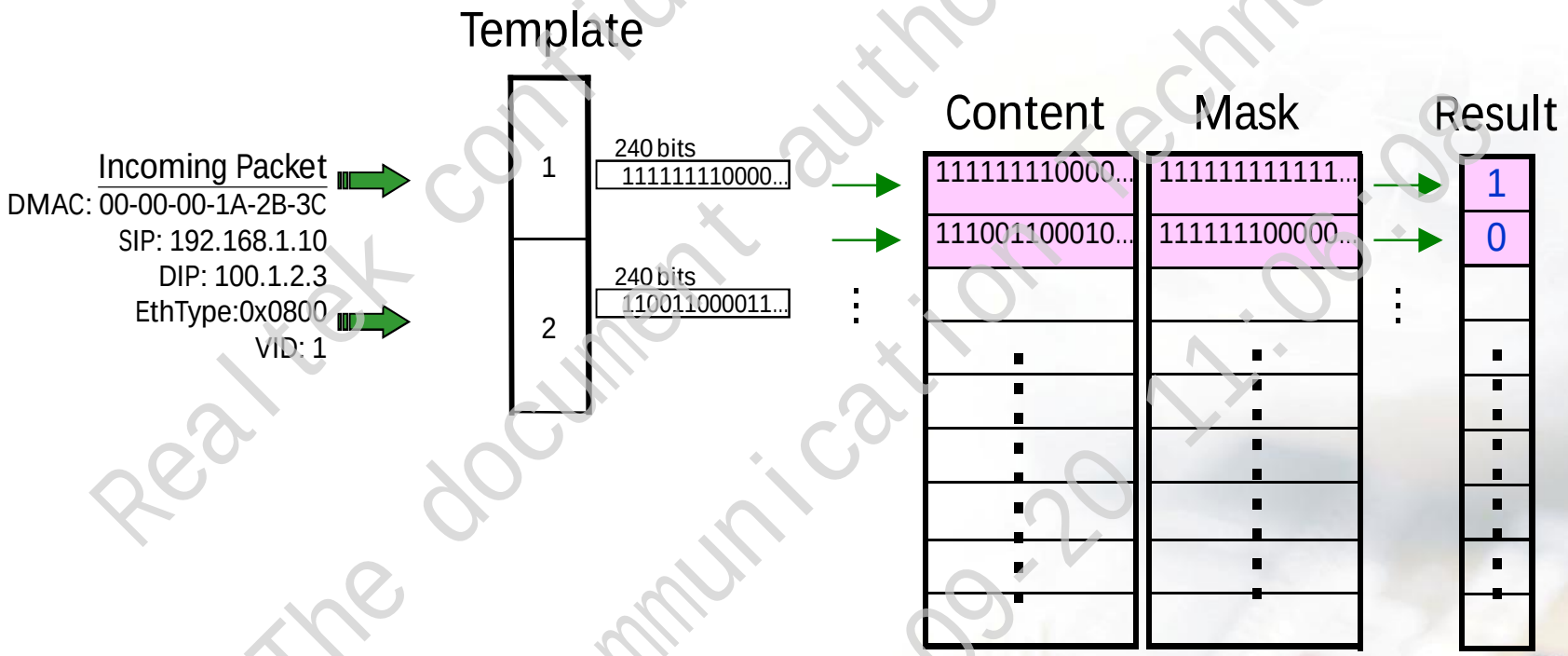
- Each entry has a “Fixed Field”
- Each ACL block can map to 2 templates in VACL and IACL
  - Template ID to identify the mapped template
  - The meaning of Field 0~13 is defined by the mapped template



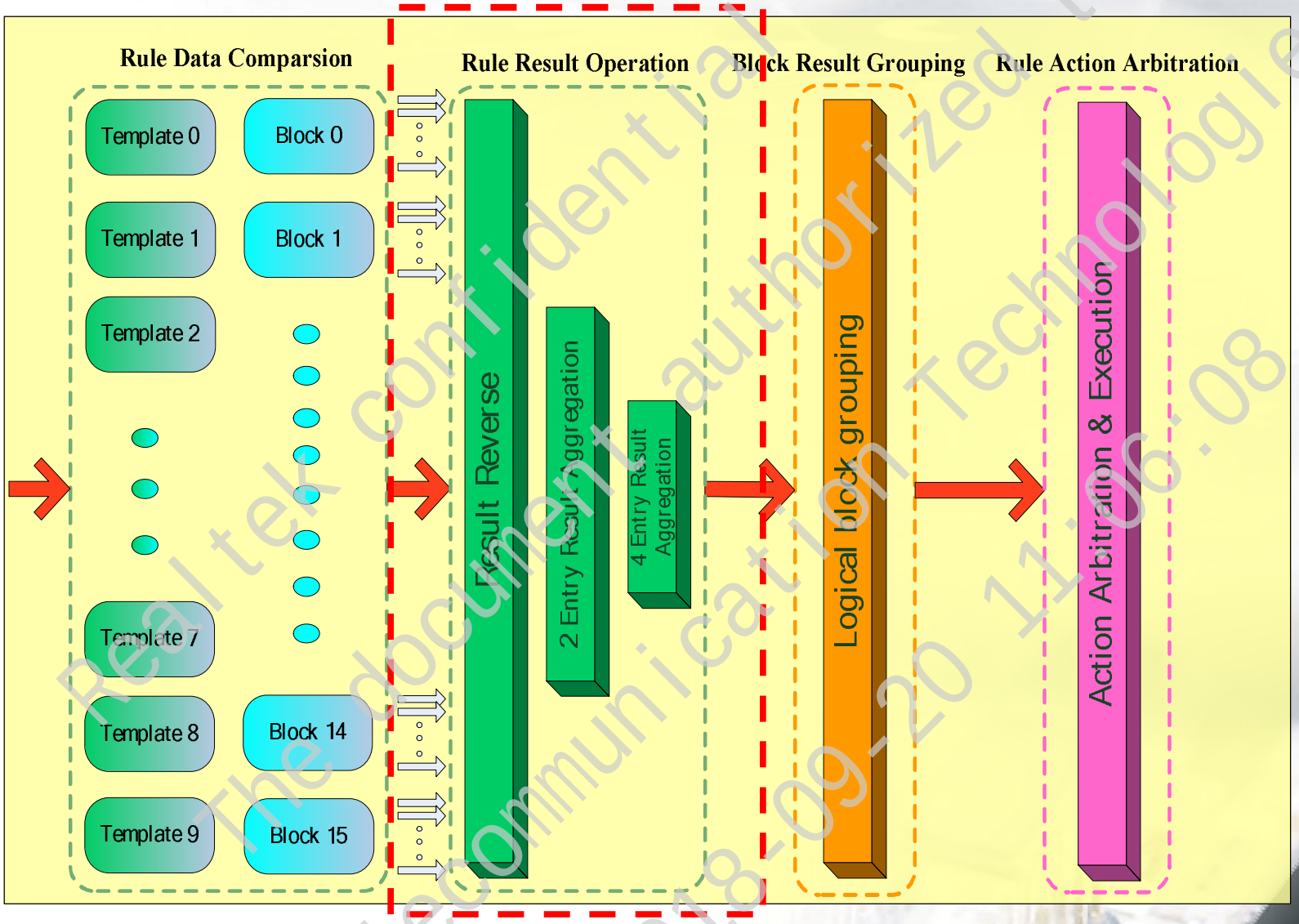
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Outer Untag or Outer Pri-Tag	Inner Untag or Inner Pri-Tag	Outer Tag or Outer Pri-Tag	Inner Tag or Inner Pri-Tag	00: Ethernet 01: LLC_SNAP 10: LLC_Other 11: Reserved		**IGR_NML_PORT/ EGR_NML_PORT**	**Template ID**
Bit 15	**Bit 14**	**Bit 13**	**Bit 12**	**Bit 11**	**Bit 10**	**Bit 9**	**Bit 8**
MGNT VLAN	For VACL Content too deep For IACL and EACL Switch MAC	IP4/IP6 offset!=0	000: UDP 001: TCP 010: ICMP/ICMPv6 011: IGMP 100: MLD 101: SCTP 110: L4 unknown			00: ARP 01: L2 ONLY 10: IPv4 11: IPv6	

Packet Inspect Flow of a Block

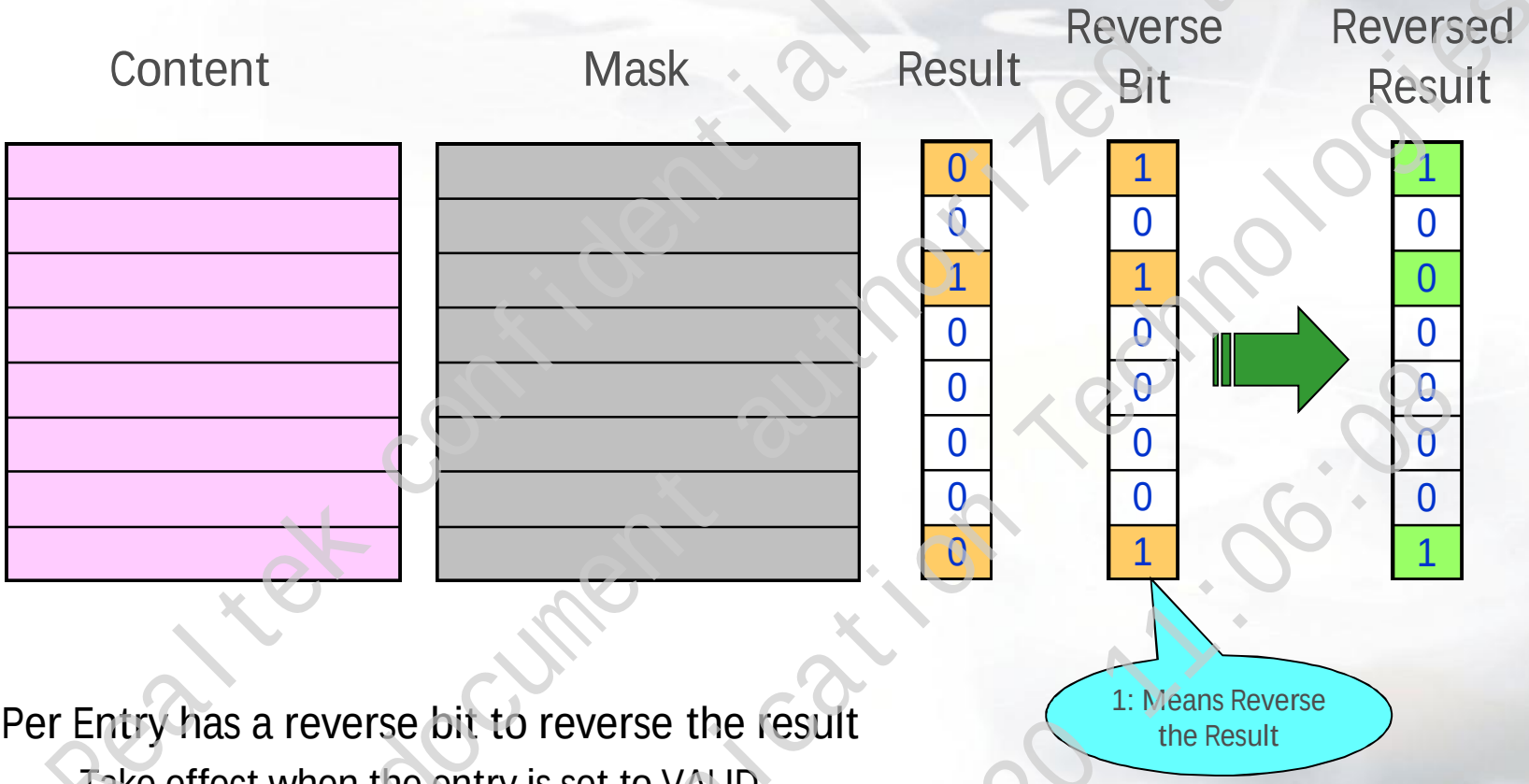
	Field13	Field12	Field11	Field10	Field9	Field8	Field7	Field6	Field5	Field4	Field3	Field2	Field1	Field0
Template 1	SPM3/ DPM3	SPM2/ DPM2	SPM1/ DPM1	SPM0/ DPM0	RANGE CHK	VLAN	L4 DPORT	L4 SPORT	TCP INFO	IPTOS PROTO	DIP1	DIP0	SIP1	SIP0
Template 2	SLP/ DLP	Meta Data	L4 DPORT	L4 SPORT	DIP1	DIP0	SIP1	SIP0	IPTOS PROTO	ETHERT YPE	VLAN	DMAC2	DMAC1	DMAC0



ACL Architecture Preview



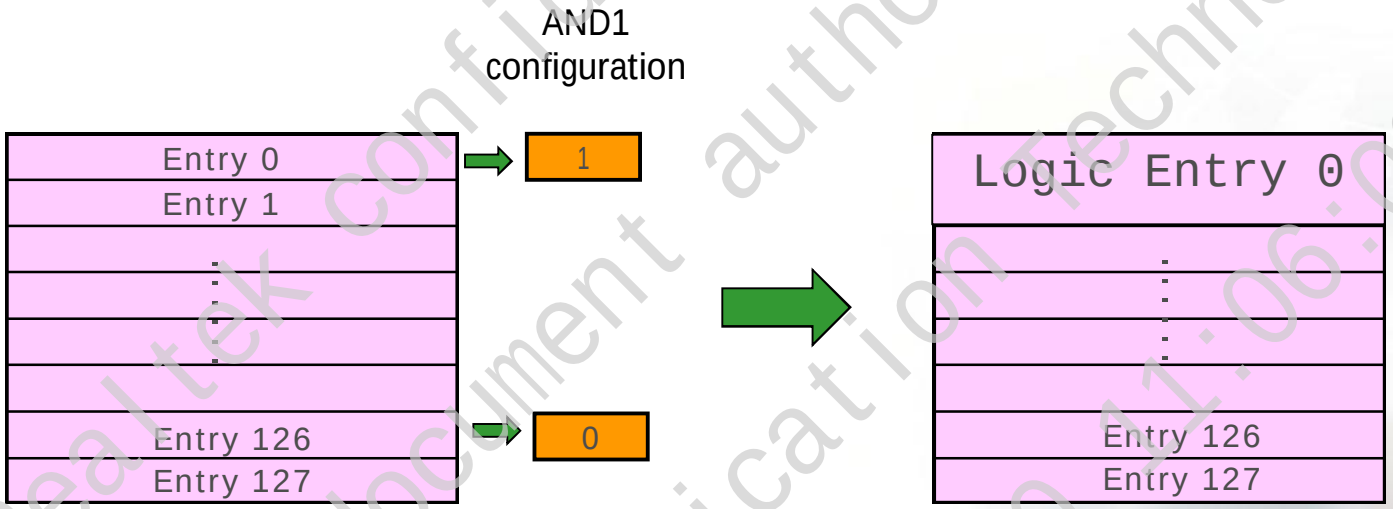
Entry Operation - Result Reverse



- Per Entry has a reverse bit to reverse the result
  - Take effect when the entry is set to VALID
- Ex: Range Check VID in [100..200]
  - → Reverse Result will be VID < 100 or VID > 200
- Ex: TCP port = 23
  - → Reverse Result will be all TCP ports except port 23

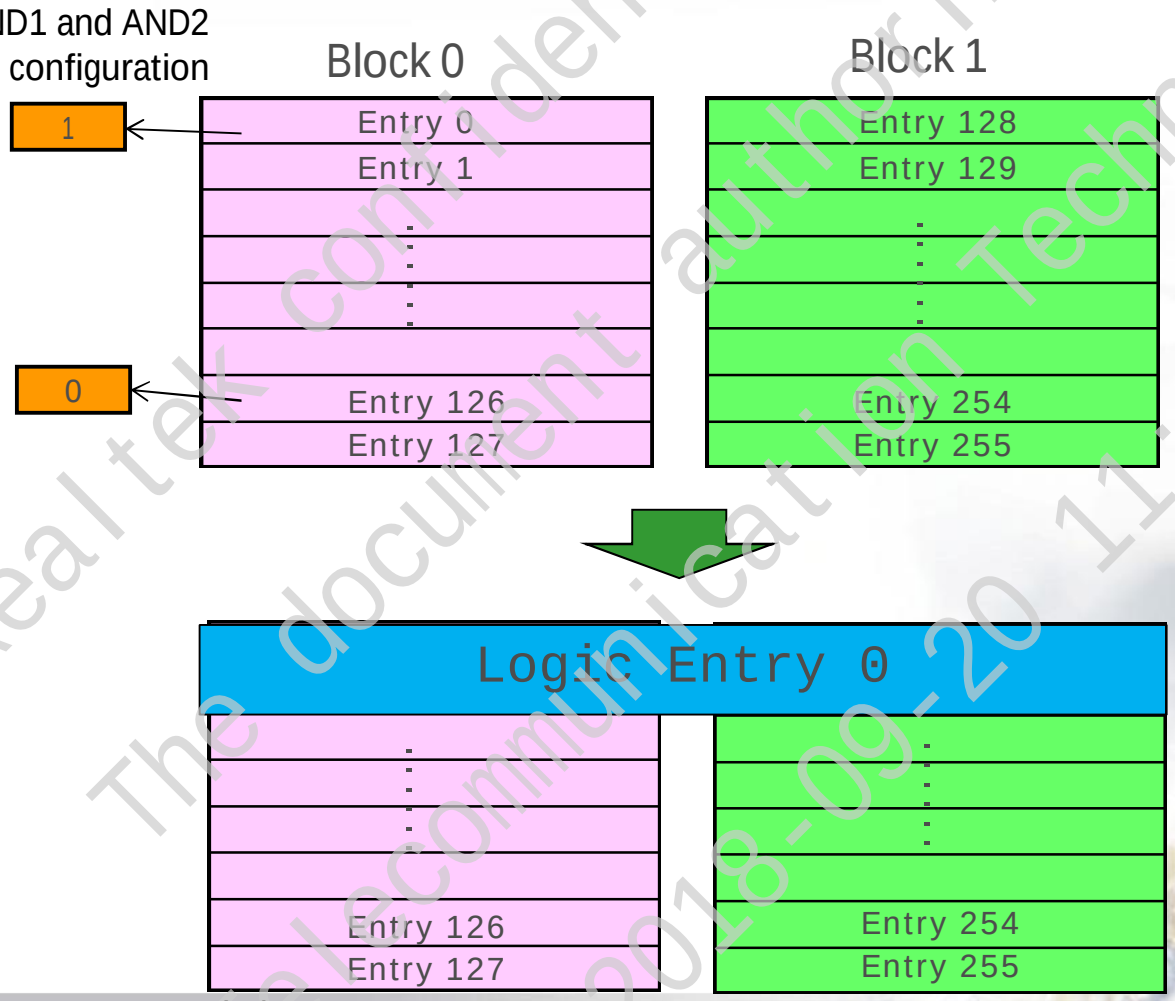
Entry Operation – Double-wide

- Double-wide key
  - Configure ADN1 operation
- Each 2 adjacent (index 2N&2N+1) entries treat as a logic entry
- Result only remains at the lowest index entry (others are cleared as 0)

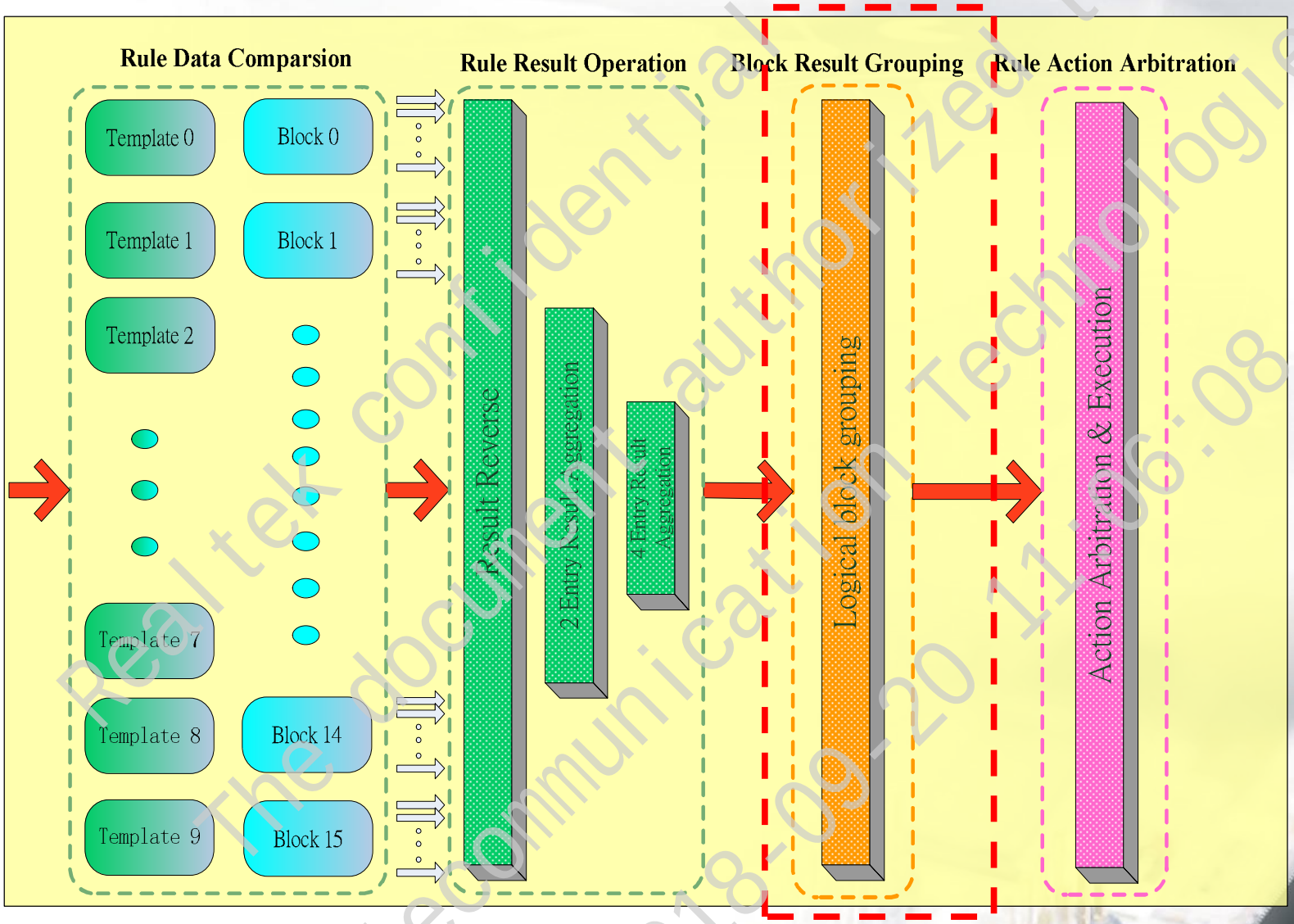


Entry Operation – Quad-wide

- Quad-wide key
  - Configure AND1 and AND2 operation both
- Each 4 entries (index 2N&2N+1&2N+128&2N+129) treat as a logic entry



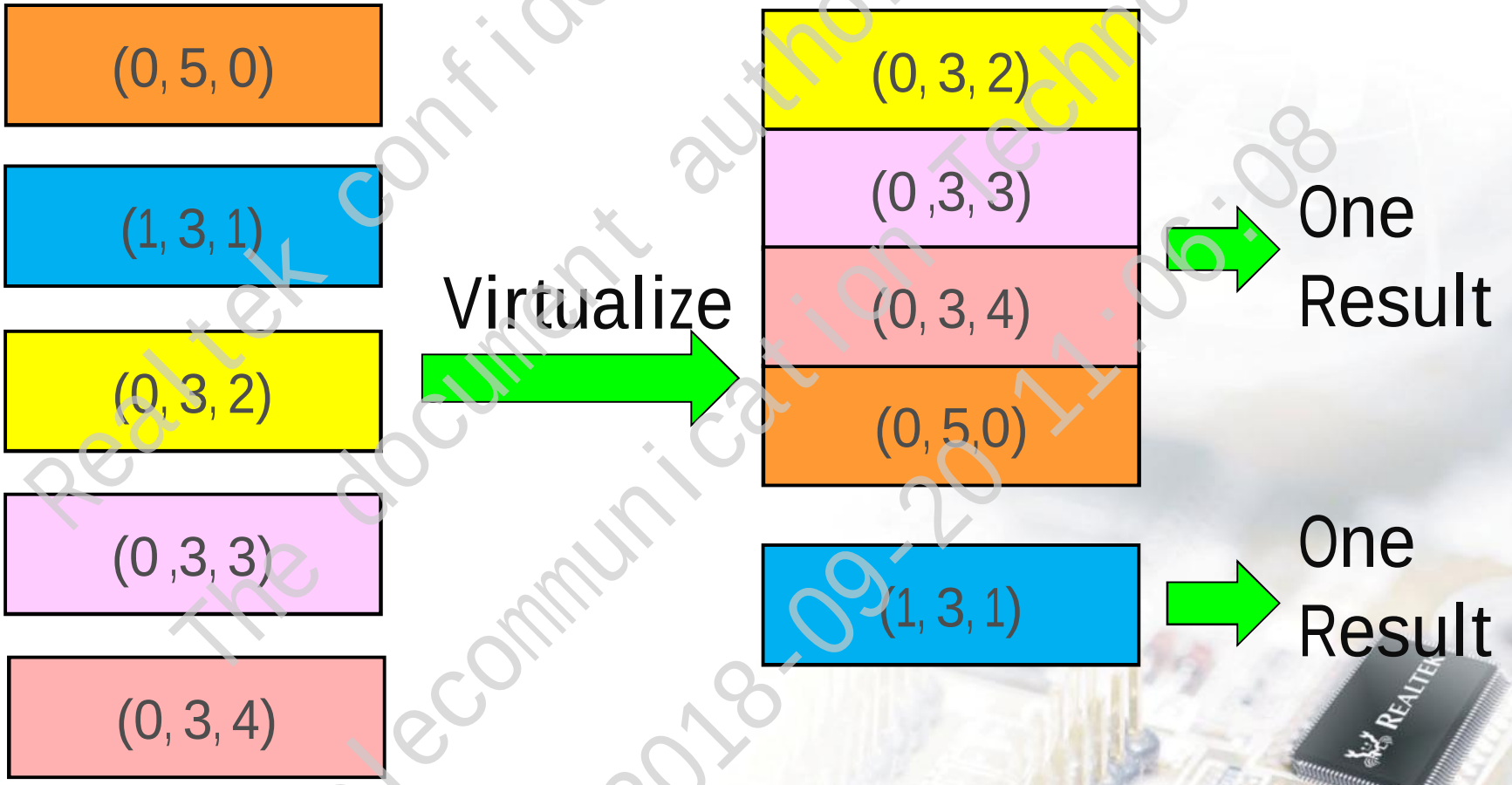
ACL Architecture Preview



Block Result Operation – Logical Block Grouping

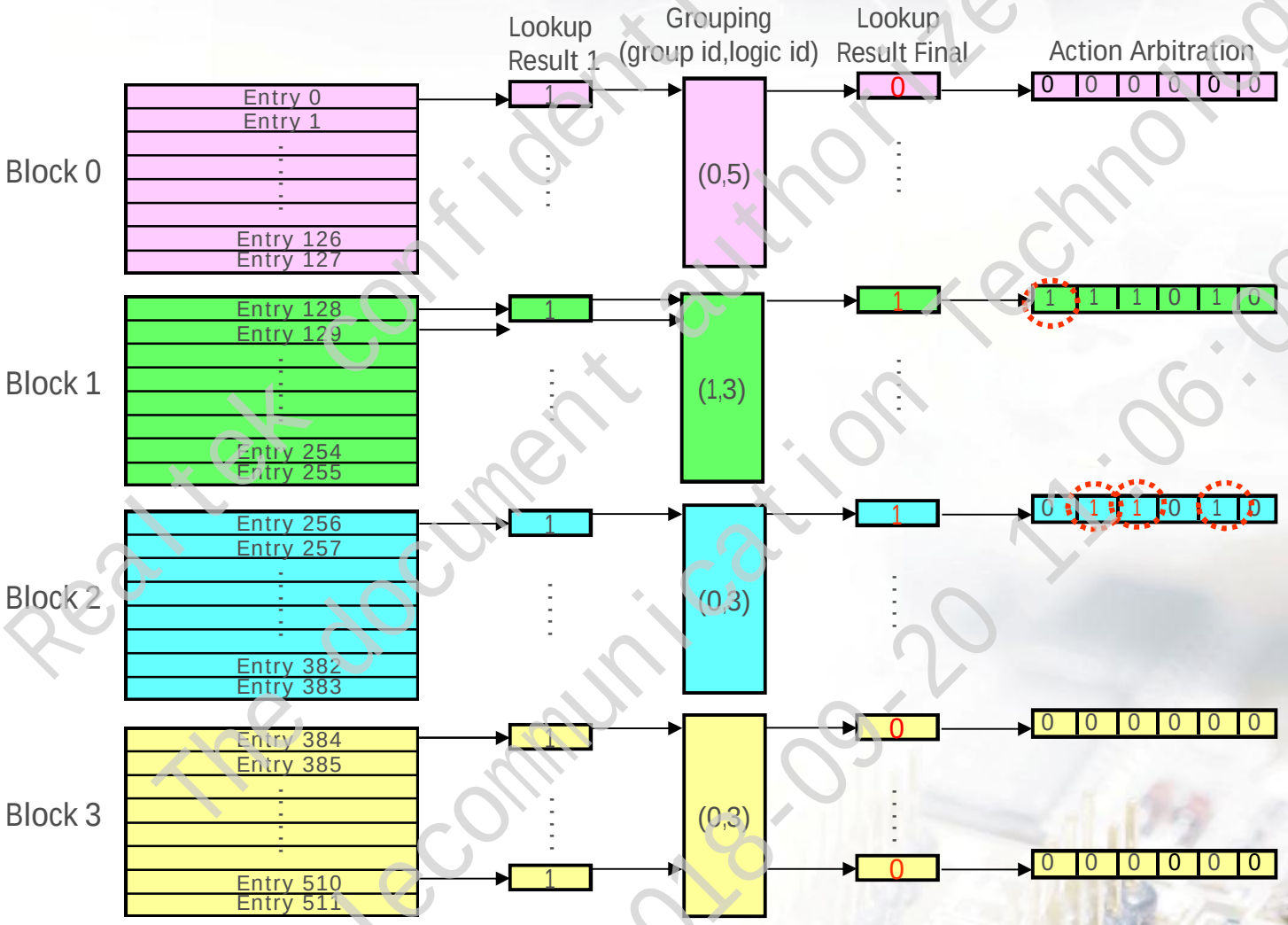
- Blocks that form a logic group only output a single hit result
  - Hit entry has lowest logic id and lowest physic block entry index
  - The same actions arbitration(group id > logic id > physical id)

(group id,logic id, physical id)



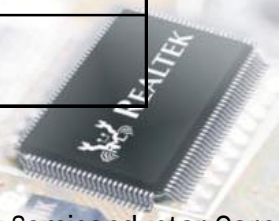
Block Result Operation Overview

- All non-conflict actions among hit entries are executed
- The same group actions arbitration(group id > logic id > physical id)



ACL Action Overview

Action	VACL	IACL	EACL
Green Drop/Yellow Drop/Red Drop	V	V	V
Forwarding	V (Extra: Unicast routing)	V (Extra: Egress Mask)	
Statistic	V	V	V
Mirror	V (Original)	V (Original or Modified)	
Policer	V	V	V
Inner VLAN Translation	V	V	V
Outer VLAN Translation	V	V	V
Inner Priority Remark	V	V	V
Outer Priority Remark	V	V	V
Priority Assignment	V	V	



ACL Action Overview

Action	VACL	IACL	EACL
Tag Status Assignment	V	V	
Bypass Dropping	V (IBW/Storm/STP/ IGR VLAN Filter and accept frame type)	V (IBW/Storm/ EGR VLAN filter)	
Meta-data	V	V	
CPU Queue ID Assignment	V	V	
DSCP Remarking	V	V	V
Invert IP Reserved Flag	V	V	

ACL Action – Drop

- Drop
  - Drop green packet
- Yellow drop
  - Drop yellow packet
- Red drop
  - Drop red packet

ACL Action – Forward

- Permit
  - To except the particular flow, packet still can be dropped by later modules
  - Drop subnet 192.168.1.x but accept 192.168.1.2 → insert permit entry to index N and insert drop entry to the index larger than N
- Drop
- Redirect
  - Trap: set port ID to CPU port
  - Redirect to single port
  - Redirect to multiple port
- Copy
  - Copy to CPU: set port ID to CPU port
  - Copy to single port
  - Copy to multiple port

ACL Action – Forward (Cont.)

- Unicast Routing (VACL only)
  - Index to L3 next-hop index
- Default Unicast Routing (VACL only)
  - Index to L3 next-hop index
- Mask (IACL only)
  - Mask the outgoing port → Flow-based port isolation
- Loopback
  - Configure maximum 7 loopback times or unlimited

ACL Action – Forward (Cont.)

- Forward Select
  - Per entry select the priority about forward and drop action
    - If FWD_SEL = 0

Fwd / Drop	Permit	Drop	Redirect	Copy	Routing
Permit	Permit	Permit	Permit	Permit	Permit
Drop	Drop	Drop	Drop	Drop	Drop

    - If FWD_SEL = 1

Fwd / Drop	Permit	Drop	Redirect	Copy	Routing
Permit	Permit	Drop	Redirect	Copy	Routing
Drop	Permit	Drop	Redirect	Copy	Routing

ACL Action – Statistics (Counter)

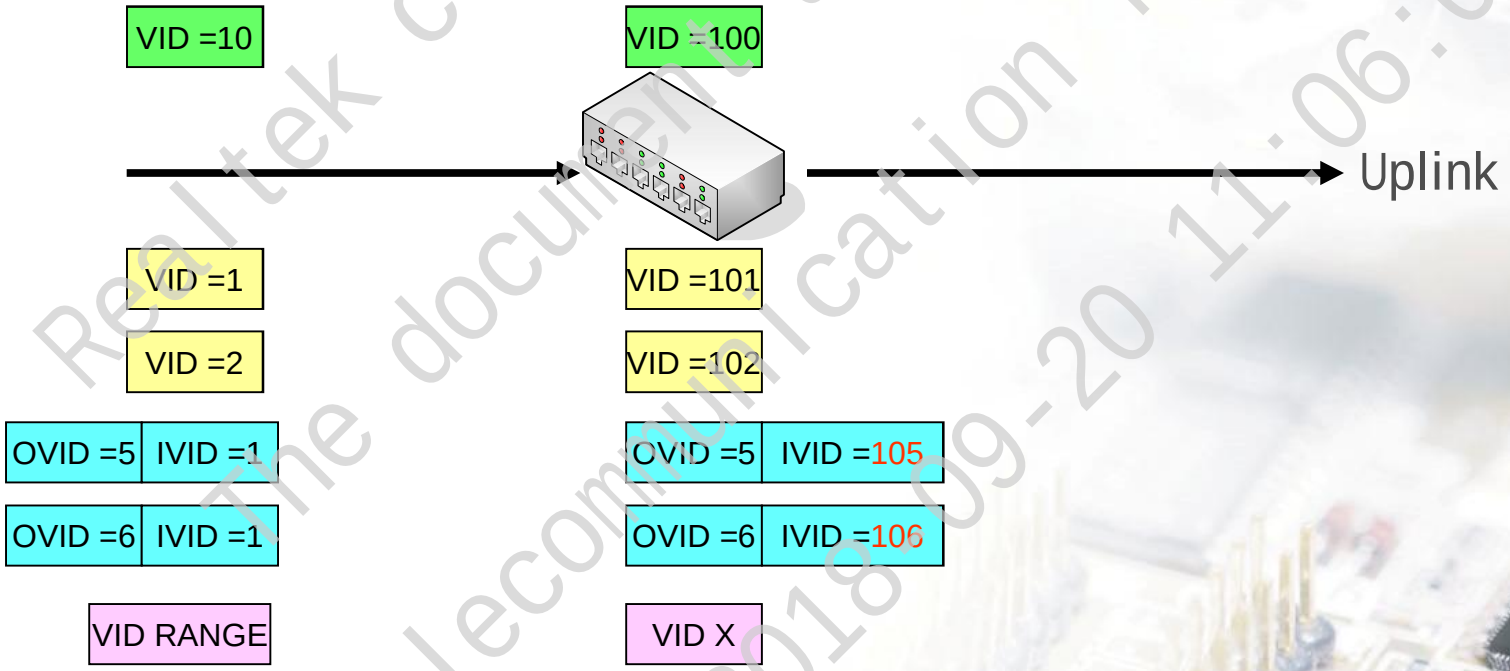
- To monitor the activity of specific flow
- 36-bit packet-based AND 42-bit byte-based per ACL entries
- One counter is increased in each logic group
- Up to 16 counter can be increased concurrently

ACL Action – Mirror

- Flow-based
- Operation
  - Cancel mirror
  - Mirror original content
  - Mirror modified content (IACL only)
- The mirroring port of the pointed mirror configuration is used
  - 4 mirror configurations are supported

ACL Action – I-VID Assignment

- Operation
  - Assign a new I-VID
  - Shift I-VID with a specified value (only applies to inner-tagged packet)
  - Shift I-VID from O-VID with a specified value (only applies to outer-tagged packet)
- Shift operation can be positive or negative



ACL Action – O-VID Assignment

- Operation
  - Assign a new O-VID
  - Shift O-VID with a specified value (only applies to outer-tagged packet)
  - Shift O-VID from I-VID with a specified value (only applies to inner-tagged packet)
- Shift operation can be positive or negative

ACL Action – Priority Assignment

- Assign a 3-bit internal priority (INT_PRI)
  - VACL assign INT_PRI is one of INT_PRI decision source
  - IACL assign INT_PRI will overwrite INT_PRI decision result

ACL Action – CPU QID Assignment

- CPU Queue ID
  - Just for CPU port
  - Assign the final queue id of packet to CPU directly
  - Could assign different queue id to the packet trap/copy to CPU
    - E.g. copy ARP packets to CPU → ARP packet to CPU could have different queue id from the ARP packet being forwarded to normal ports

ACL Action – Tag Priority Remarking

- Inner Tag Priority Operation
  - Force inner tag priority
  - Copy from outer tag priority
  - Keep inner tag priority
  - Determined by other modules
- Outer Tag Priority Operation
  - Force outer tag priority
  - Copy from inner tag priority
  - Keep inner tag priority
  - Determined by other modules

ACL Action – Tag Status Assignment

- Assign inner or outer tag status
  - Force untag
  - Force tag
  - Keep

ACL Action – DSCP Remarking

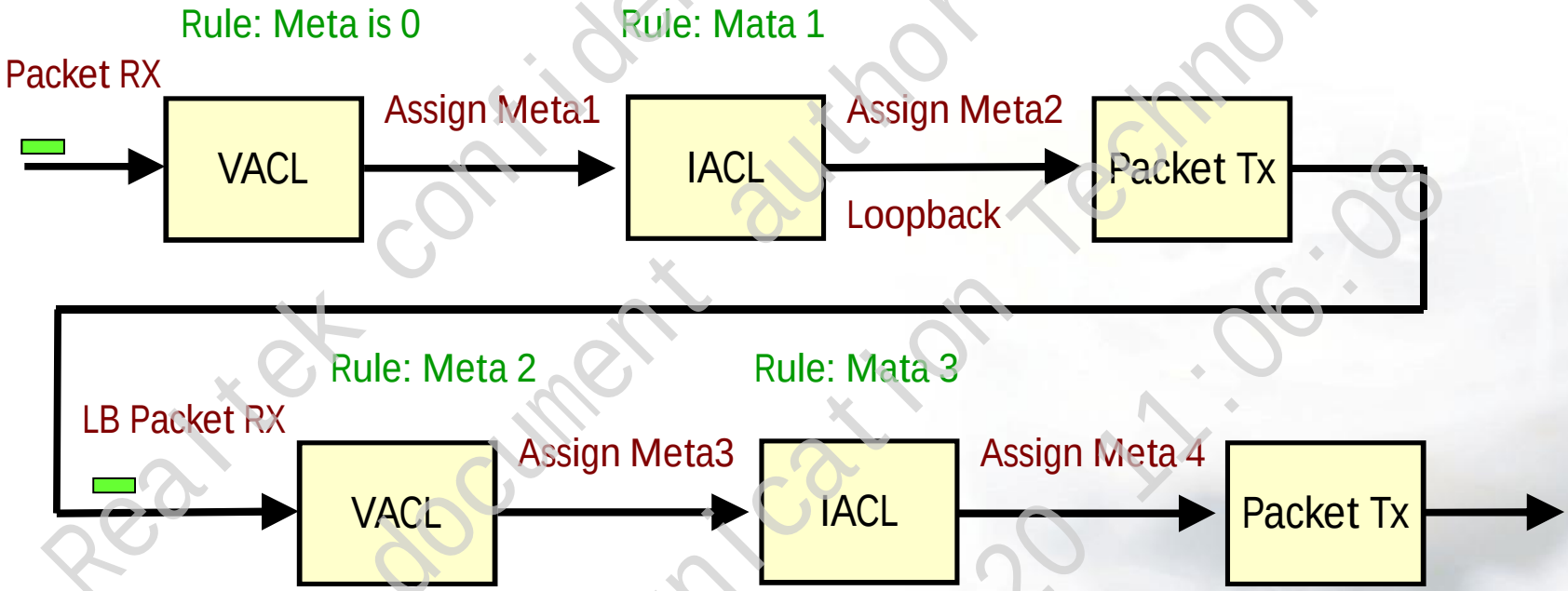
- Must be IP packet
- Green Remark
  - Remark Green packet
  - Can select remark dscp,tos or IP precedence
- Yellow Remark
  - Remark Yellow packet
  - Can select remark dscp,tos or IP precedence
- Red Remark
  - Remark Red packet
  - Can select remark dscp,tos or IP precedence

ACL Action – Bypass Dropping

- Bypass action only takes effect on packets destined to CPU
- VACL bypass action:
  - Bypass ingress port bandwidth control and storm control filtering
    - Bandwidth is not counted
  - Bypass ingress spanning tree filtering
  - Bypass ingress VLAN filtering and accept frame type
- IACL bypass action
  - Bypass ingress port bandwidth control and storm control filtering
    - Bandwidth is not counted
  - Bypass egress VLAN filtering

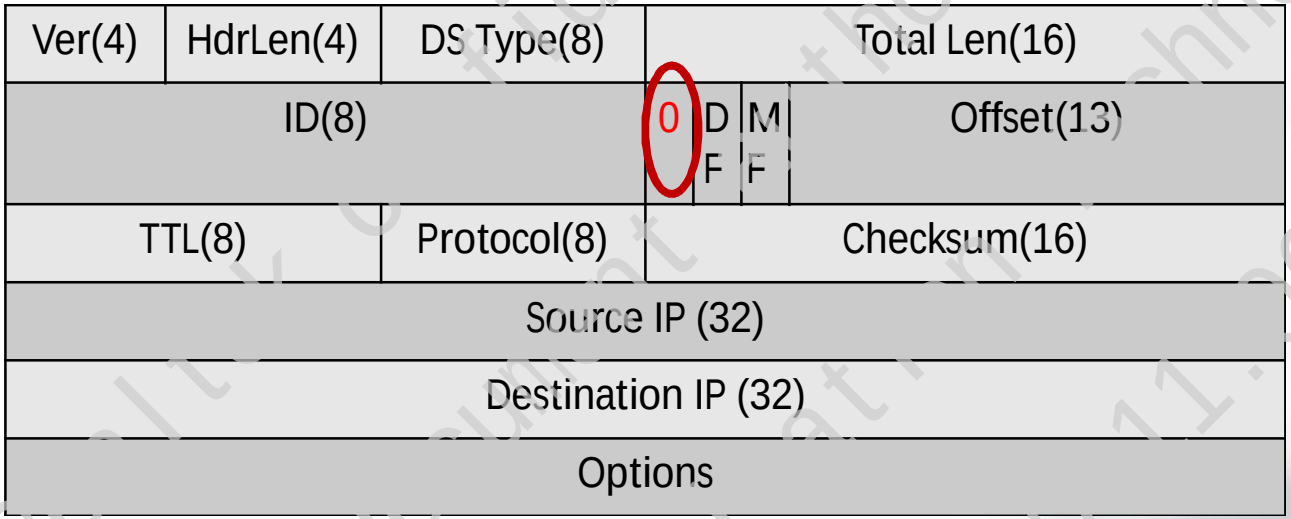
ACL Action – Metadata & Loopback

- 12-bits meta-data
- Co-operate with next phase ACL



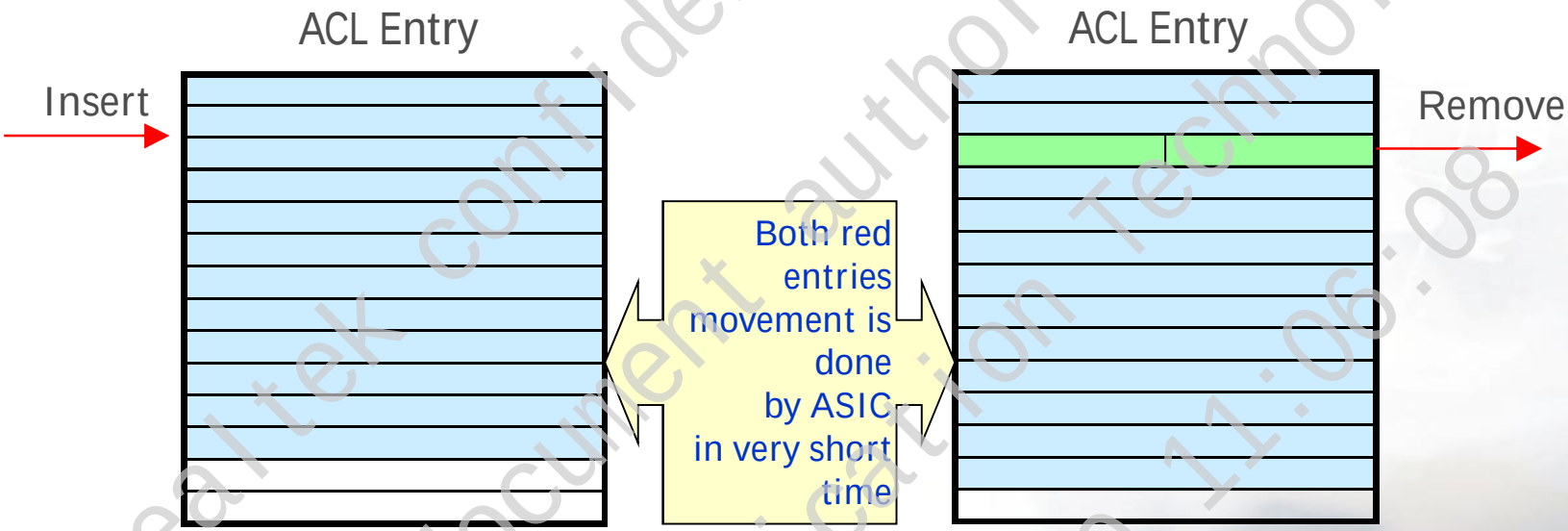
ACL Action – IP Reserved Flag Invert

- Operation
  - Invert Reserved Flag in IPv4 Header (0->1,1->0)
  - Flag bit as following:



ACL Content Control – Clearance and Move

- ACL entries clearance and movement can be done by ASIC
  - Entry clearance/movement includes entry data/mask + operations + actions
  - The vacant entries due to clearance/movement are cleared



- How to Insert an entry?

  → Mark the entries and ask ASIC to shift down one entry

  → Insert the entry

- How to delete an entry?

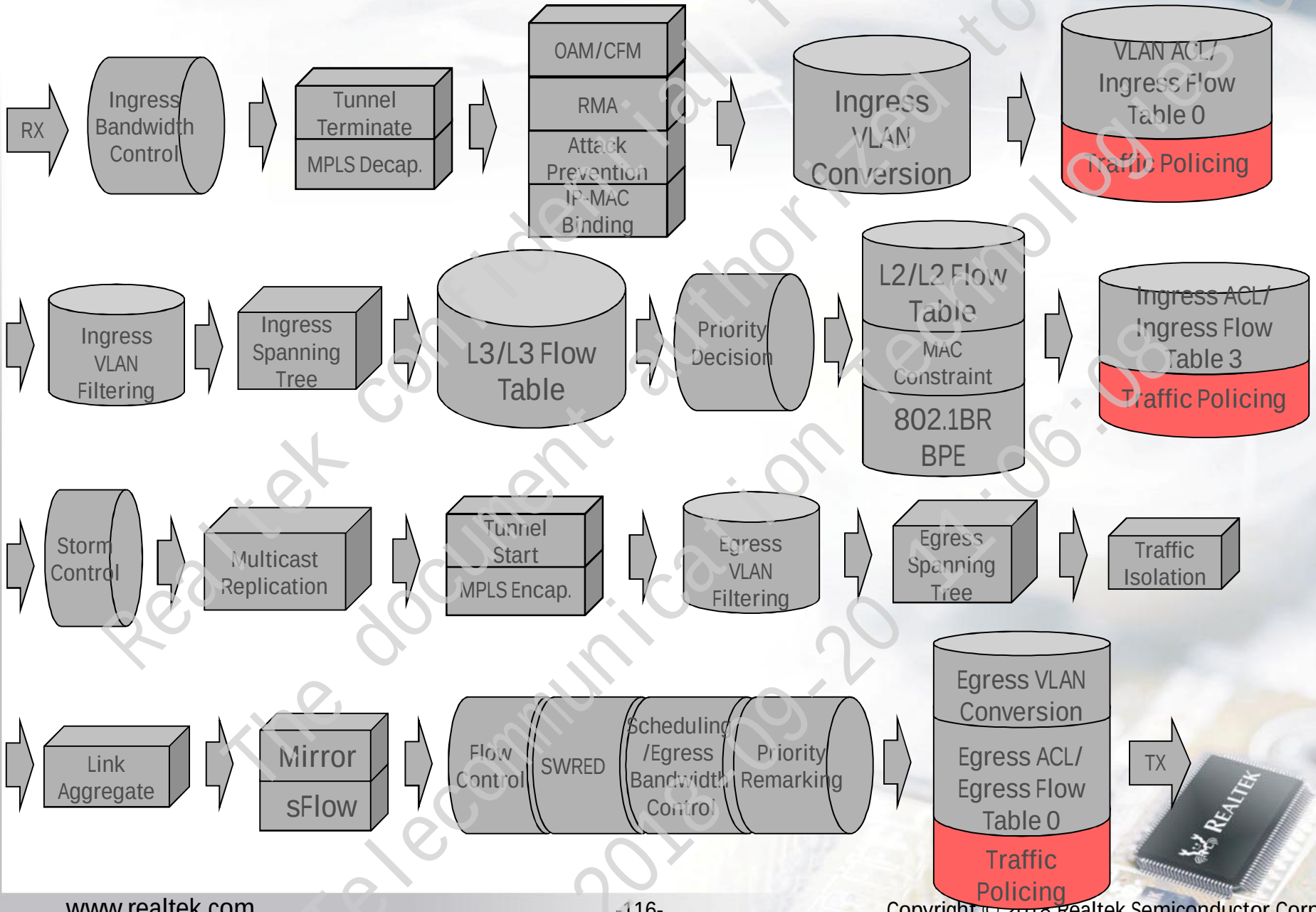
  → Delete the entry

  → Mark the entries and ask ASIC to shift up one entry

Realtek Semiconductor Corp.

Traffic Policer

Packet Processing Pipeline

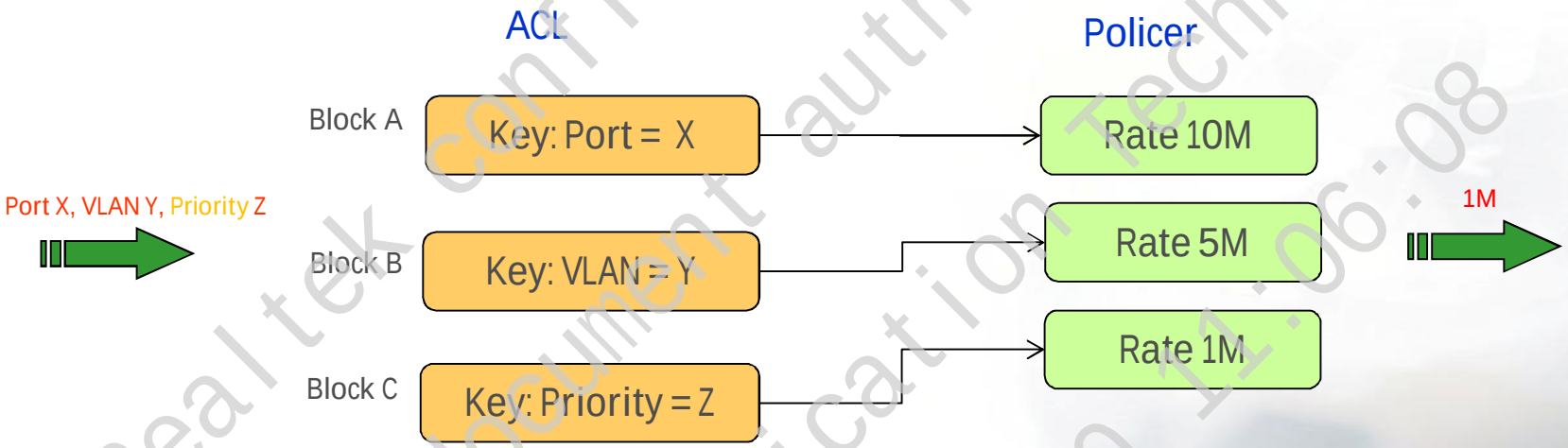


ACL Action – Policer

- 512 policers in total
- Three kinds of traffic policer
  - Dual leaky bucket (DLB)
  - RFC2697 Single Rate Three Color Marker (srTCM)
  - RFC2698 Two Rate Three Color Marker (trTCM)
- Per entry specify BPS(byte-count) or PPS(packet-count) mode

ACL Action – Hierarchy Policer

- Port X: 10M
- VLAN Y: 5M
- Priority Z: 1M

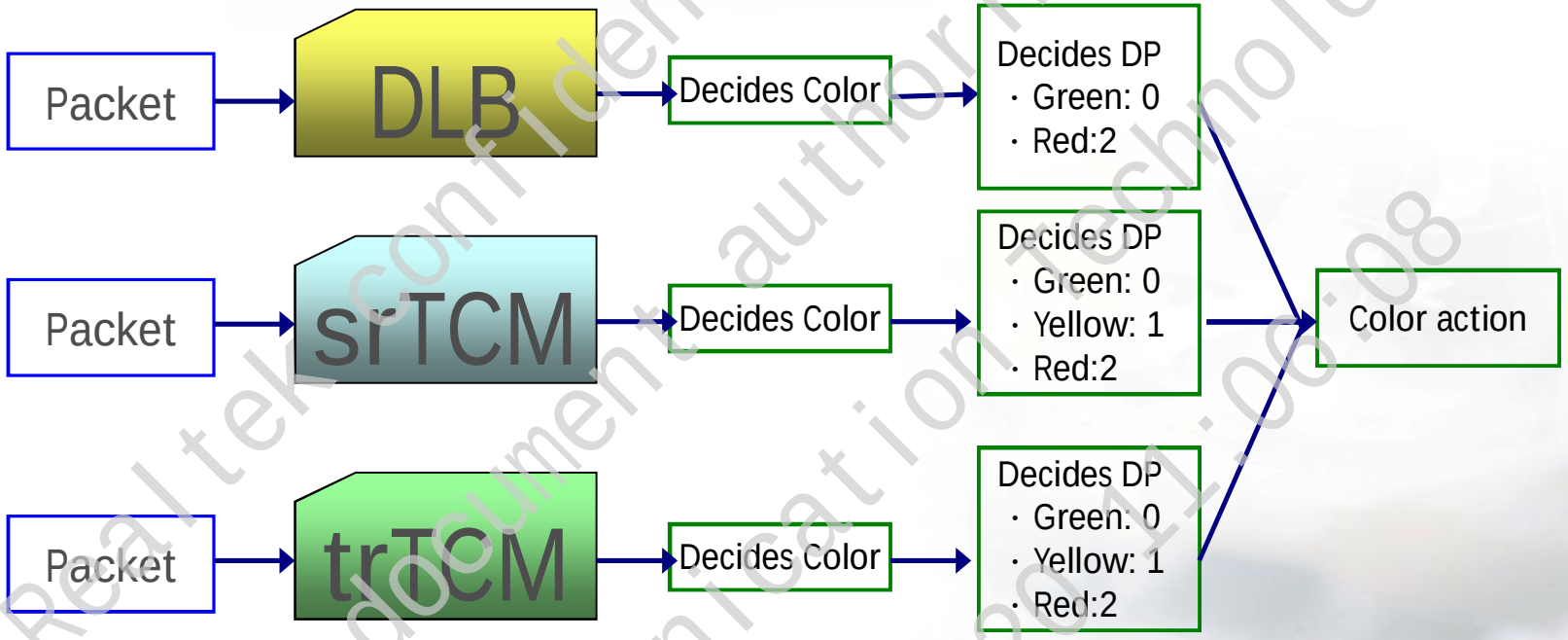


Policer table

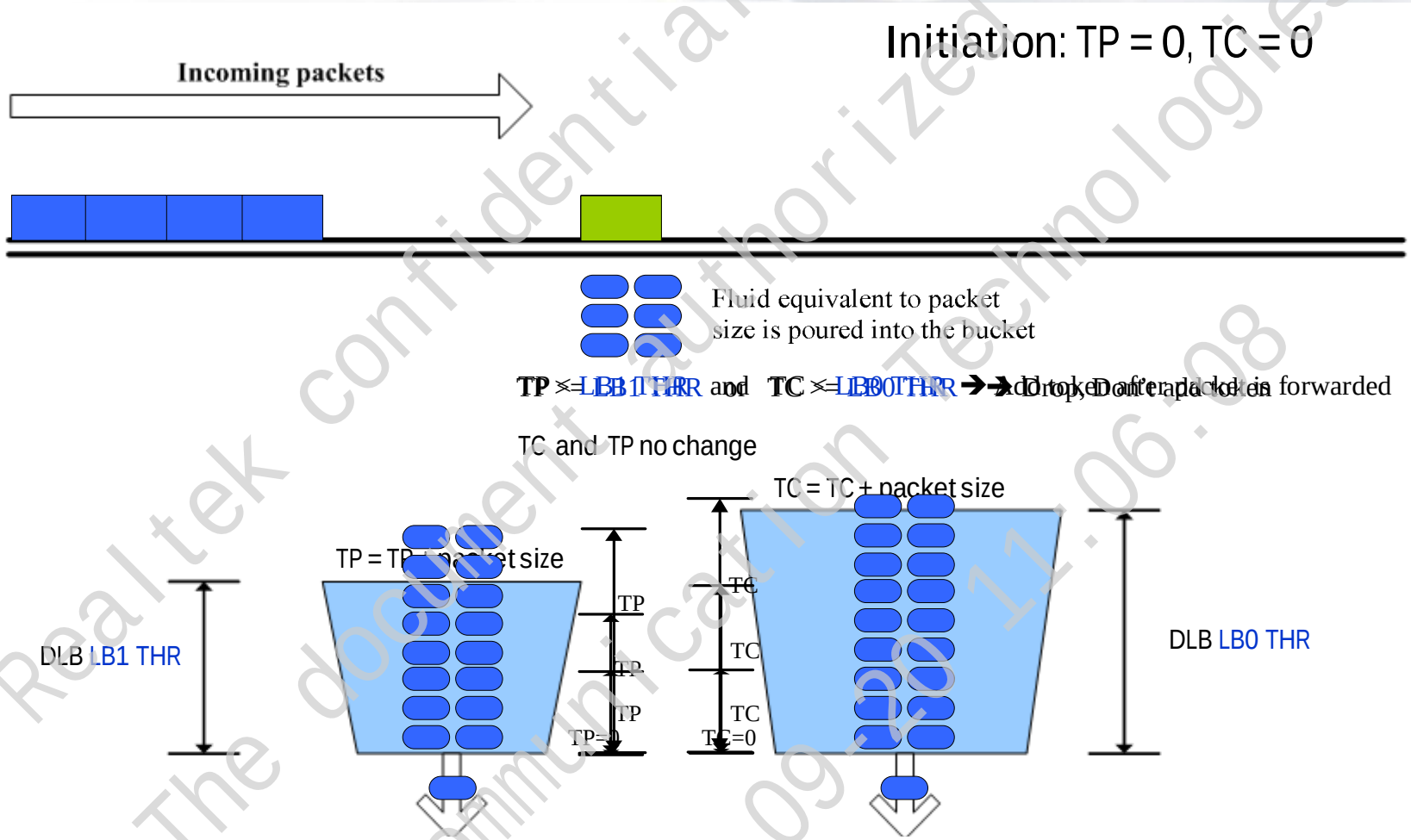
- Policer table entry format:

Field	Description
Type	invalid, DLB, srTCM, trTCM
Mode	Byte-based, Packet-based
ColorAware	Policer is color aware. This bit takes effect when type is srTCM or trTCM
LB0 Rate	Leaky bucket 0 rate For METER_MODE = 0, unit is 16Kbps For METER_MODE = 1, unit is 1 pps
LB0 Burst	Burst size of leaky bucket 0 For METER_MODE = 0, unit is 128 bytes (MAX 16 M bytes) For METER_MODE = 1, unit is 1 packet
LB1 Rate	Leaky bucket 1 rate
LB1 Burst	Burst size of leaky bucket 1

Policer Mechanism

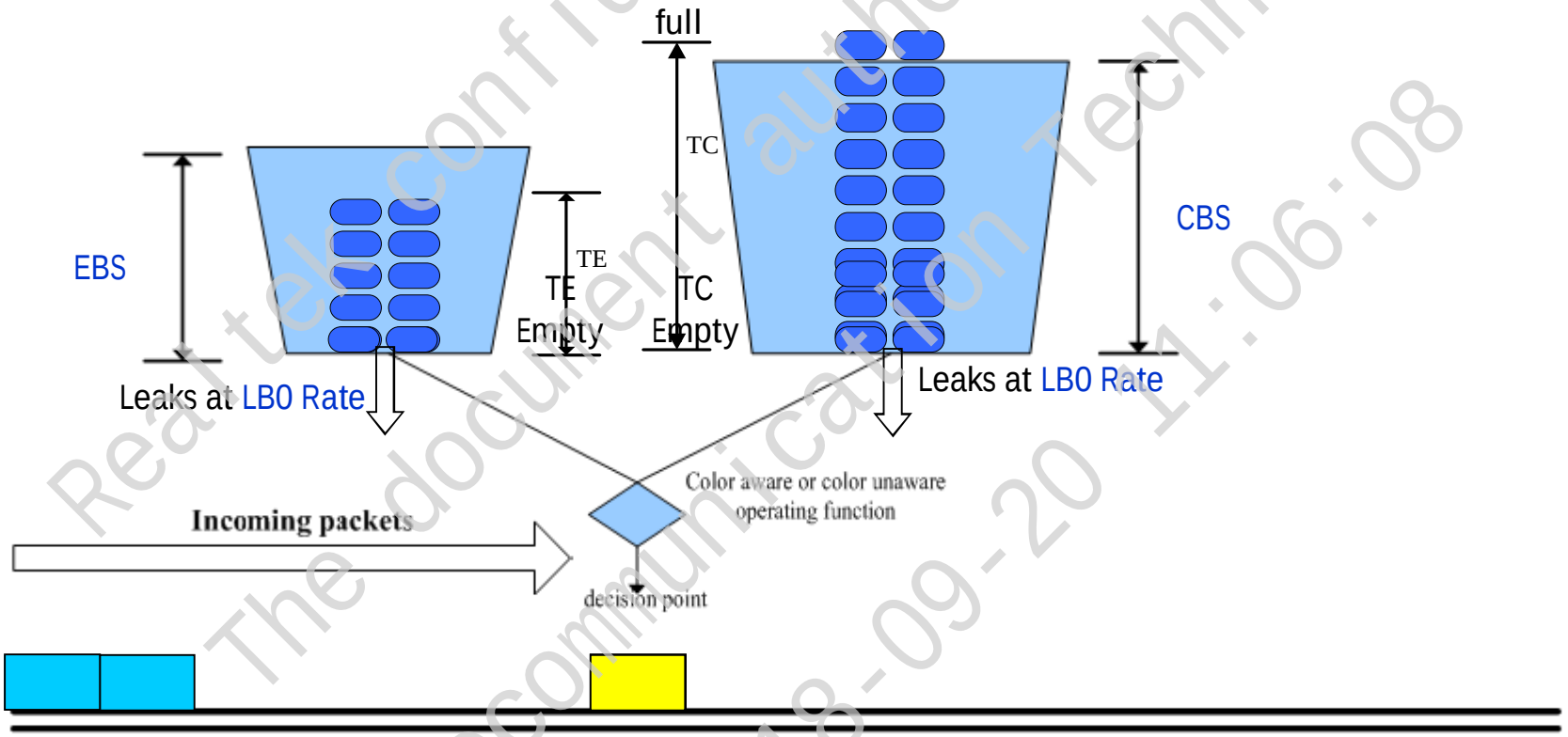


Dual Leaky Bucket



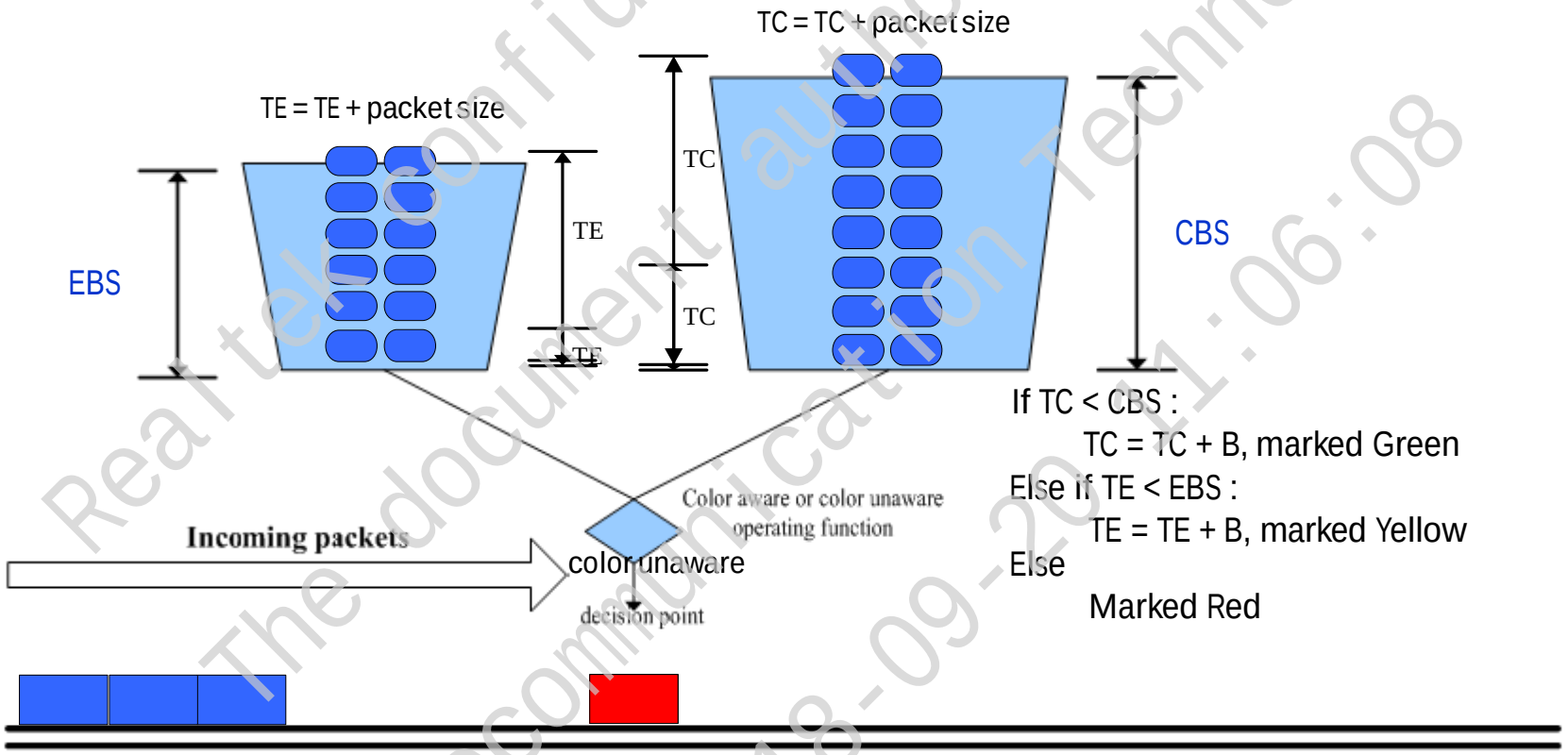
Single Rate Three Color Meter

- Initiation: TE = 0 (Empty), TC = 0 (Empty)
- Token increases when packet pass (Committed bucket has higher priority)
- Token leaks at LB0 Rate (Committed bucket has higher priority)



Single Rate Three Color Meter

Packet policing – color unaware



If TC < CBS :
TC = TC + B, marked Green
Else if TE < EBS :
TE = TE + B, marked Yellow
Else
Marked Red

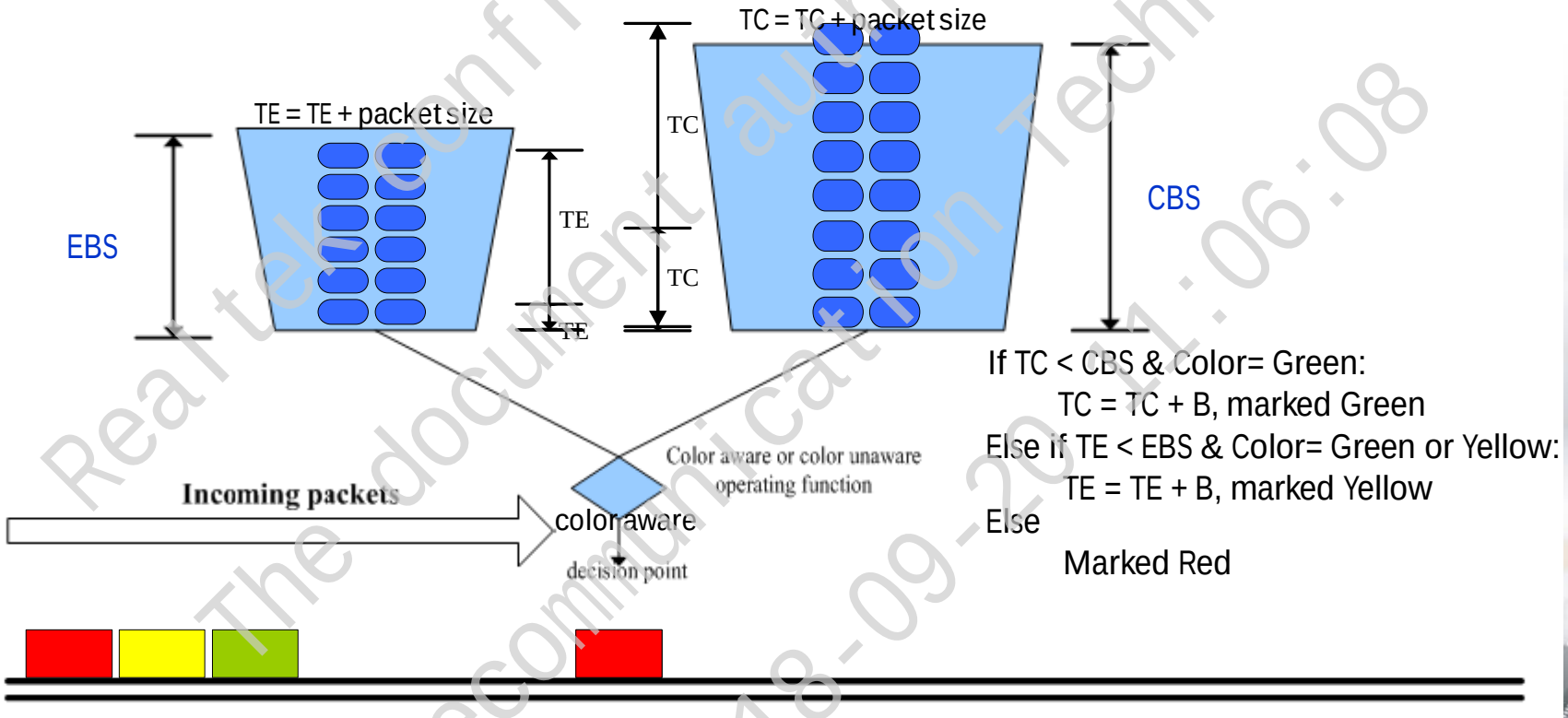
Single Rate Three Color Meter

Packet policing – color aware

Green -> Green/Yellow/Red

Yellow -> Yellow/Red

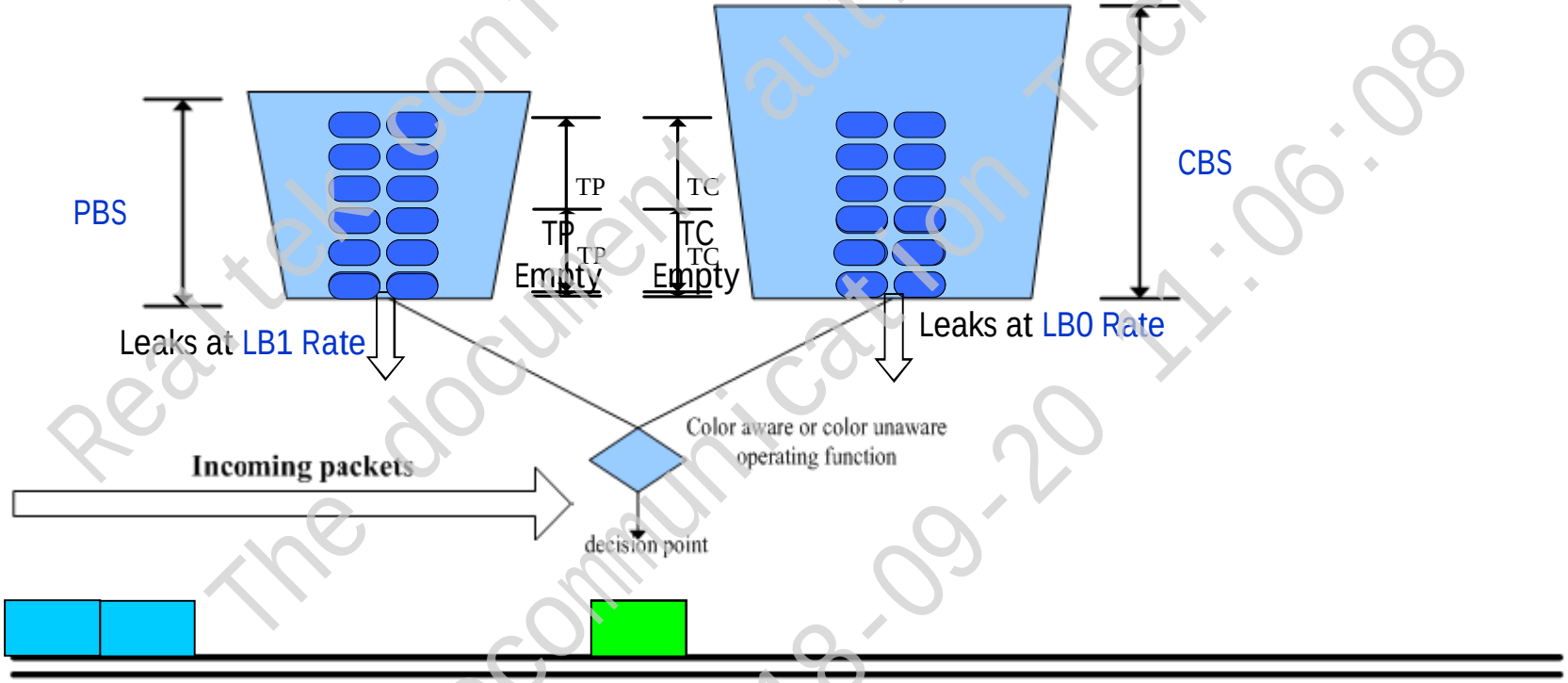
Red -> Red



If TC < CBS & Color= Green:
TC = TC + B, marked Green
Else if TE < EBS & Color= Green or Yellow:
TE = TE + B, marked Yellow
Else
Marked Red

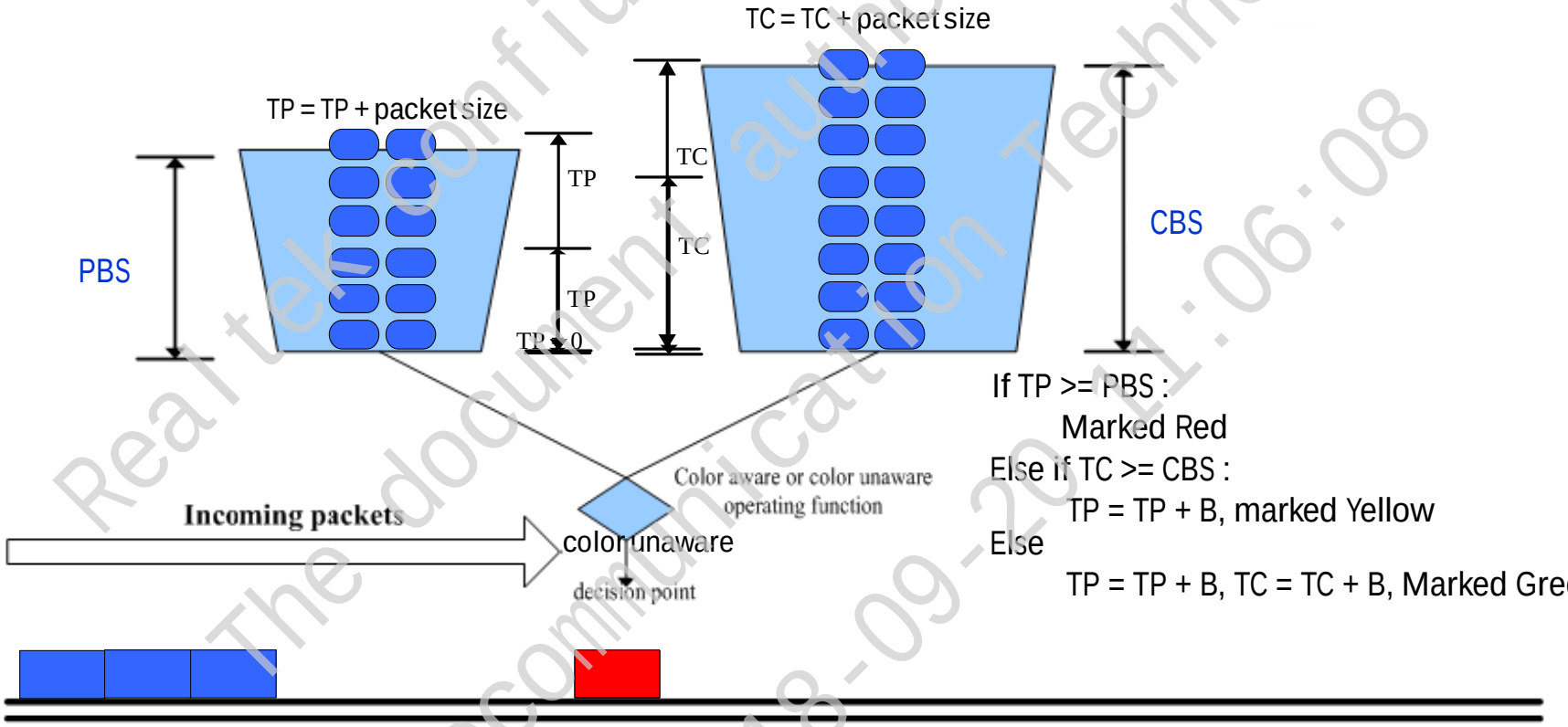
Two Rate Three Color Meter

- Initiation: TP = 0 (Empty), TC = 0 (Empty)
- Token increases when packet pass (Both buckets increase)
- Token leaks at LB0 Rate for Committed Bucket, LB1 Rate for Peak Bucket



Two Rate Three Color Meter

Packet policing – color unaware



If TP >= PBS :
Marked Red
Else if TC >= CBS :
TP = TP + B, marked Yellow
Else
TP = TP + B, TC = TC + B, Marked Green

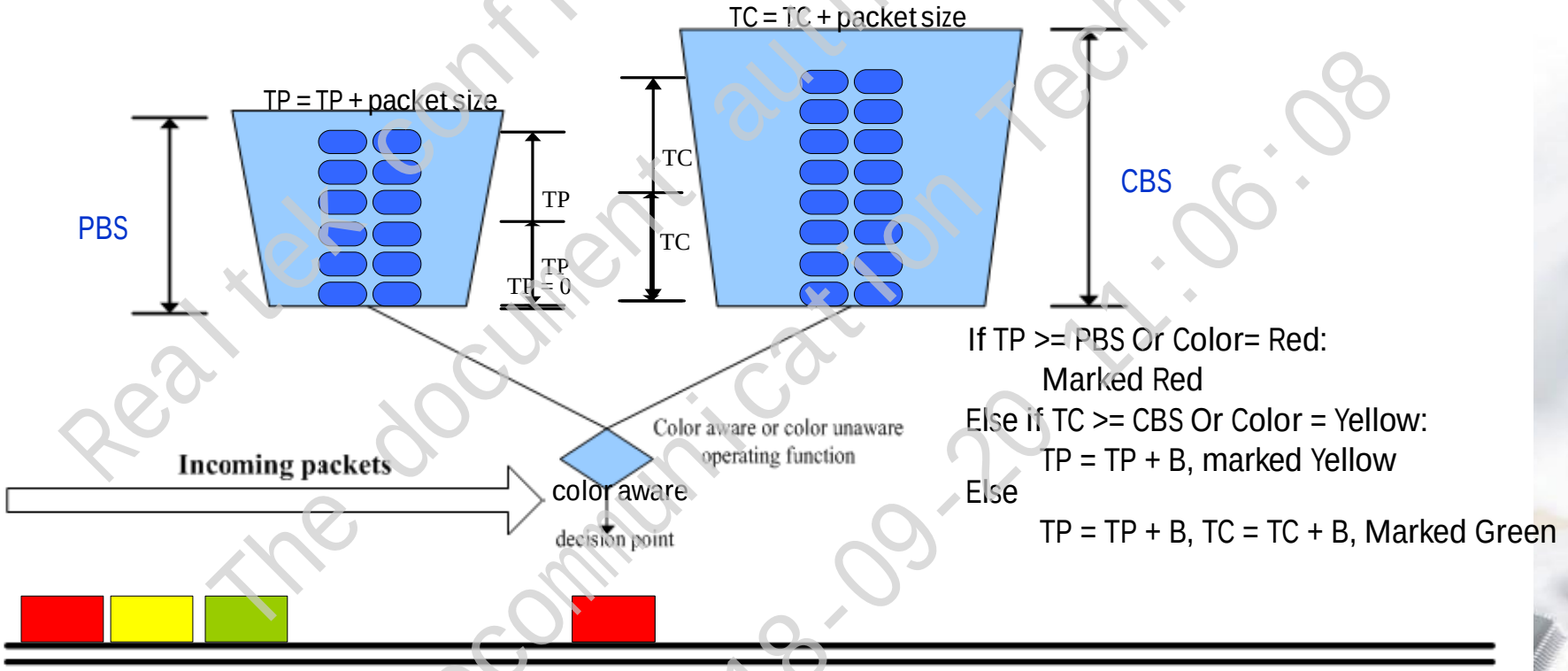
Two Rate Three Color Mode

Packet policing – color aware

Green -> Green/Yellow/Red

Yellow -> Yellow/Red

Red -> Red



If TP >= PBS Or Color= Red:

Marked Red

Else if TC >= CBS Or Color = Yellow:

TP = TP + B, marked Yellow

Else

TP = TP + B, TC = TC + B, Marked Green

Condition Table

	Forward	-	Drop
DLB	TC < LB0 TH and TP < LB1 TH	-	TC >= LB0 TH or TP >= LB1 TH

	Green	Yellow	Red
srTCM color unaware	TC < CBS	TC >= CBS and TP < EBS	TC >= CBS and TP >= EBS
srTCM color aware	Green: TC < CBS	TC >= CBS and TP < EBS	TC >= CBS and TP >= EBS
	Yellow: N/A	TP < EBS	TP >= EBS
	Red: N/A	N/A	Red

trTCM color unaware	TP < PBS and TC < CBS	TP < PBS and TC >= CBS	TP > =PBS
trTCM color aware	Green: TP < PBS and TC < CBS	TP < PBS and TC >= CBS	TP >= PBS
	Yellow: N/A	TP < PBS	TP >= PBS
	Red: N/A	N/A	Red

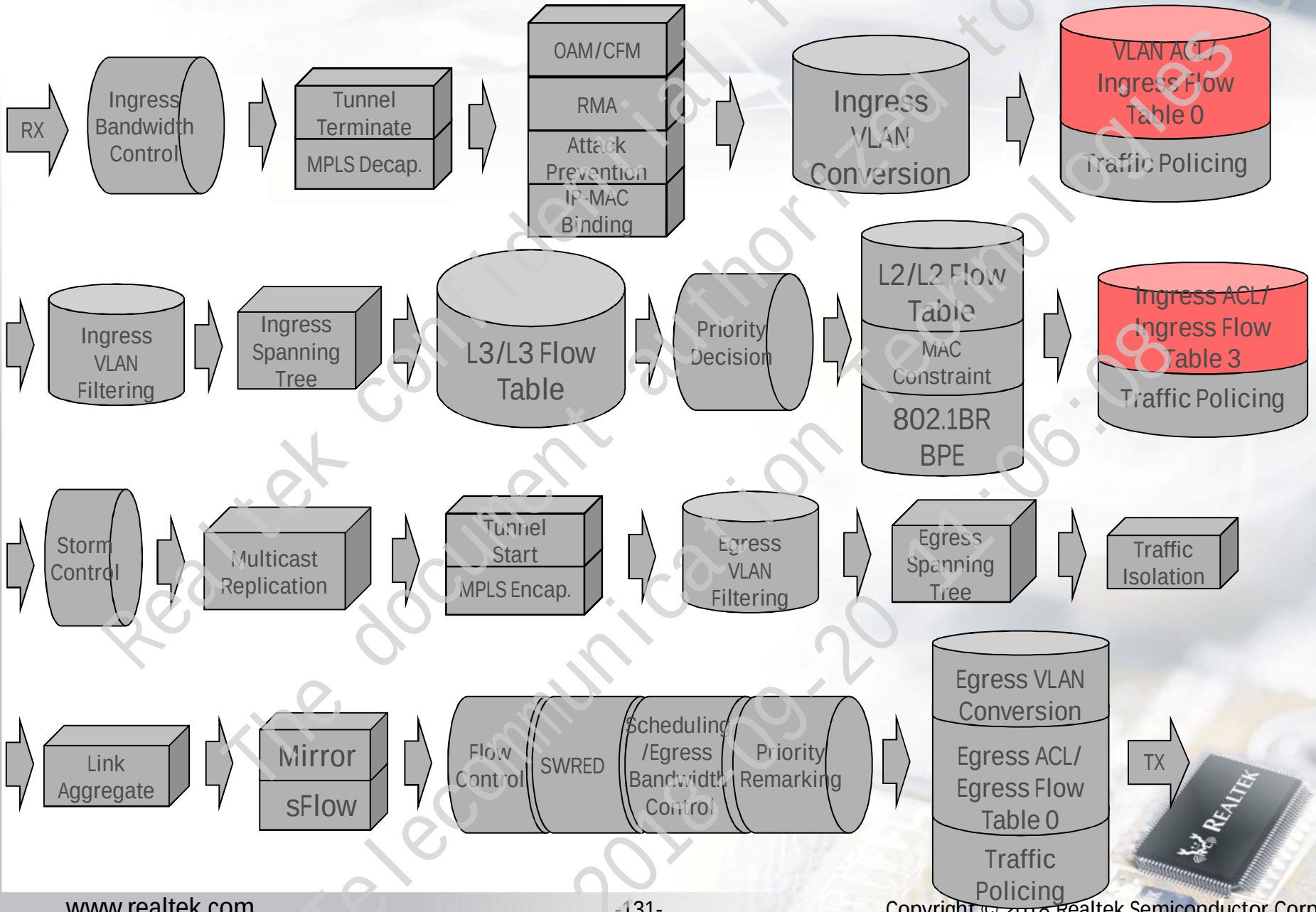
Realtek Semiconductor Corp.

Field Selector

Content

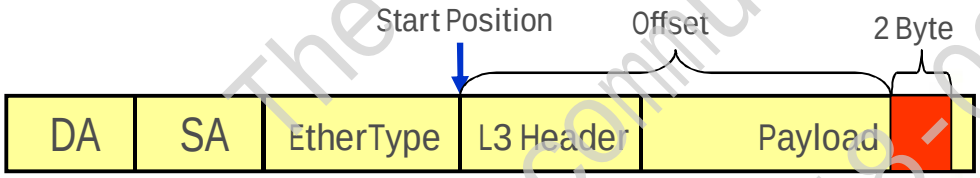
- System Description
- VLAN
- L2 MAC Address Lookup
- VLAN/Ingress/Egress ACL
- L3
- IP Tunnel
- Spanning Tree
- Traffic Isolation
- Link Aggregation
- Mirror
- RMA
- sFlow
- Ingress/Egress Bandwidth Control
- Storm Control
- Attack Prevention
- IP MAC Binding
- Priority Decision
- Scheduling
- Priority Remarking
- Flow Control
- Traffic Policing
- S-WRED
- NIC
- OAM/CFM
- MPLS
- OpenFlow
- Stacking
- 802.1BR Port Extension
- EEE
- LED
- MIB Statistics

Packet Processing Pipeline



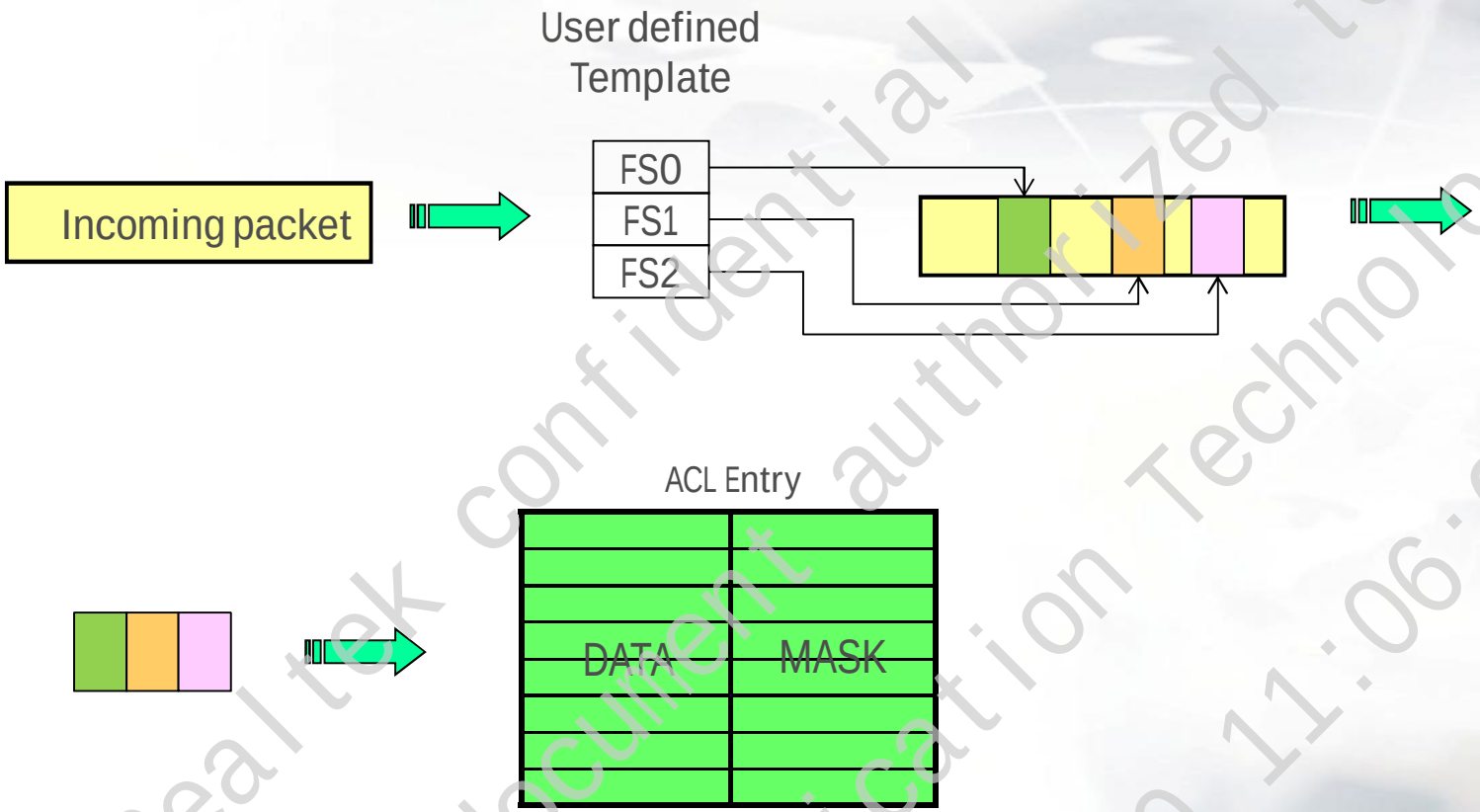
Field Selector Overview

- User defined field is used for the field type which is not defined yet
- Global 14 Field Selectors
  - 16 bits width
  - Starting position and an offset
- Select inner or outer for tunnel packet



- 7 Starting Position Options
  - Raw packet (begin with DA)
  - LLC packet
    - Begin with length “0000-05FF”
  - L3 packet
    - Start after EtherType(EthernetII)
    - VLAN tag is bypassed
  - RARP/ARP Header
    - Start from ARP/RARP header
  - IP Header
    - Start from IPv4/IPv6 header
  - IP Payload
    - Start from IP Payload
  - L4 Payload
    - Start after TCP/UDP Header

Field Selector

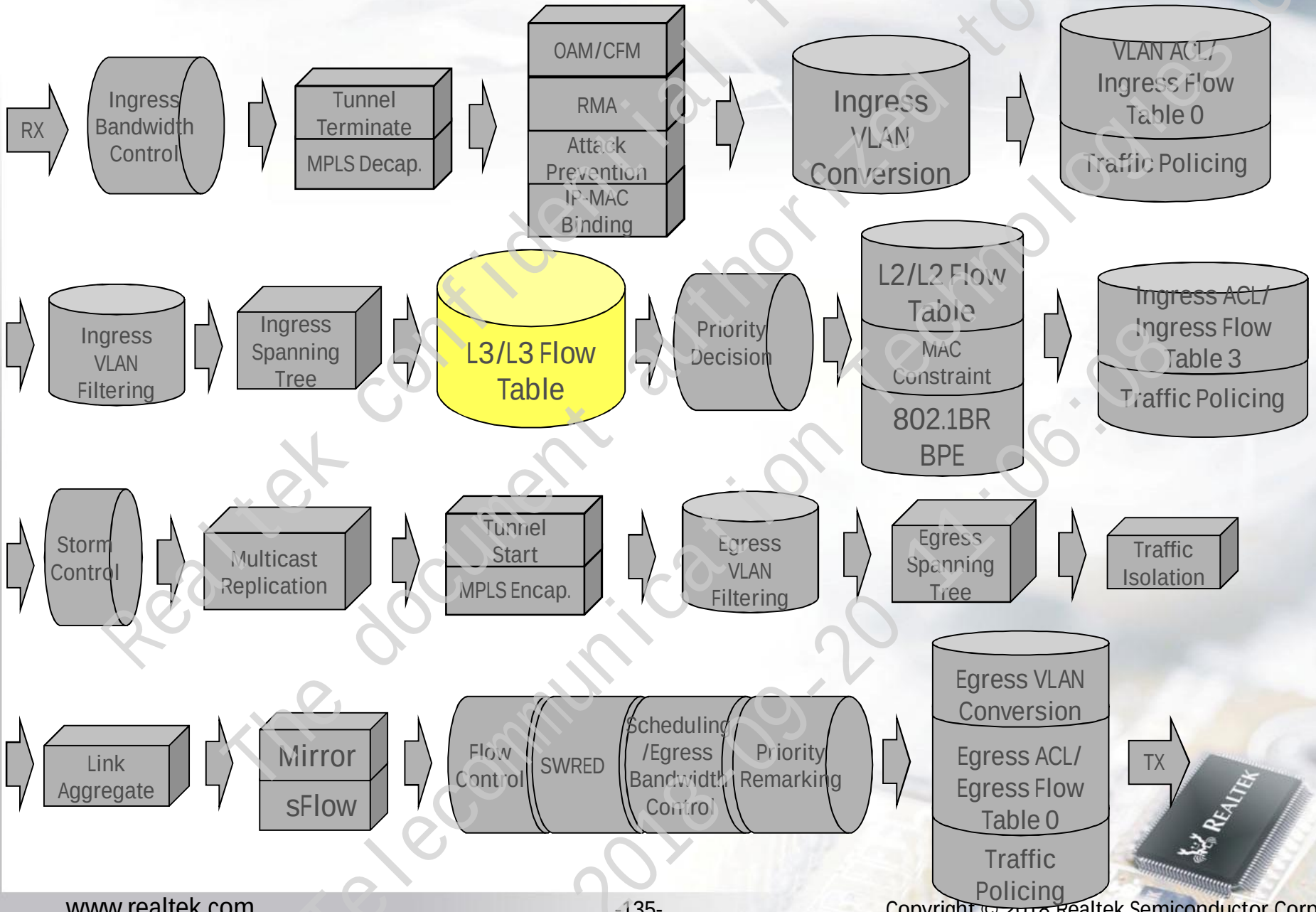


- Deepest length is 200 Byte from DA
- VLAN tags are excluded in field selector
- Field selector 6~11 is combo with IPv6 SIP[127:32] → field selector 6~11 will be IPv6[127:47] if receiving packet is an IPv6 packet

Realtek Semiconductor Corp.

L3

Packet Processing Pipeline



Overview (1/2)

- Unicast Routing
  - IPv4/IPv6 unicast routing support
  - 1K L3 interfaces
    - 256 VRF (VPN/Virtual Routing & Forwarding) instances
    - 16 MTU entries for different MTU values (per interface specifies)
    - 1K router MAC entries
  - 12K IPv4 or 4K IPv6 host entries
    - 2-left 6-way host table
  - 12K IPv4 or 4K IPv6 L3 LPM (Longest Prefix Match) entries
    - Hardware-based searching/moving/clearing entries support
  - 8K next-hop entries
  - 256 ECMP group entries, up to 8 next-hop paths for each
    - DLB (Dynamic Load Balancing) support by monitoring the traffic
  - uRPF (Unicast Reverse Path Forwarding) check
  - PBR (Policy-Based Routing) support by VLAN ACL
    - 2 types: Unicast and Default routes

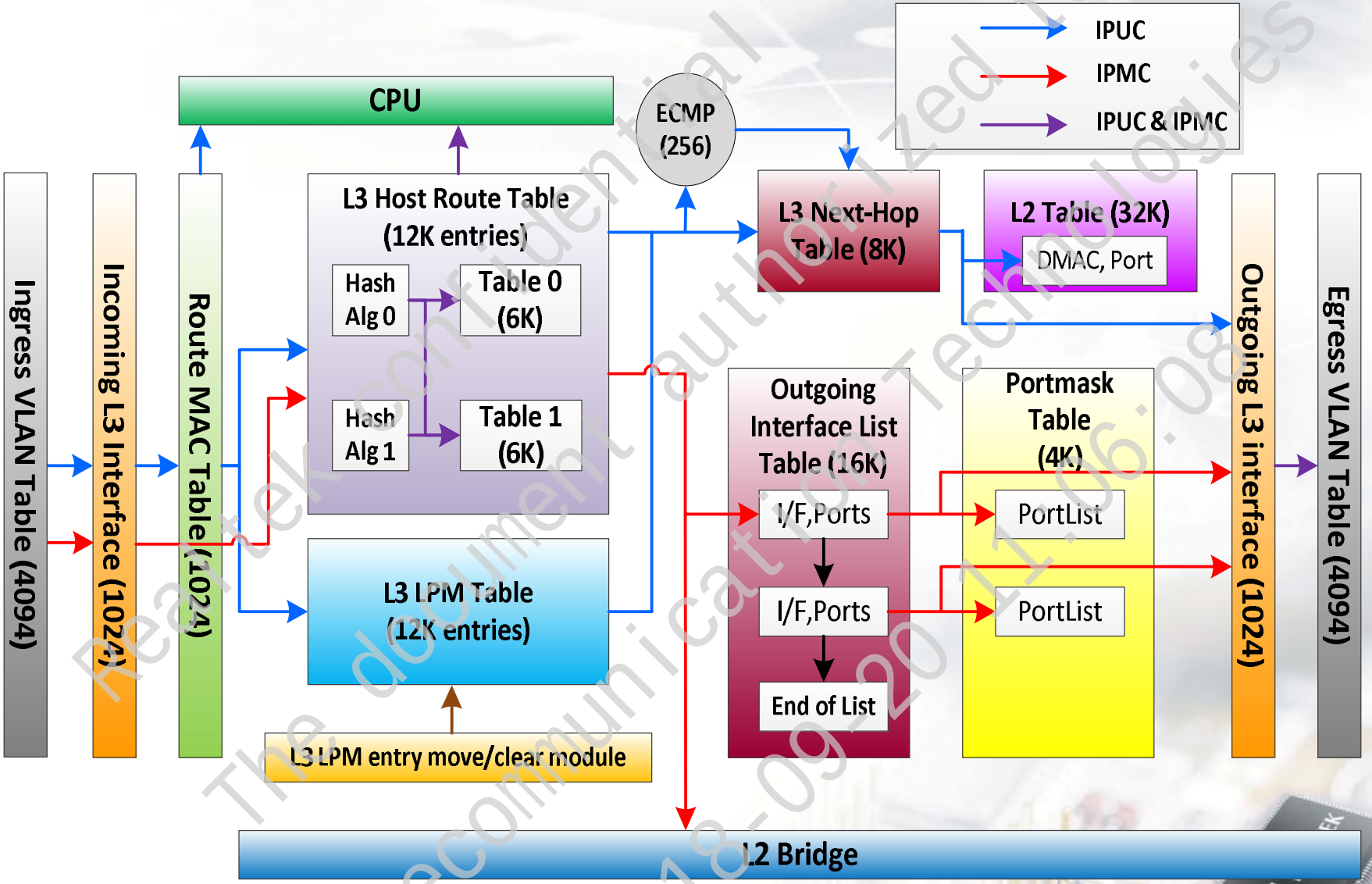
Overview (2/2)

- Multicast Routing
  - 1K L3 interfaces
    - 256 VRF (VPN/Virtual Routing & Forwarding) instances
    - TTL scope
  - 2 rounds lookup of L3 table:
    - 1st round – (VRF, SIP, GIP, VID)
    - 2nd round – (VRF, *, GIP, VID)
  - 2-left 6-way host table (shared with IPUC)
    - 6K IPv4 multicast entries
    - 2K IPv6 multicast entries
  - 16K L3 OIL (Outgoing Interface List) nodes for IPMC replication
  - 4K multicast destination port-mask entries
  - IP-based bridging support

Agenda

- L3 Block Diagram
- Interface (Incoming I/F, IIF)
- Router MAC Table
- Pre-Routing
- Host & LPM Tables
  - Unicast Entry
- Unicast Routing
  - Next-hop Entry
  - ECMP Group Entry
- Interface (Outgoing I/F, OIF)
- IP Unicast Routing
  - Policy-based Routing (PBR)
  - uRPF
- IPMC Group Table
  - Multicast Entry
- IP Multicast Bridging & Routing

L3 Block Diagram



L3 Interface (Incoming I/F, IIF)

- 1K interfaces (shared with Tunnel Interface)
- Ingress L3 Interface Decision (Tunnel → IVC → VLAN)
  - Tunnel interface (tunnel termination)
  - Ingress VLAN conversion (IVC), qualified by { Port and/or VLAN }
  - VLAN-based interface (default)
- Properties
  - IPv4/IPv6 unicast/multicast routing ability configuration
  - VRF_ID (256 independent tables)
  - IP Unicast options:
    - uRPF
  - IP Multicast options:
    - Reserved multicast addresses (forwarding actions)

L3 Router MAC Table

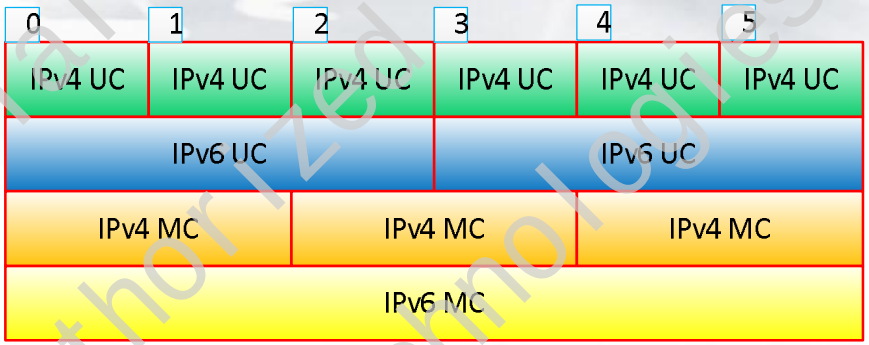
- 1K Router MAC entries
- Associated with L3_IIF (normally, one-to-one mapping)
  - Match field:
    - L3 Incoming Interface ID (L3_IIF)
    - DMAC
    - Port ID or Trunk ID
  - Action field:
    - L3 process (default)
    - Trap

L3 Pre-Routing

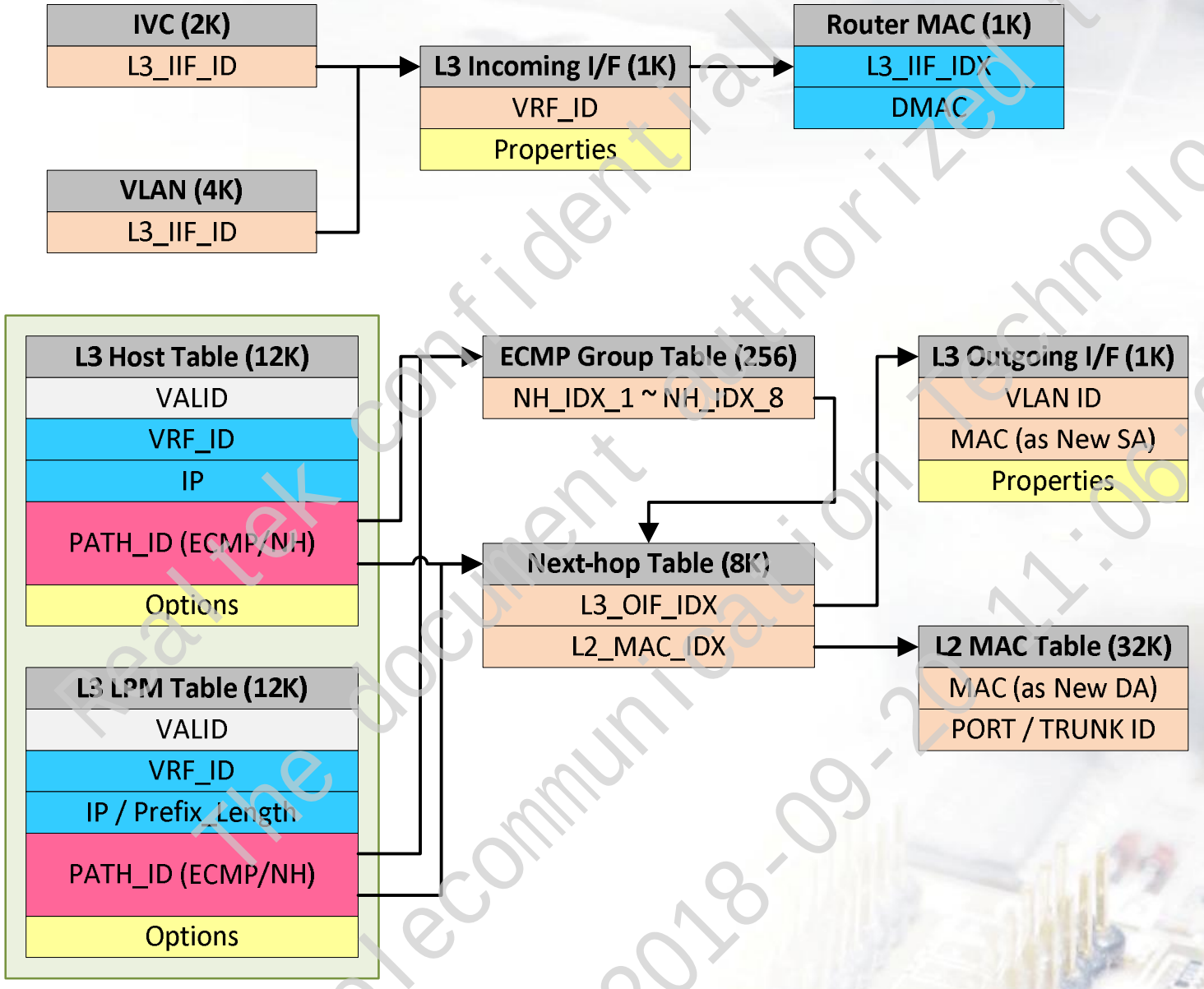
- L3 Pre-Routing Process (before looking up L3 host/LPM tables)
- Global IPv4/IPv6 unicast/multicast routing ability
- DMAC check (trigger to do IP routing processing)
  - Unicast DA: must hit Router MAC entry of L3 Incoming I/F
  - Multicast DA: check if it's mapped to packet's DIP address as multicast (01-00-5E-xx-xx-xx for IPv4, 33-33-xx-xx-xx-xx for IPv6)
- IP Header Validation:
  - IP Header: Length, Version, IP Checksum
  - Illegal IP address check (including SIP and DIP)
  - DIP/DMAC mismatch check (for IPMC packet)
  - Header Option/Extension handling (forwarding option)
    - IPv4 option, IPv6 hop-by-hop and routing extension headers.
- Lookup Priority: (high → low)
  - Host route → LPM/Net route

L3 Table – Host & LPM/Route

- L3 Host Table (SRAM)
  - 12K host entries
  - Hash-based IPUC/IPMC lookups
  - IPUC host entry
  - IPMC group entry
- L3 LPM Table (TCAM)
  - 12K LPM entries
  - LPM (Longest Prefix Match) route entry
  - IPUC LPM lookup (network route)
  - IPUC host entry (for host table full or hash collision)
- Logical entry types:
  - IPv4 Unicast (consumes 1 physical entry, up to 12K IPv4 UC entries)
  - IPv6 Unicast (consumes 3 physical entries, up to 4K IPv6 UC entries)
  - IPv4 Multicast (consumes 2 physical entries, up to 6K IPv4 MC entries)
  - IPv6 Multicast (consumes 6 physical entries, up to 2K IPv6 MC entries)



L3 Unicast Routing – Table Relationships



L3 Table – Host & LPM (Unicast Routing)

- L3 Host/LPM Table lookups (2 rounds for each packet)
  - 1st round – (VRF, SIP) lookup (uRPF check)
  - 2nd round – (VRF, DIP) lookup (routing)
- L3 Host Table (2-left 6-way hash table)
  - 2 tables, each has 6K physical entries (total: 2 tables * 6 ways * 1K entries = 12K)
  - 2 different hash algorithms
    - Each table can specify a hash-algorithm for UC and MC separately
- L3 LPM Table (used for Net-route entries)
- Hit status bit support (for DIP lookup only)
- Per entry configurations:
  - Forwarding action (forward/drop/trap/copy)
  - Destination assignment
    - Null interface (discard the packet)
    - Path ID assignment (ECMP group or Next-hop entry)
  - TTL – 1 (Hop-limit – 1)
  - MTU check
  - Internal priority assignment (optional)

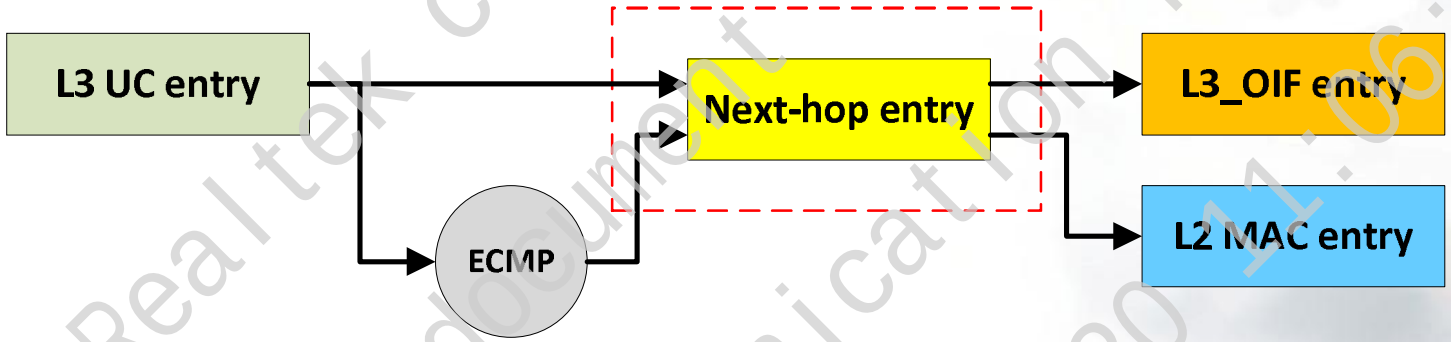
L3 Unicast Routing – Table Relationships

LPM Table – H/W Lookup/Move/Clear entries

- Hardware-aided operations:
  - Lookup (return the first matched entry)
  - Move a bulk of entries (SOURCE, DESTINATION, LENGTH)
    - Copy entries from SOURCE to DESTINATION, then remove the original entries.
    - Driver should keep entry unbroken (moving entries by rows is always safe)
  - Clear a bulk of entries (START, LENGTH)
- SDK uses H/W-accelerate to improve the performance of table maintenance
  - Check entry existence (lookup)
  - Insert a new entry in the middle (move + add)
  - Delete an entry from the table (clear + move)
  - Remove many entries at one time (clear)

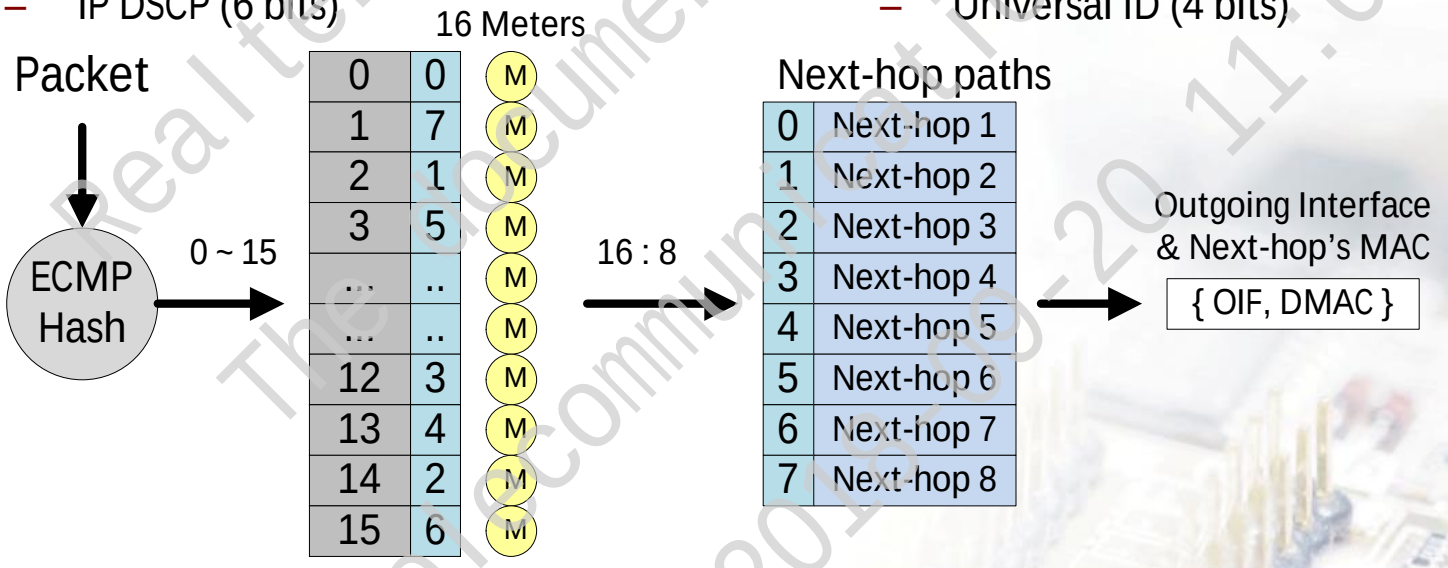
L3 Unicast Routing – Next-hop

- 8K next-hop entries
- Indications:
  - L3 Outgoing Interface (L3_OIF)
  - L2 MAC entry index
    - Associated with L2 MAC entry (as new DMAC)
    - Destination Port (dynamic learn, or static entry)



L3 Unicast Routing - ECMP

- 256 ECMP (Equal-cost multi-path) groups
  - 8 next-hop paths (maximum)
  - 4-bit hash result (16 hash results to 8 next-hop paths)
  - Per group metering support (shared with ACLs) for Dynamic Load Balancing (DLB)
    - consumes 16 meters at one time (for one ECMP group)
- Hash Keys (multiple selection)
  - Rx Port ID
  - Rx Trunk ID
  - IPv4/IPv6 Source IP (rotatable)
  - IPv4/IPv6 Destination IP (rotatable)
  - IP DSCP (6 bits)
  - IPv6 Flow Label (20 bits)
  - IP Protocol
  - L4 Source Port (rotatable)
  - L4 Destination Port (rotatable)
  - Universal ID (4 bits)



L3 Interface (Outgoing I/F, OIF)

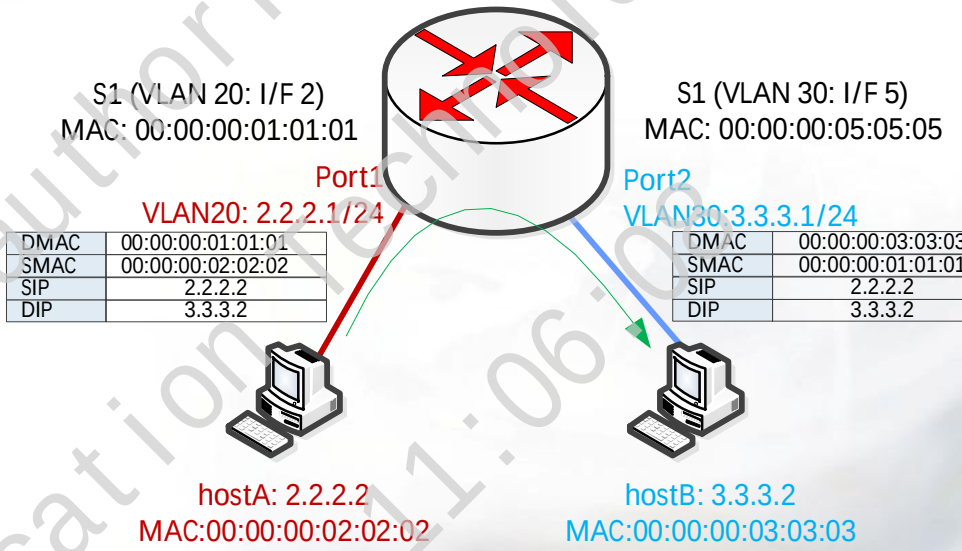
- 1K interfaces (shared with Tunnel Interface)
- Types:
  - Normal L3 interface
  - Tunnel interface (Encapsulation of Tunnel Header)
- Properties
  - Destination VLAN ID
  - MAC address (used for Encapsulation as Source MAC)
  - MTU value (16 global entries)
  - IP Unicast options:
    - ICMP redirect (action: forward/trap/drop/copy) for IPv4, IPv6
    - PBR-based ICMP redirect support for IPv4, IPv6
  - IP Multicast options:
    - IPv4 TTL and IPv6 Hop-limit Scopes (thresholds)

L3 IP Unicast Routing

- Lookup
  - Lookup Router MAC table via DMAC
  - L3 host/LPM tables will be searched
  - uRPF check & DIP routing
- Action
  - Assign outgoing port
  - Modify DMAC
  - Modify forward VID
  - Modify SMAC
  - TTL check and TTL decrement
  - MTU check
  - ICMP redirect check

Router MAC Table

I/F	MAC Address	ACTION
1	00:00:00:01:01:01	L3
5	00:00:00:05:05:05	L3



L3 Host Routing Table

IP	NEXT_HOP_IDX
2.2.2.2	1
3.3.3.2	2

L3 Next hop Table

DMAC_IDX	OUT_INTF_IDX
1251	1
6276	5

L2 MAC Table

DPORT	DMAC_IDX
1	1
2	2

Nexthop MAC Table

MAC Address
00:00:00:02:02:02
00:00:00:03:03:03

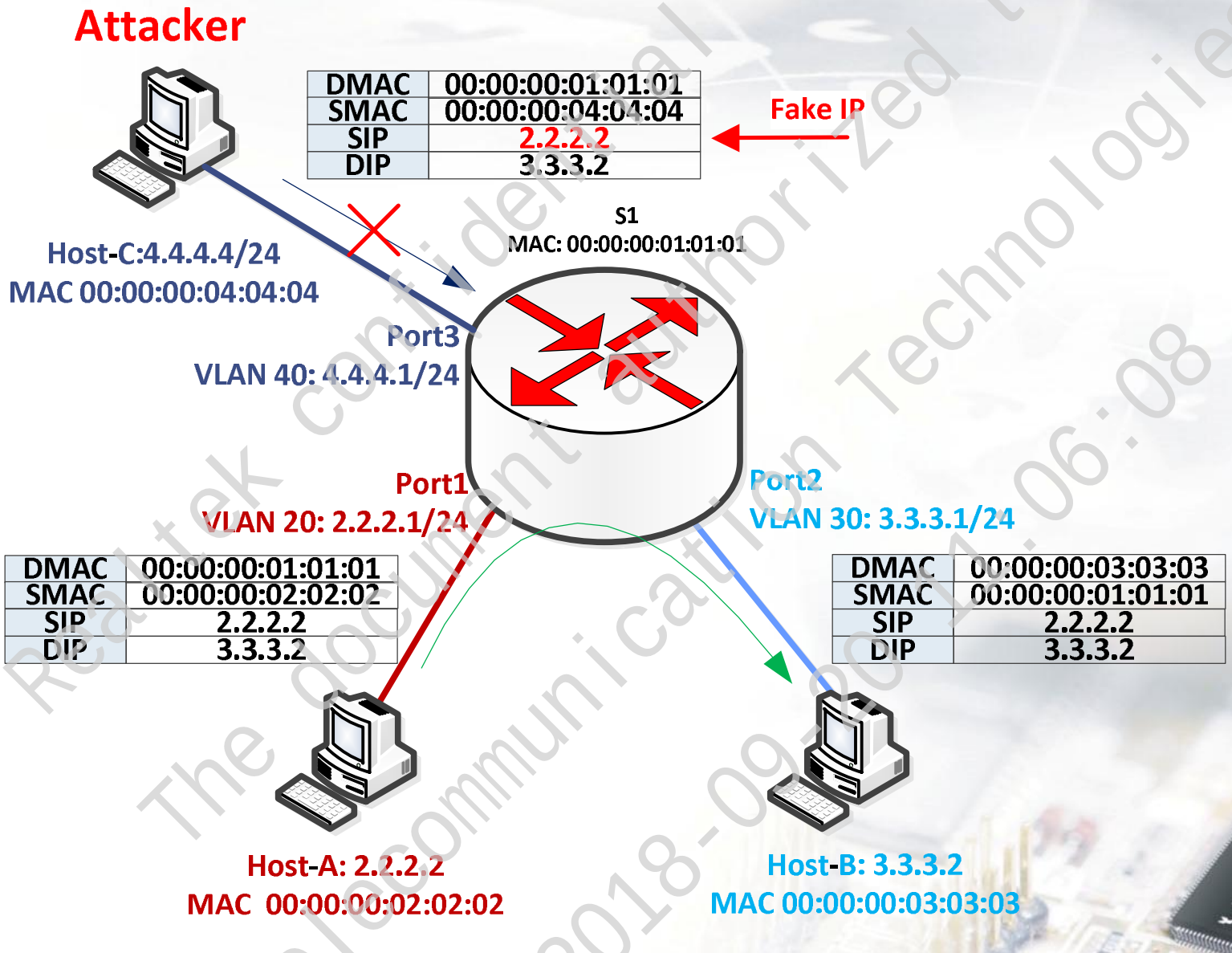
Outgoing L3 interface Table

DST_VID	SMAC_ADDR
20	00:00:00:01:01:01
30	00:00:00:05:05:05

L3 Unicast Routing - Policy-Based Routing

- PBR is supported by VLAN ACL (VACL)
  - Force Unicast Routing
  - Default Unicast Routing
- Policies can be based on:
  - SIP/DIP
  - Layer 3/4 protocols
  - L3 packet length
- Unicast Routing Precedence: (from high to low)
  - PBR: Force Unicast Routing (assigned by VACL)
  - Non-default Route in L3 Routing Table
  - PBR: Default Unicast Routing (assigned by VACL)
  - Default Route in L3 Routing Table

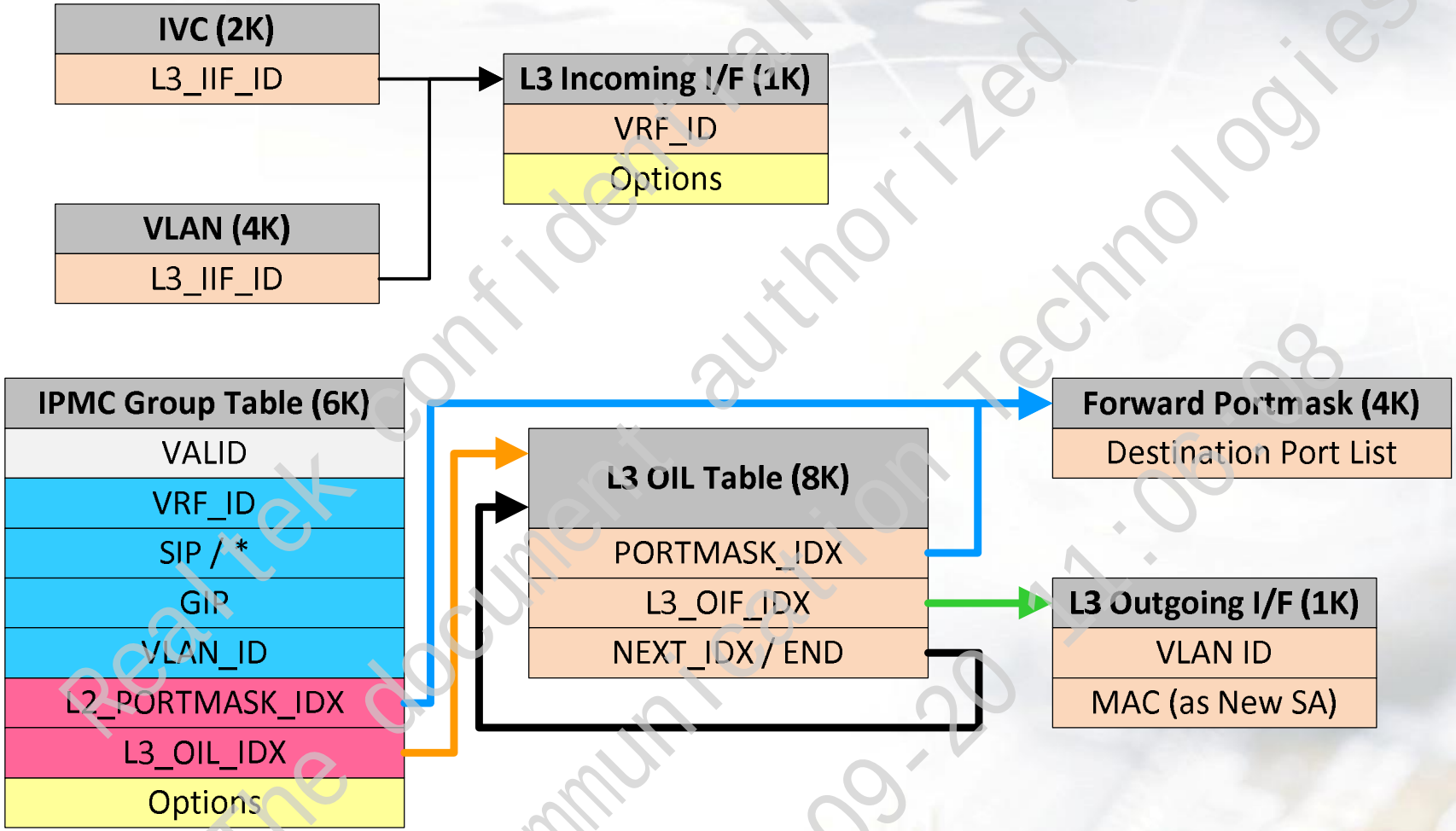
L3 Unicast Routing – uRPF check function



L3 Unicast Routing – uRPF

- Global Configuration
  - Port-based or Interface-based
- uRPF Configurations:
  - Enable/Disable uRPF check
  - Check mode:
    - Strict mode (SIP + L3 interface)
    - Loose mode (SIP only)
  - Include or exclude default route
  - Check failed action: forward/drop/trap/copy
- IACL permit/drop uRPF check failed packets with uRPF result as key

L3 Multicast Routing – Table Relationships



L3 Multicast Entry (Routing Table)

IPMC Group Table (shared with L3 IPUC host table)

Key				Data	
VRF	Source IP	Group IP	VID	L2_PM_IDX	L3_OIL_IDX
1	192.168.100.1	224.111.1.1	10	10	101
1	255.255.255.255	224.111.1.1	10	-	222
1	192.168.200.1	239.222.1.100	20	20	151
1	255.255.255.255	239.222.1.100	20	-	372
2	192.168.100.1	224.111.1.1	10	-	534
2	255.255.255.255	224.111.1.1	10	50	765

Forwarding Port-mask Table (4K)

IDX	Destination Portmask
10	00010110101 000110010
20	00000000001 000110100
50	00101000101 011100110
65	11001110010 101011100
70	00000000001 000110100
78	00010110101 000110010
82	00000000001 000110100
94	11001110010 101011100
99	00101000101 011100110

L3 Outgoing Interface List (16K)

IDX	L3_OIF	L3_PM_IDX	NEXT
101	100	65	112
112	200	78	123
123	300	94	-

L3 Outgoing Interface (1K)

IDX	VID	MAC
100	11	00:11:11:11:11:11
200	12	00:22:22:22:22:22
300	13	00:33:33:33:33:33

- Lookup Priority: (high → low)
  - (VRF, SIP, GIP, VID) → (VRF, *, GIP, VID)

IP Multicast Routing – IPMC Group Table Lookup

- 2 rounds lookup
  - IPMC lookup priority:
  - (VRF, SIP, GIP, VID) > (VRF, *, GIP, VID)

L3 IPMC Table

No.	VRF	SIP	GIP	VID	L2_PM_IDX	L3_OIL_IDX
1	5	1.2.3.4	224.1.1.1	10	1238	1
2	5	255.255.255.255	224.1.1.1	10	5842	10
3	5	1.2.3.4	224.1.1.1	20	2045	20
4	5	255.255.255.255	224.1.1.1	20	8652	30

Packet flow (VRF = 5)

VID	SIP	GIP
10	1.2.3.4	224.1.1.1
10	6.6.6.6	224.1.1.1
20	1.2.3.4	224.1.1.1
20	6.6.6.6	224.1.1.1

Round 1

Hit
1
N
3
N

Round 2

Hit
2
2
4
4

Final result

Hit
1
2
3
4

L3 Multicast Routing – Host Table

- L3 Host Table Lookups (2 rounds for each packet)
  - 1st round – (VRF, SIP, GIP, VID) group lookup (including SIP)
  - 2nd round – (VRF, *, GIP, VID) group lookup (excluding SIP)
    - Use all-1s SIP to represent * (don't care)
- Per entry configurations:
  - Forwarding action (forward/drop/trap/copy)
  - OIL (Outgoing Interface List) assignment
  - Min TTL check (optional, copy to CPU)
  - Max MTU check (optional, copy to CPU)
  - Internal priority assignment (optional)

L3 Multicast Routing-related Tables

- OIL (Outgoing Interface List) Table
  - 16K OIL node entries (used for IPMC replication)
  - Indexed by IPMC group entry (using L3 host table)
  - Per node configuration:
    - TTL – 1 (optional)
    - MTU check (optional)
    - L3 Outgoing Interface index (SMAC, Destination VLAN)
    - Forwarding Port-mask entry index (Destination port list)
- Multicast Forwarding Port-mask Table
  - 4K port-mask entries
  - As an L2 bridge destination port-mask (indexed by IPMC group entry)
  - As an L3 route destination port-mask (indexed by OIL node entry)
- IPMC Traffic Monitoring Table
  - 4 sets of monitoring counter
  - Counting mode: packets (frame numbers) / bytes (IP payload length)
  - Specify IPMC group entry

IPMC Bridge & Route

- Application
  - IGMP v1/v2/v3 Snooping support
  - MLD v1/v2 Snooping support
  - MVR support
  - Support 2 bridge modes
    - IP-based & MAC-based
  - DVMRP/PIM-DM/PIM-SM/MOSPF support
  - IPMC L2/L3 Lookup Miss Action drop/trap priority
    - L3 lookup miss trap (reason) > l2 lookup miss trap > l2 lookup miss drop

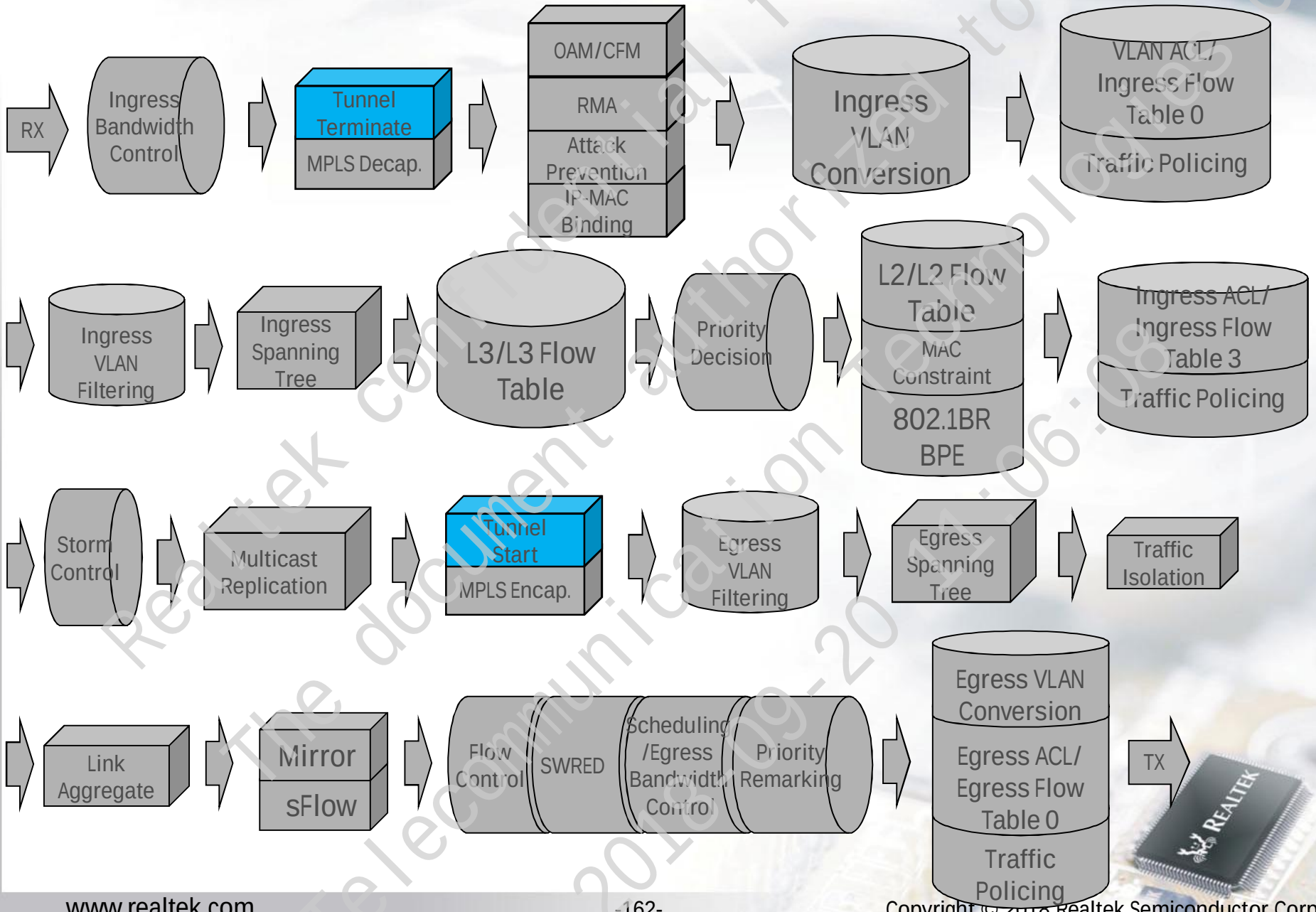
L3-related Counters

- L3 Interface-based Counters
  - RX
    - Octets (bytes)
    - Unicast Packets
    - Multicast Packets
    - Broadcast Packets
    - Discards
  - TX
    - Octets (bytes)
    - Unicast Packets
    - Multicast Packets
    - Broadcast Packets
    - Discards

REALTEK
Realtek Semiconductor Corp.

IP Tunnel

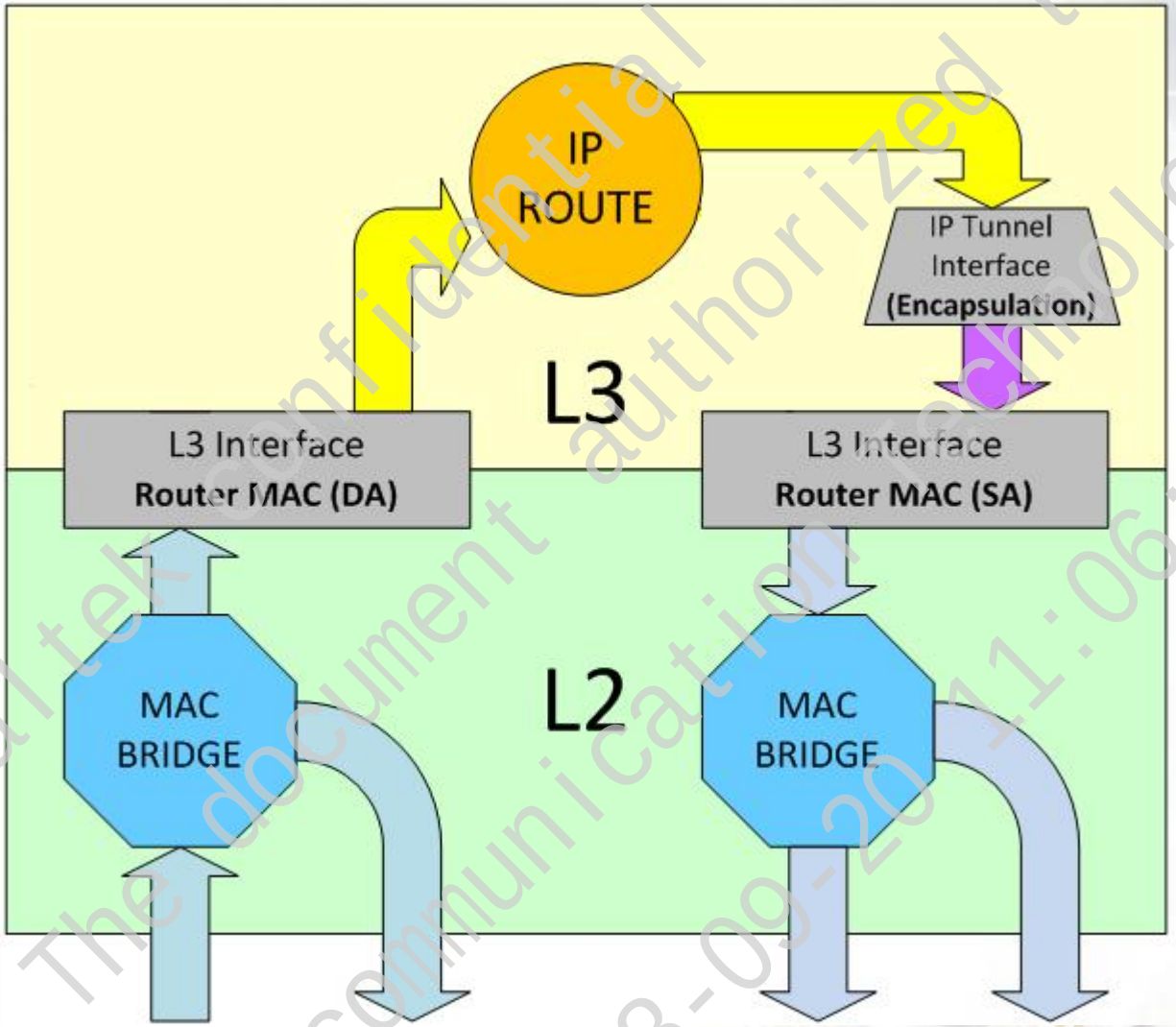
Packet Processing Pipeline



Overview

- 384 tunnel entries (both decapsulation and encapsulation)
  - 384 IPv4 or 128 IPv6 tunnels (IPv6 tunnel consumes 3 physical entries)
- Supported Tunnel Types
  - Manually configure tunnel
    - IPv4/IPv6 over IPv4 (RFC 1853)
    - IPv4/IPv6 over IPv6 (RFC 2473)
    - IP GRE (IPv4/IPv6 over GRE, RFC 1701/1702/2784/2890)
  - Automatic tunnel (IPv6 over IPv4)
    - ISATAP (RFC 4214)
    - 6to4 (RFC 3056)
    - 6rd (RFC 5969)

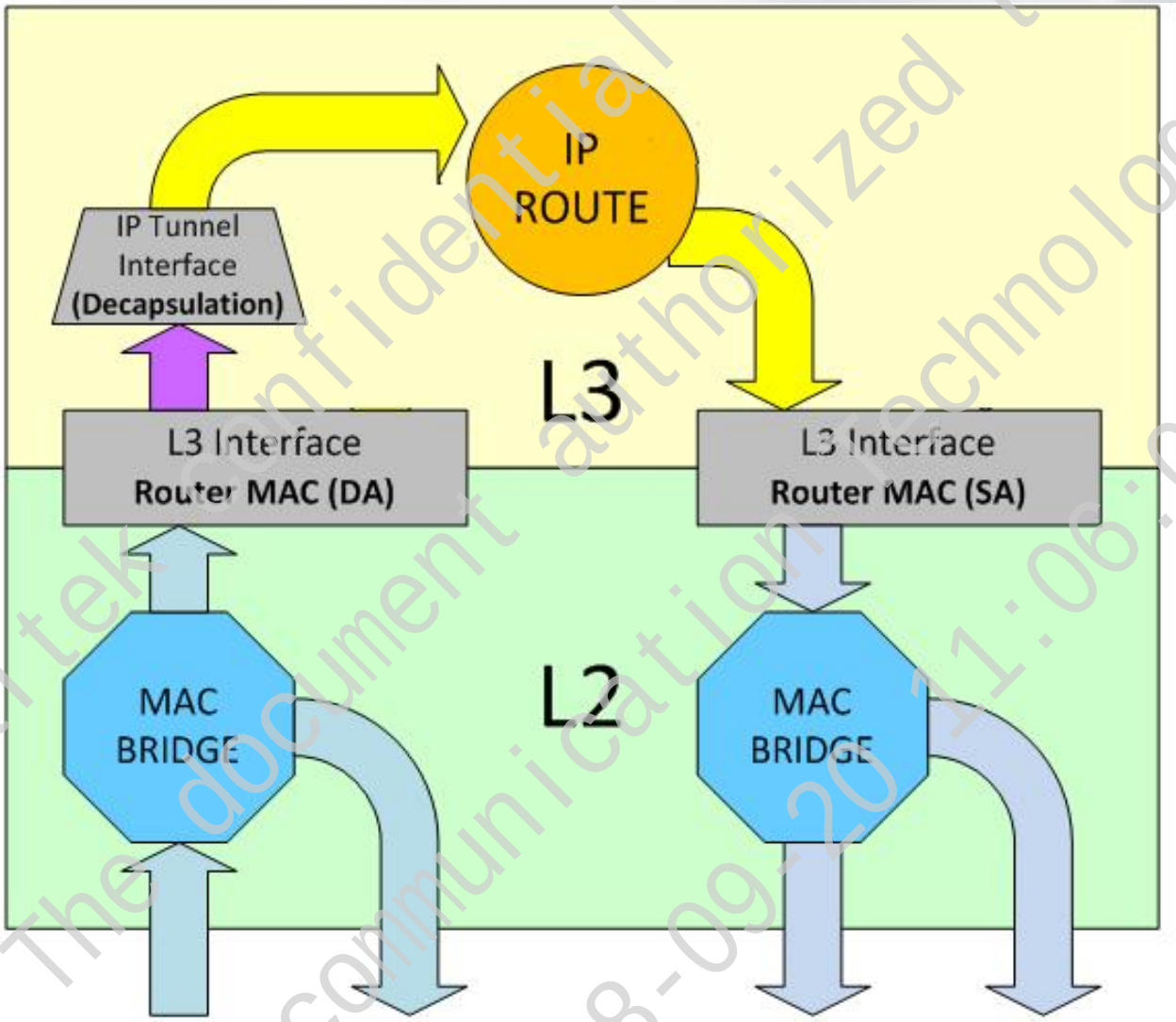
IP Tunneling – Encapsulation



IP Tunnel Encapsulation

- 384 Encap. entries indexed by routing
- Per entry (ingress tunnel interface) configuration
  - Tunnel Type : Configured / ISATAP / 6to4 / 6RD / GRE
  - SIP: usually set as router IP for egress
  - DIP: for configured tunnel, no need assign for automatic tunnel
  - L3 outgoing interface & Nexthop DA
  - TTL
    - TTL/Hop-limit assignment or inherit passenger's TTL/HL value
    - TTL_DEC (TTL – 1) option
  - 64 profiles for DSCP: Assigned DSCP/ inherit / internal priority remark
  - GRE key assignment (optional)
  - IPv4 DONT-FRAGMENT bit assignment or inherit passenger's for IPv4 tunnel
  - IPv6 Flow Label assignment (transport by IPv6)
- Tunnel Encap. check
  - TTL failed (== 0) action: drop/trap
  - MTU failed action: drop/trap

IP Tunneling – Decapsulation

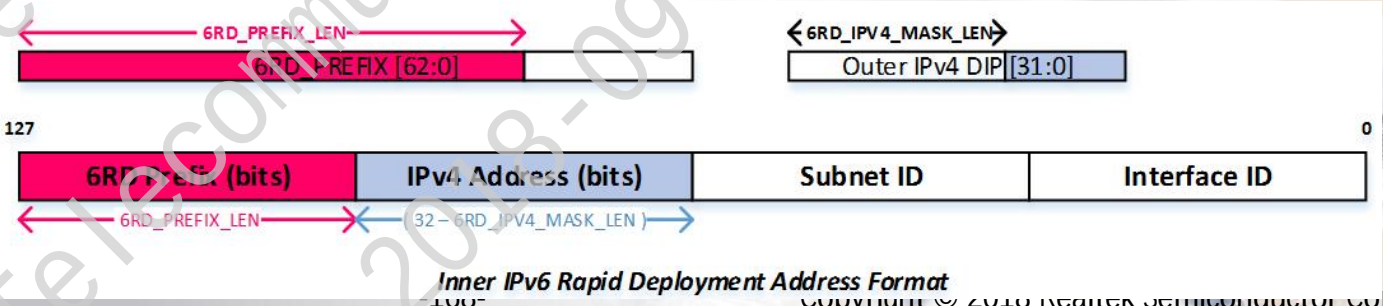


IP Tunnel Decapsulation

- 384 Decap. entries
- Per entry (ingress tunnel interface) configuration
  - Ingress interface ID
  - DIP(Local IP) and SIP(Remote IP) configuration
  - GRE 32-bit key option header check (optional for GRE tunnel)
  - Passenger type : IPv4/IPv6/GRE
  - Permit passenger's IP version: IPv4 and/or IPv6 (enable/disable: permit/deny)
  - Inherit carrier's TTL
  - Qos
    - Inherit carrier's DSCP
    - Priority Selection table assignment
    - Tunnel-based internal priority assignment
    - Don't touch passenger's DSCP on Egress Remarking

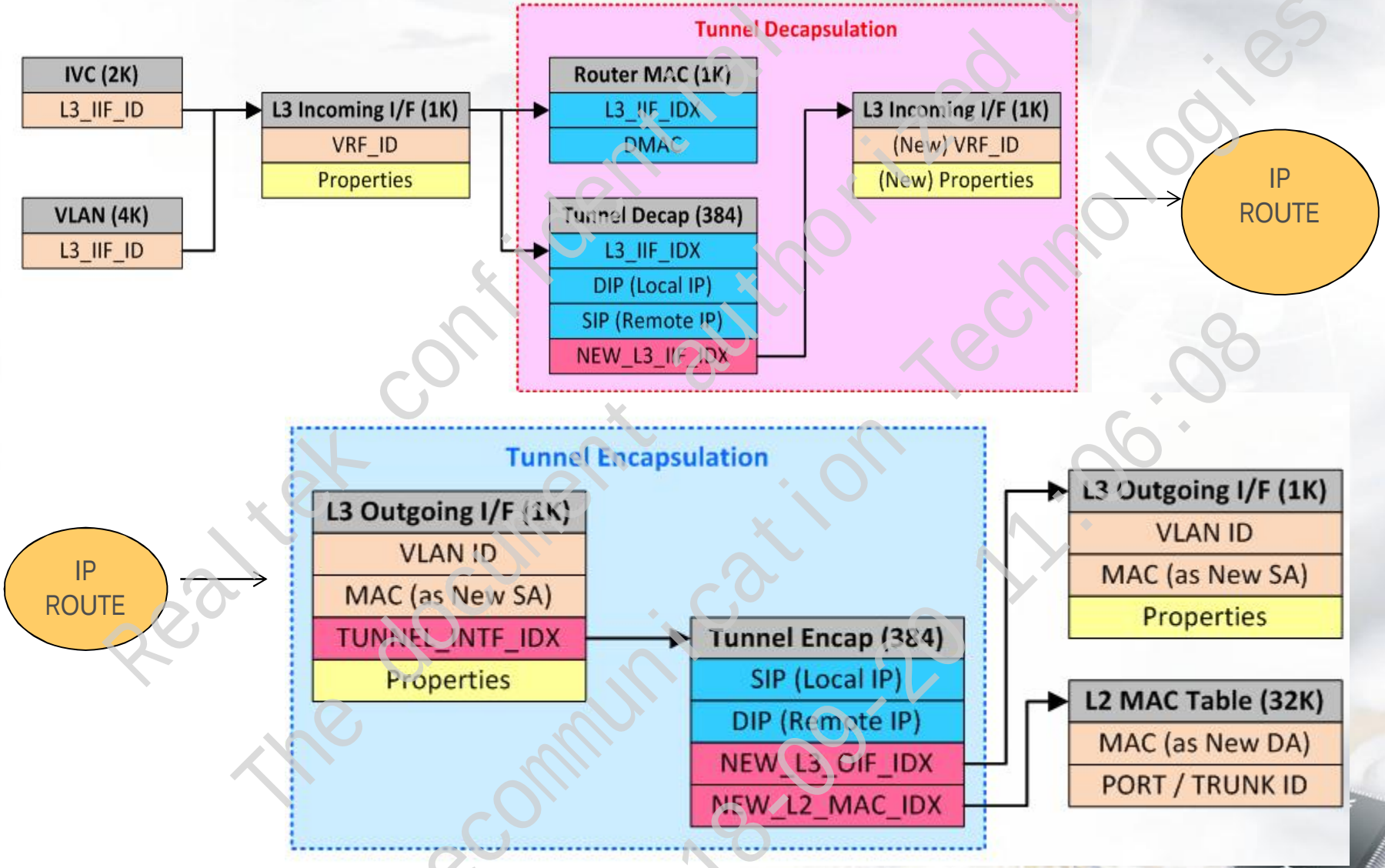
IP Tunnel Decapsulation (cont'd)

- Once tunnel decap. entry hit, below check can be performed
  - Inner SIP check (for IPv6 passengers of configured tunnel)
    - Permit/Deny IPv4-mapped IPv6 addresses (i.e., 0::FFFF:IPv4)
    - Permit/Deny IPv4-compatible IPv6 addresses (i.e., 0::IPv4)
    - Drop/Trap
  - ISATAP SIP type check
    - Outer IPv4: 10/8, 172.16/12, 192.168/16 (private IP)
      Inner IPv6: 0::5EFE:X:X
    - Outer IPv4: Public IP
      Inner IPv6: 0::2005EFE:X:X
    - ISATAP SIP mapping check (if outer IP and inner IP are mapped)
    - Drop/Trap
  - 6to4 SIP and DIP check
    - Outer IPv4[31:0] = Inner IPv6[111:80]
      Inner IPv6 [127:112] = 0x2002
    - Drop/Trap
  - 6RD domain (DIP) check (support 2 domains as maximum)
    - Drop/Trap



Inner IPv6 Rapid Deployment Address Format

IP Tunneling – Decapsulation & Encapsulation



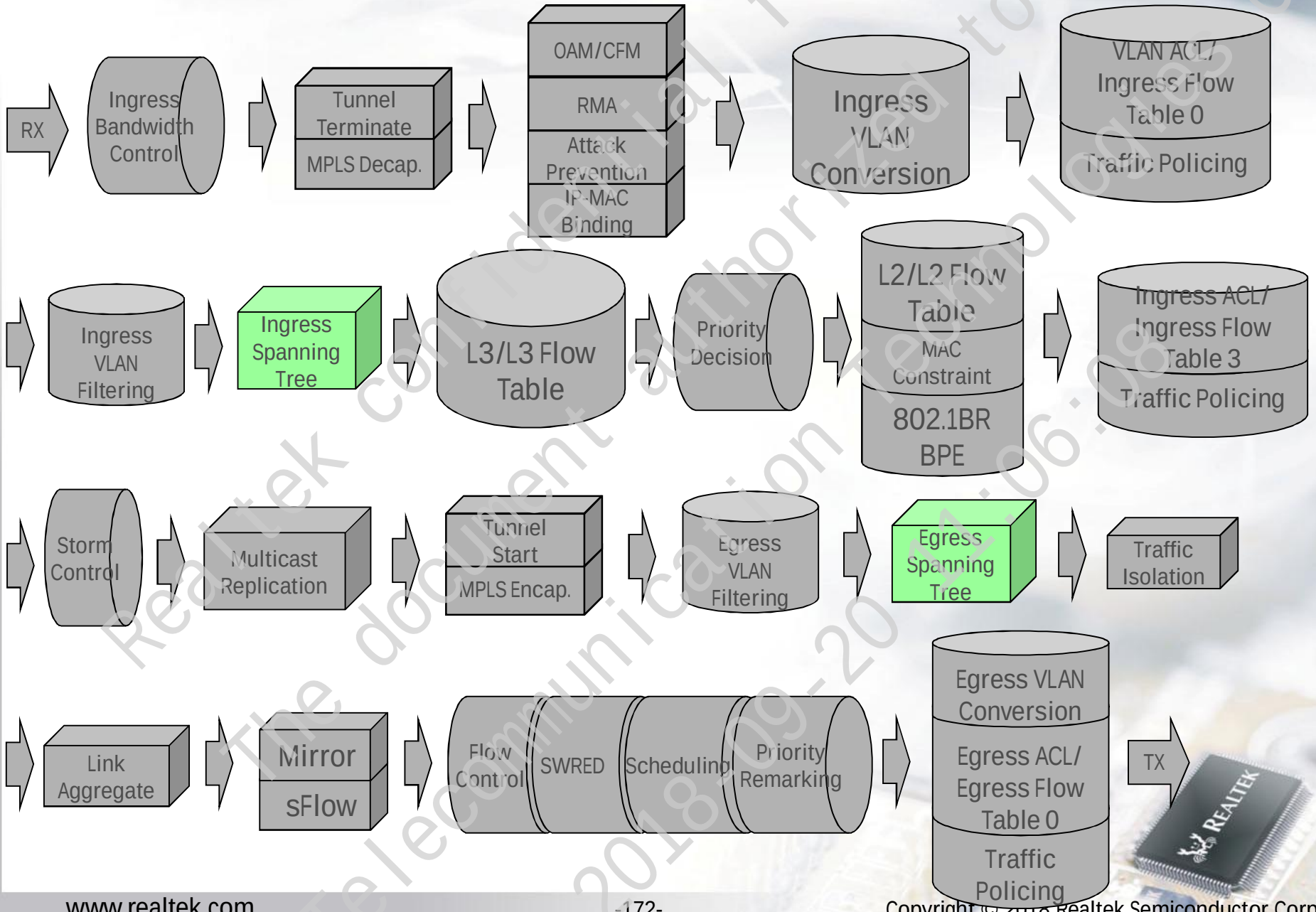
Tunnel-related Counters

- Tunnel entry based counters
  - RX (Decap.)
    - Octets (bytes)
    - Incoming Packets
    - Drops
  - TX (Encap.)
    - Octets (bytes)
    - Outgoing Packets
    - Drops

Realtek Semiconductor Corp.

Spanning Tree

Packet Processing Pipeline



Spanning Tree Overview

- STP (IEEE 802.1D)/RSTP (IEEE 802.1w)/MSTP (IEEE 802.1s)
- 128 instances
- MSTI mode
  - MSTI is either from VLAN table or a fixed value "0" by configuration
- Per port configuration
  - BPDU packet forward action
- Per instance per port
  - STP States
    - Disable
    - Blocking/Listening
    - Learning
    - Forwarding
- Bypass Spanning Tree Filtering
  - CPU-TX Packet
  - Specific RMAs
  - ACL

Spanning Tree Protocol

Port state	RX packet	BPDU (RMA Fwd)	BPDU (RMA Trap)	CPU TX packet (without bypass-STP)	CPU TX packet (bypass-STP)	Learn address	TX packet
Disable	No	No	No	No	No	No	No
Blocking/ Listening	No	No	Yes	No	Yes	No	No
Learning	No	No	Yes	No	Yes	Yes	No
Forwarding	Yes	Yes	Yes	Yes	Yes	Yes	Yes

"bypass-STP" is a field of CPU TX Tag (CPU_TX_TAG.BP_STP)

Bypass Spanning Tree Filtering

- CPU Packet
  - Configure BP_STP field in CPU TX TAG
- Specific RMAs
  - EAPOL
  - 01-80-C2-00-00-02
  - LLDP
  - PTP
  - 01-80-C2-00-00-2X
  - User-Defined RMA

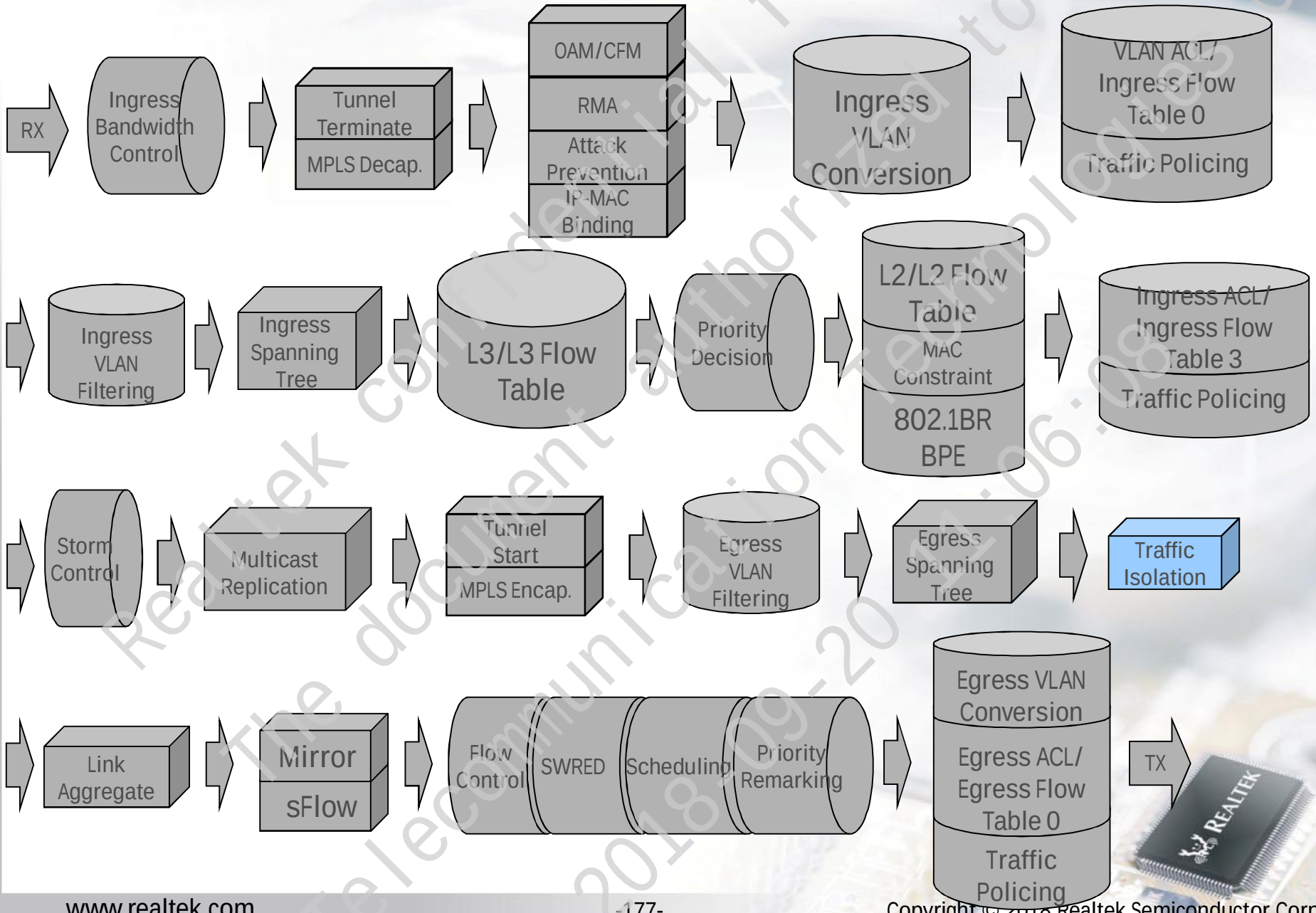
  (Take effect only if RMA action is trapping to CPU)
- ACL
  - Configure bypass Ingress Spanning Tree action

  (Take effect only if forward action is to CPU)

REALTEK
Realtek Semiconductor Corp.

Traffic Isolation

Packet Processing Pipeline



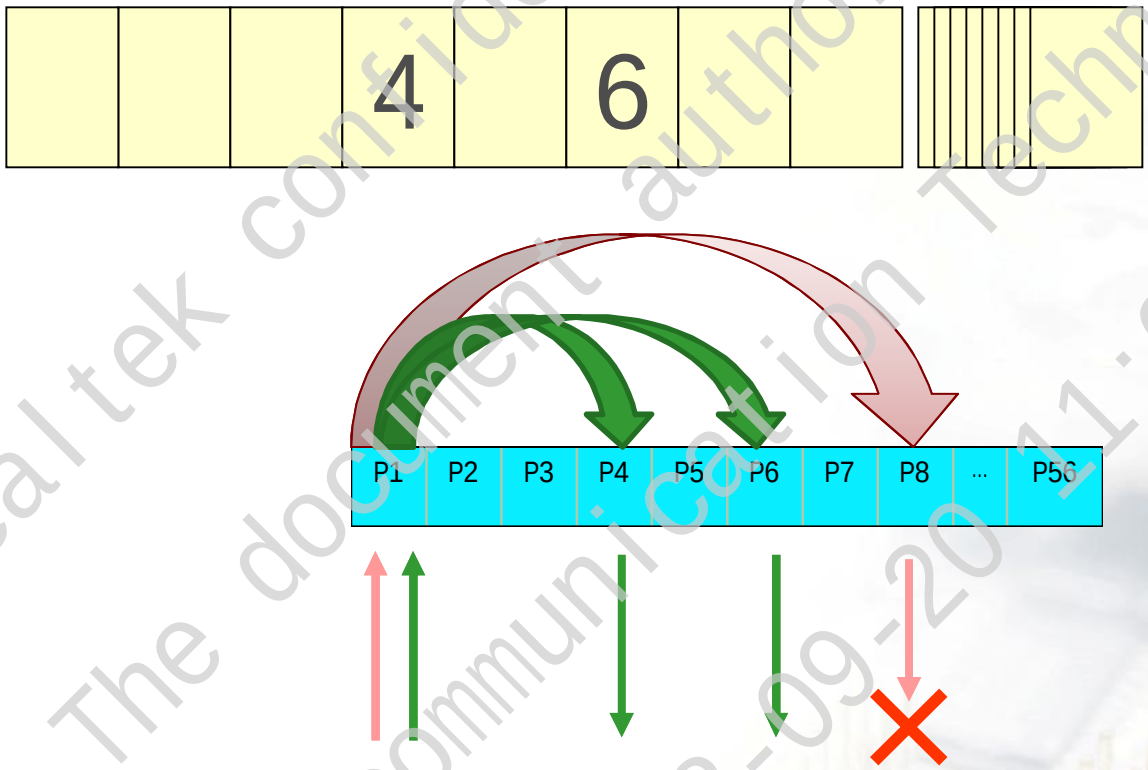
Overview

- Port-based Isolation
  - Per-unit and per-ingress port defines local egress port list.
- VLAN-based Isolation
  - 32 sets
  - Per-set configuration
    - Upper bound VLAN ID
    - Lower bound VLAN ID
    - Downlink port/Uplink port in a portmask
  - VLAN is based on forwarding VLAN
- Routing Traffic with Traffic Isolation

Port-Based Traffic Isolation (1/3)

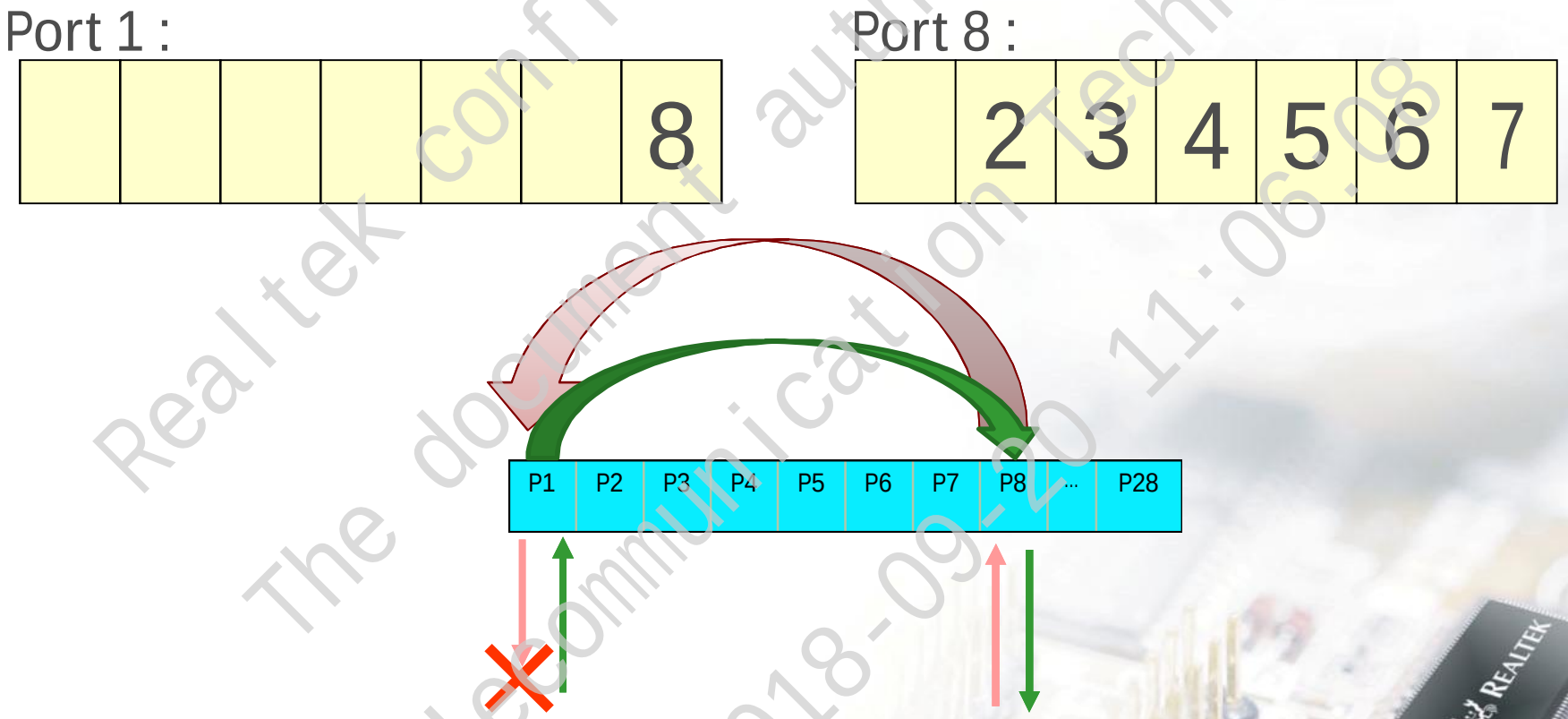
- Control Egress port list

Isolation port list of Port 1 :



Port-Based Traffic Isolation (2/3)

- Asymmetric Transmission

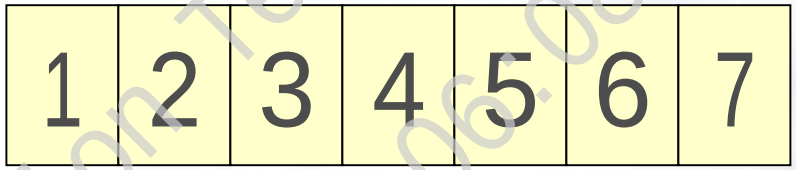


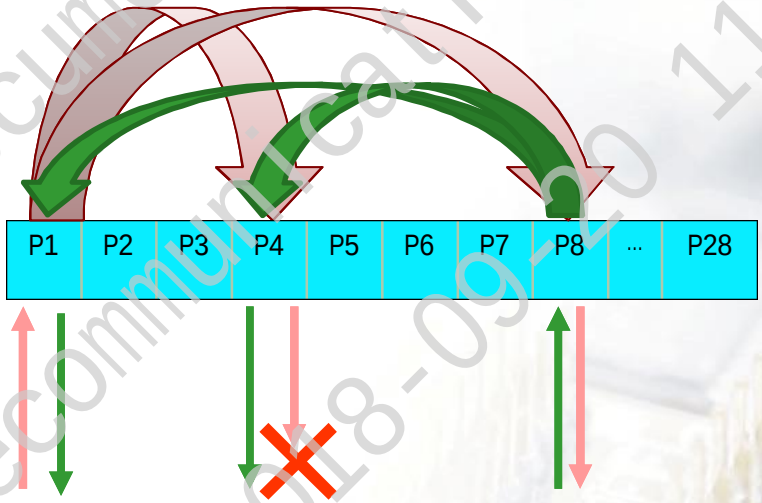
Port-Based Traffic Isolation (3/3)

- Uplink port service:
  - Uplink port can egress to all ports
  - Downlink port can only egress to uplink port
  - Downlink port can not egress to another downlink ports

Port 1 : Downlink Port

Port 8: Uplink Port



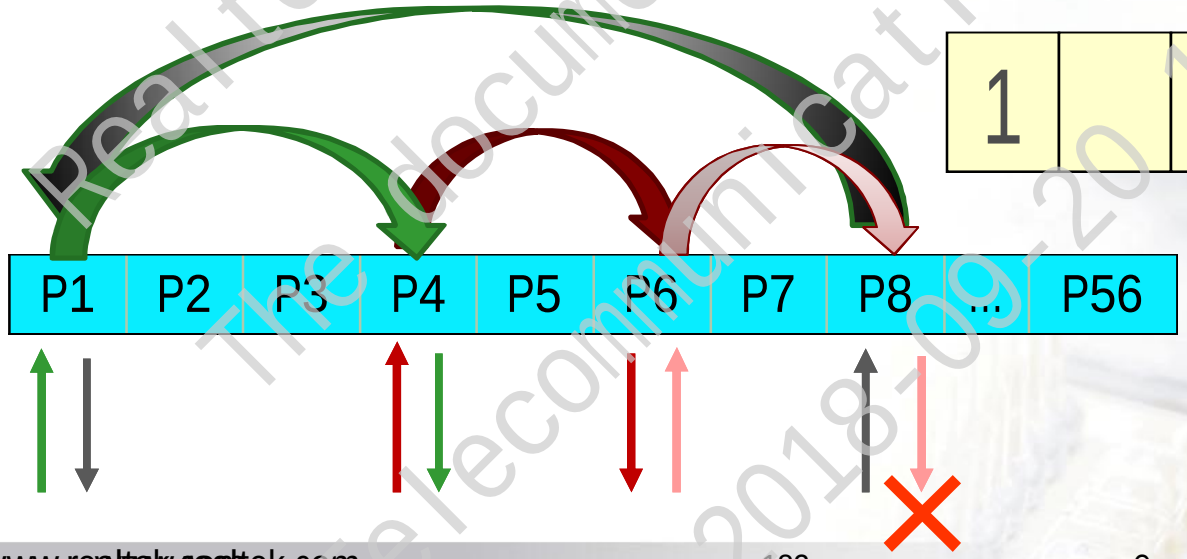


VLAN-Based Traffic Isolation (1/2)

- 32 sets
- Uplink ports can communicate with all ports
- Downlink ports can't communicate with other downlink ports but can communicate with uplink ports
- Per set configuration:
  - Upper bound VLAN ID
  - Lower bound VLAN ID
  - Port role (0: Downlink 1: Uplink Port)

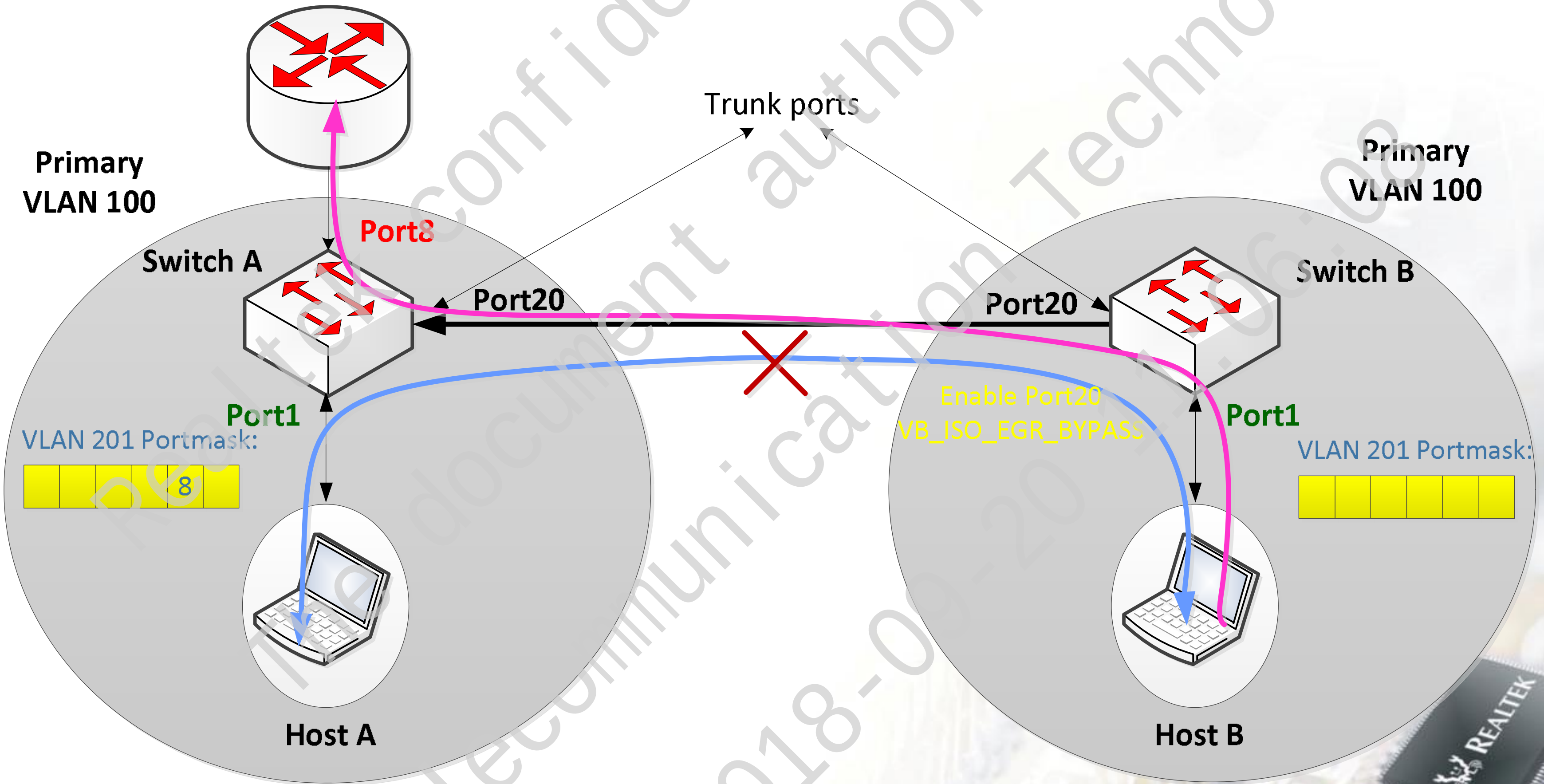
VLAN 1 isolation :
uplink ports: P1, P4:
downlink ports: P6, P8

1			4			



VLAN-Based Traffic Isolation (2/2)

- Bypass VLAN-based isolation in egress direction by per-egress port setting
  - Uplink port : Switch A's port8
  - Downlink port : Switch A's port1 & port20, Switch B's port1 & port20



Port-Based & VLAN-Based Traffic Isolation Hybrid

VLAN 1 isolation :
P1, P4 is uplink ports,
P6, P8 is downlink ports.

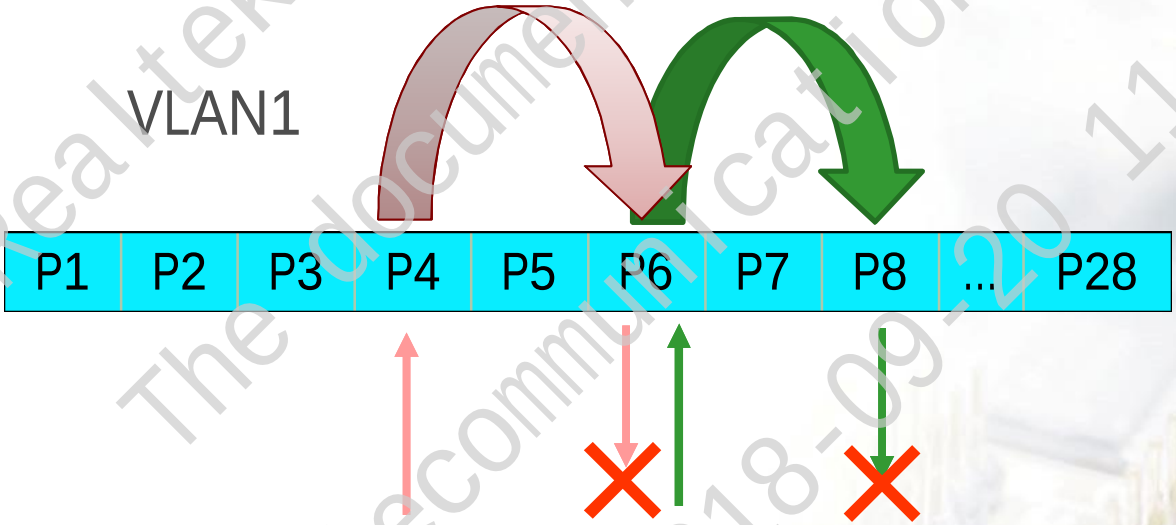
1			4			

Port 4 isolation:

1						8

Port 6 isolation:

1			4			8

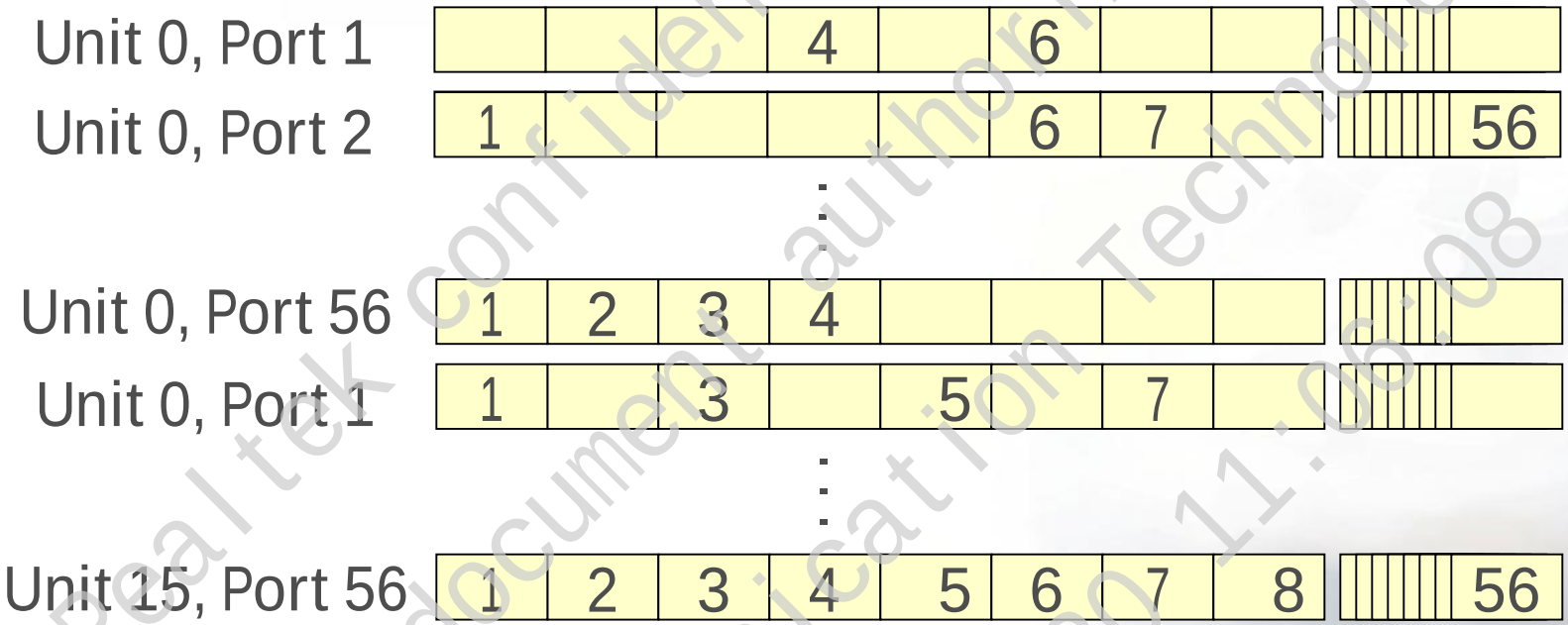


Routing Traffic with Traffic Isolation

- Port-based
  - Routing traffic has global configuration to bypass port-based isolation
- VLAN-based
  - Routing traffic always bypass VLAN-based isolation

Port-Based Traffic Isolation in Stacking

- To support port-based traffic isolation in stacking system, each unit has a table recording the allowed egress port list



- Trunk is supported through packet's physical ingress (unit, port)

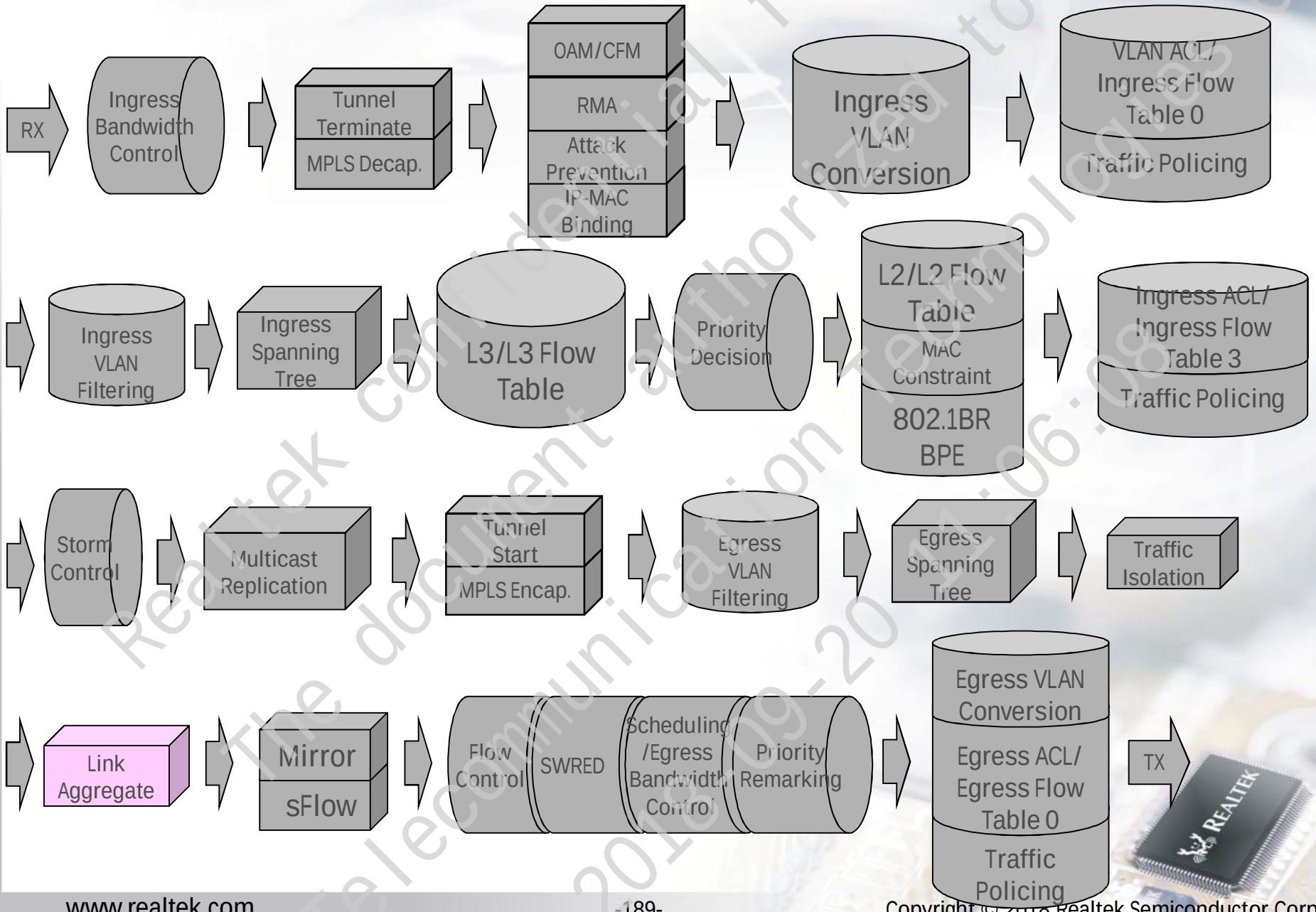
VLAN-Based Traffic Isolation in Stacking

- Port role (uplink or downlink port) is assigned according its ingress port of each packet.
- Port role property is kept when forward to other units
- Find the allowed egress port list through port property in destination unit.

REALTEK
Realtek Semiconductor Corp.

Link Aggregation

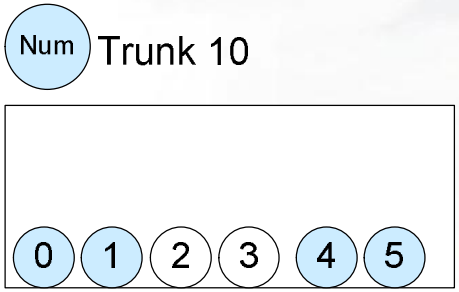
Packet Processing Pipeline



Link Aggregation Overview

- 128 Groups in Stacking Mode
- 26 Groups in Stand-Alone Mode
- 8 Ports Per Group
- Hash Algorithm
  - Selectable L2/L3 Hash Keys
  - Hash Key Shift
- Traffic Separation Mechanism
- Auto Fail-over only in Stand-alone mode
- Local First Load Balance supported in stacking mode

Link Aggregation Member Configuration



- Source port mapping table (Ingress trunk ID table)
  - Decides L2 learnt trunk ID
- Local trunk table
  - The member ports are logically one port for
    - Source port filtering
    - Multi-target packet forwarding
- Trunk egress table
  - Decides the TX candidate ports

Ingress Trunk ID Table		
Port	TRK_VLD	TRK_VLD
0	1	10
1	1	10
2	0	-
3	0	-
4	1	10
5	1	10

Local Trunk Table	
Trunk	Member Ports
10	0, 1, 4, 5

Trunk Egress Table		
Trunk	NUM_TX_Candidate	TX_Candidate_N
10	4	0, 1, 4, 5

LACP Collection and Distribution State Support

- LACP collection state: port can receive packet and packet is learnt on trunk ID but would not forward packet
  - Add port to trunk local table and ingress trunk ID table

Ingress Trunk ID Table		
Port	TRK_VLD	TRK_VLD
0	1	10
1	1	10
2	0	-

Local Trunk Table	
Trunk	Member Ports
10	0, 1

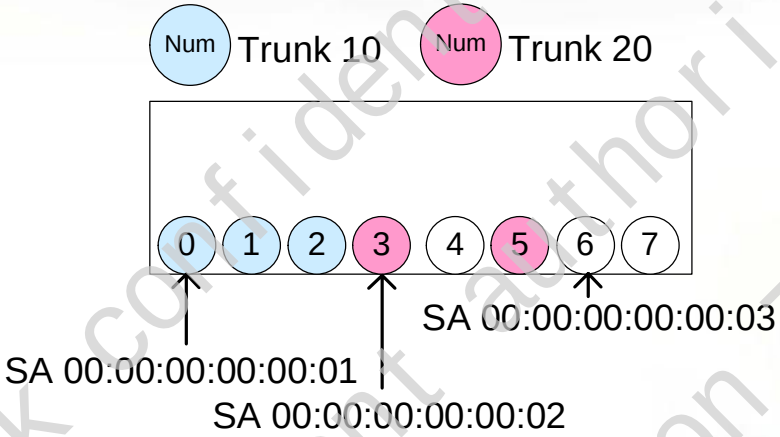
Trunk Egress Table		
Trunk	NUM_TX_ Candidate	TX_Candidate
10	1	0

- LACP distribution state: port can forward packet
  - Add port to trunk egress table

Trunk Egress Table		
Trunk	NUM_TX_ Candidate	TX_Candidate
10	2	0, 1

SA Learning

- Trunk ID is learnt in L2 table for packet received from trunk member port
  - Ingress trunk ID table decides RX port mapped trunk ID

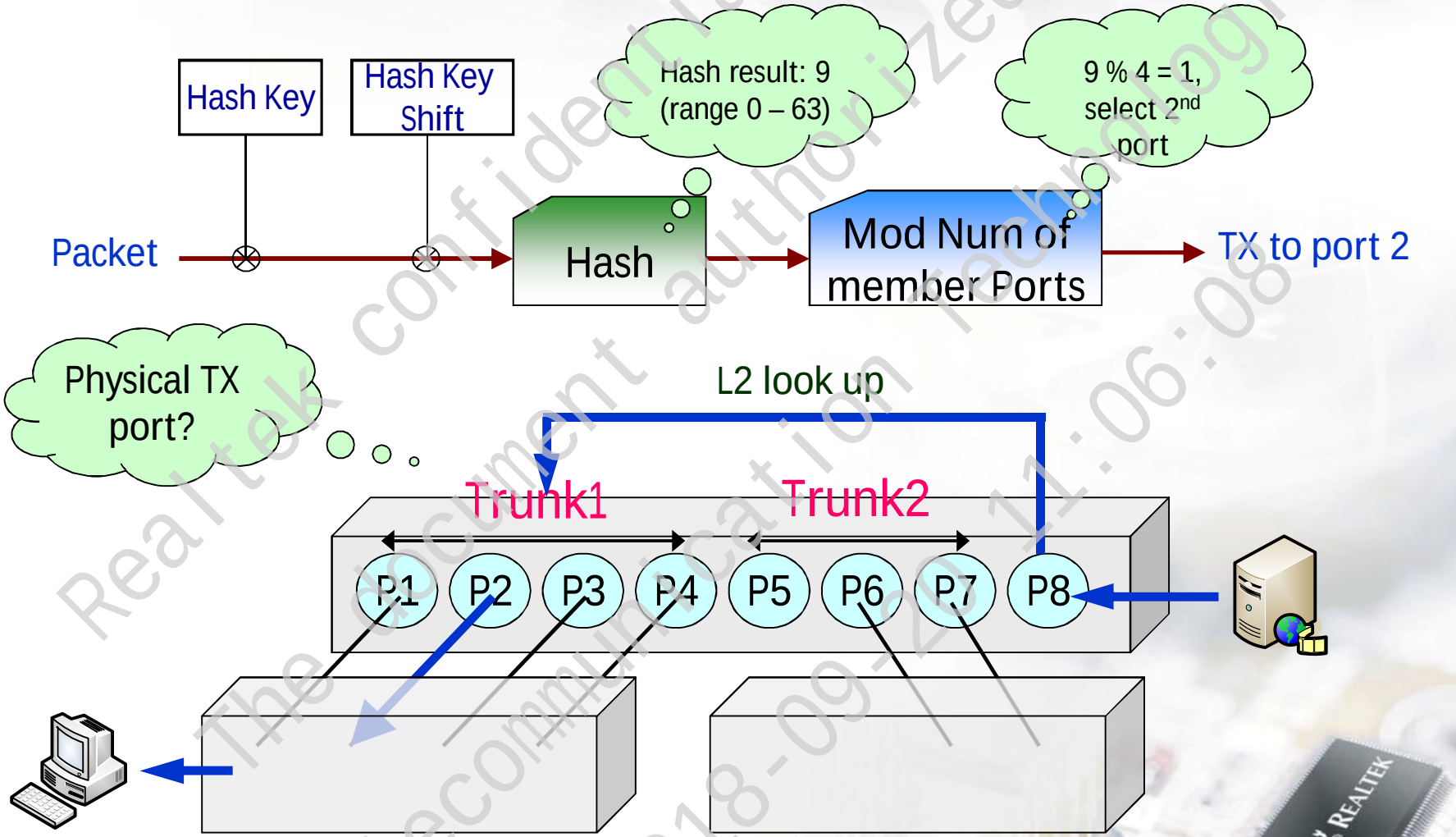


Ingress Trunk ID Table		
Port	TRK_VLD	TRK_VLD
0	1	10
1	1	10
2	1	10
3	1	20
4	0	-
5	1	20
6	0	-
7	0	-
…	…	…

Learnt L2 Table		
SMAC	is_trunk	Port ID/ Trunk ID
00:00:00:00:00:01	TRUE	10
00:00:00:00:00:02	TRUE	20
00:00:00:00:00:03	FALSE	6

Load Balance

- Evenly distribute the traffic TX to trunk member port



Hash Key

- Known-Unicast packet hash key
  - 2 global sets for L2 packet
    - L2 Hash Key (Selectable): RX Port/SMAC/DMAC/VID
  - 2 global sets for IPv4/IPv6 packet
    - L3 Hash Key (Selectable): RX Port/SMAC/DMAC/VID/SIP/DIP/SPORT/DPORT/IP PROTO/FLOW LABEL
- Other packet type hash key
  - L2 packet (Fixed): RX Port/SMAC/DMAC
  - L3 packet (Fixed): RX Port/SIP/DIP
- SPORT/DPORT keys are ignored for IP fragment packet

Hash Key Shift

- To improve traffic balancing in traffic pattern such as:
  - IP and connected physical port are both incremental, shift RX port hash key to get different hash result

SIP	RX Port	Hash Result (No Shift)			Hash Result (RX Port Shift 1 bit)		
		SIP	RX Port	Final	SIP	RX Port	Final
140.113.0.1	1	3	1	2	3	2	2
140.113.0.2	2	0	2	2	0	4	4
140.113.0.3	3	1	3	2	1	6	7

RX Port

Hash per 6 bit

Shift 2 bit

0	0	0	0	0	0
bit5	bit4	bit3	bit2	bit1	bit0

0	0	0	0	bit5	bit4
bit3	bit2	bit1	bit0	0	0

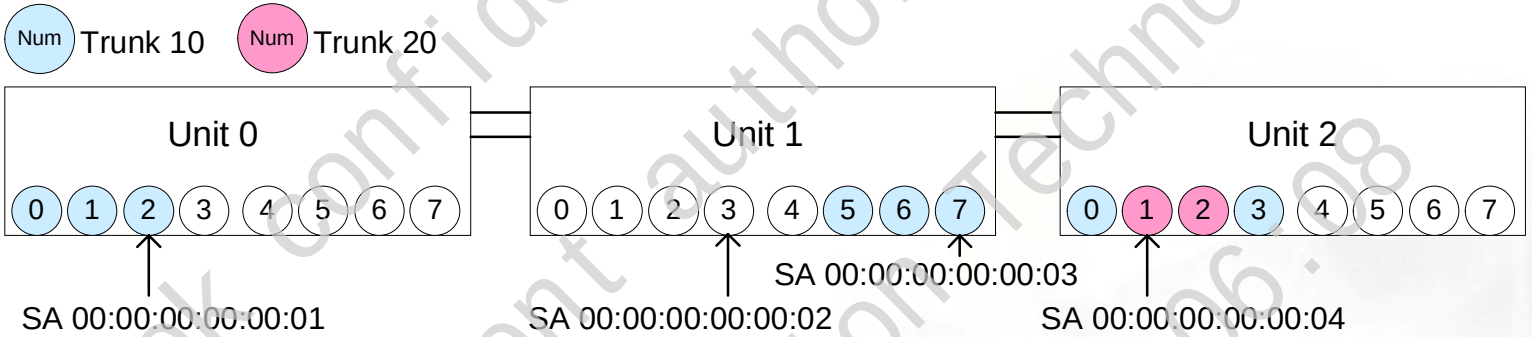
- Per key configured shift
  - Shift bits range from 0 to 5 bits

Traffic Separation

- Supported traffic separation scenario:
  - Separate Flood Traffic to MSB Port
  - Separate Known Multicast Traffic to MSB Port
  - Separate Flood and Known Multicast Traffic to MSB Pot
  - Separate Flood Traffic to MSB and Known Multicast Traffic to second MSB
- Separated ports is dedicated for separated traffic and don't forward other traffic

Stacking Mode: SA Learning

- Trunk ID is learnt in L2 table for packet received from trunk member port
  - Ingress trunk ID table decides RX port mapped trunk ID
  - Packet received from stacking port also lookup the table for learning

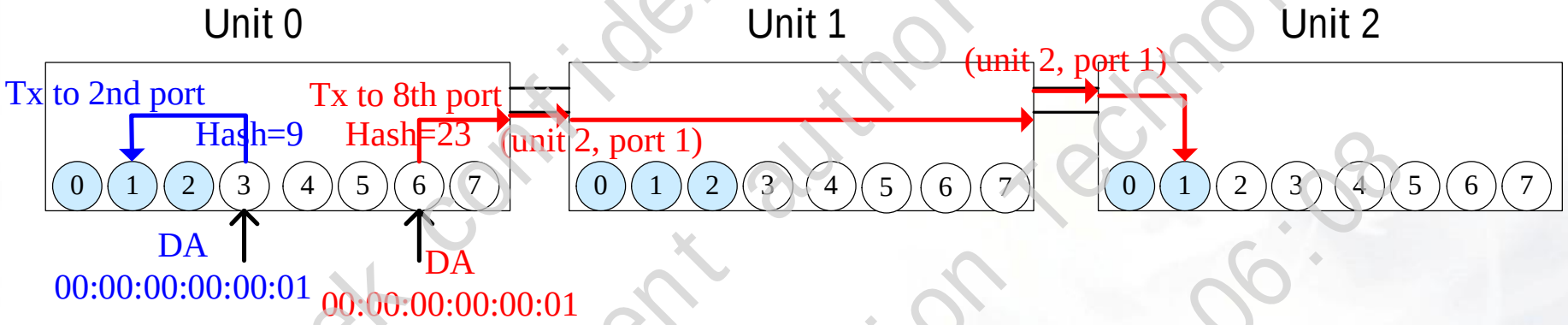


Ingress Trunk ID Table		
(Unit, Port)	TRK_VLD	TRK_VLD
…	…	…
(0, 2)	1	10
(0, 3)	0	-
…	…	…
(1, 3)	0	-
…	…	…
(1, 7)	1	10
…	…	…
(2, 1)	1	20
…	…	…

Learnt L2 Table			
SMAC	is_trunk	Unit ID / Trunk ID	Port ID
00:00:00:00:00:01	TRUE	10	
00:00:00:00:00:02	FALSE	1	3
00:00:00:00:00:03	TRUE	10	
00:00:00:00:00:04	TRUE	20	

Stacking Mode: Unicast Traffic

- First unit calculate hash result and decide remote TX unit/port
- TX unit/port information recorded in stacking header and is followed by further units

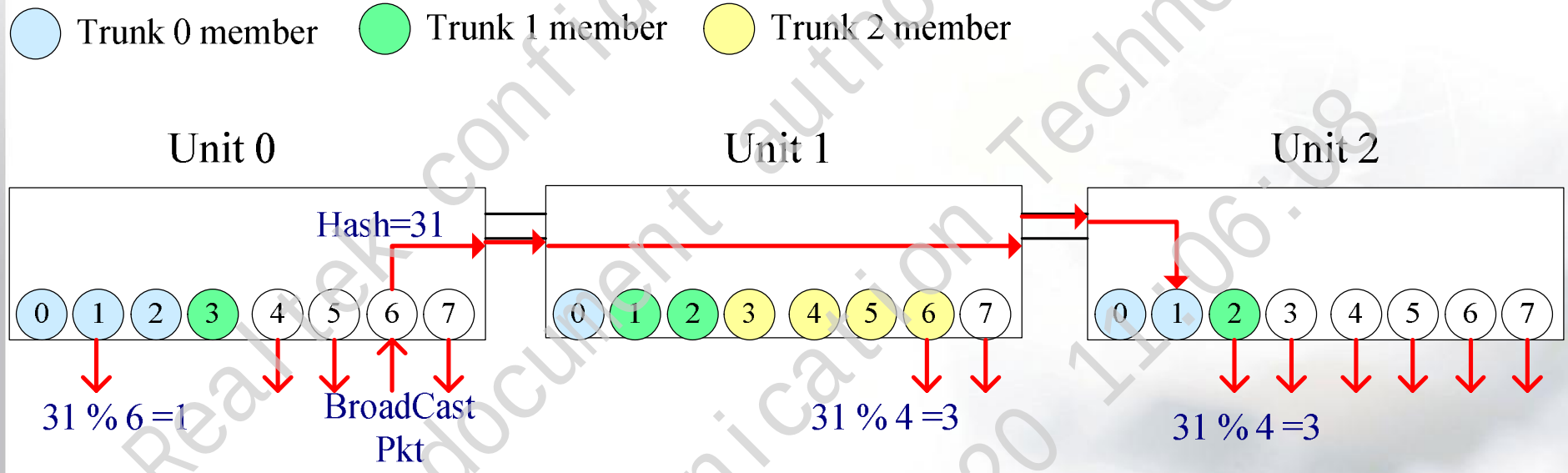


L2 Table			
SMAC	is_trunk	Unit ID	Port ID
		Trunk ID	
00:00:00:00:00:01	TRUE	10	
00:00:00:00:00:02	FALSE	1	3
00:00:00:00:00:03	TRUE	10	

Trunk Egress Table		
IDX	NUM_TX_ Candidate	TX Candidate Ports
10	8	(2, 1), (2, 0), (1, 2), (1, 1), (1, 0), (0, 2), (0, 1), (0, 0)

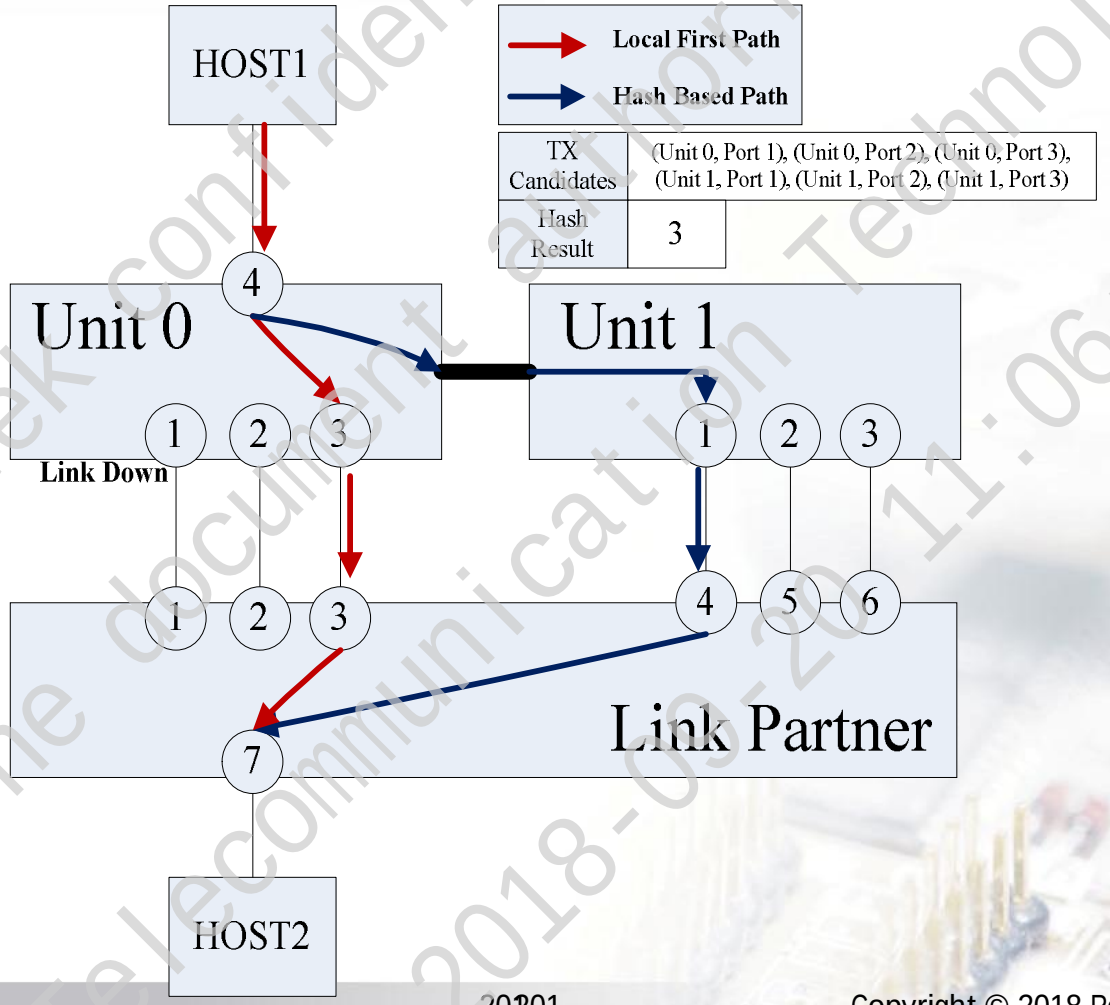
Stacking Mode: Multi-Target Traffic

- Hash result is calculated by the first unit and recorded in stacking header
- Each unit check if local trunk member ports should TX the packet



Link Aggregation: Local First

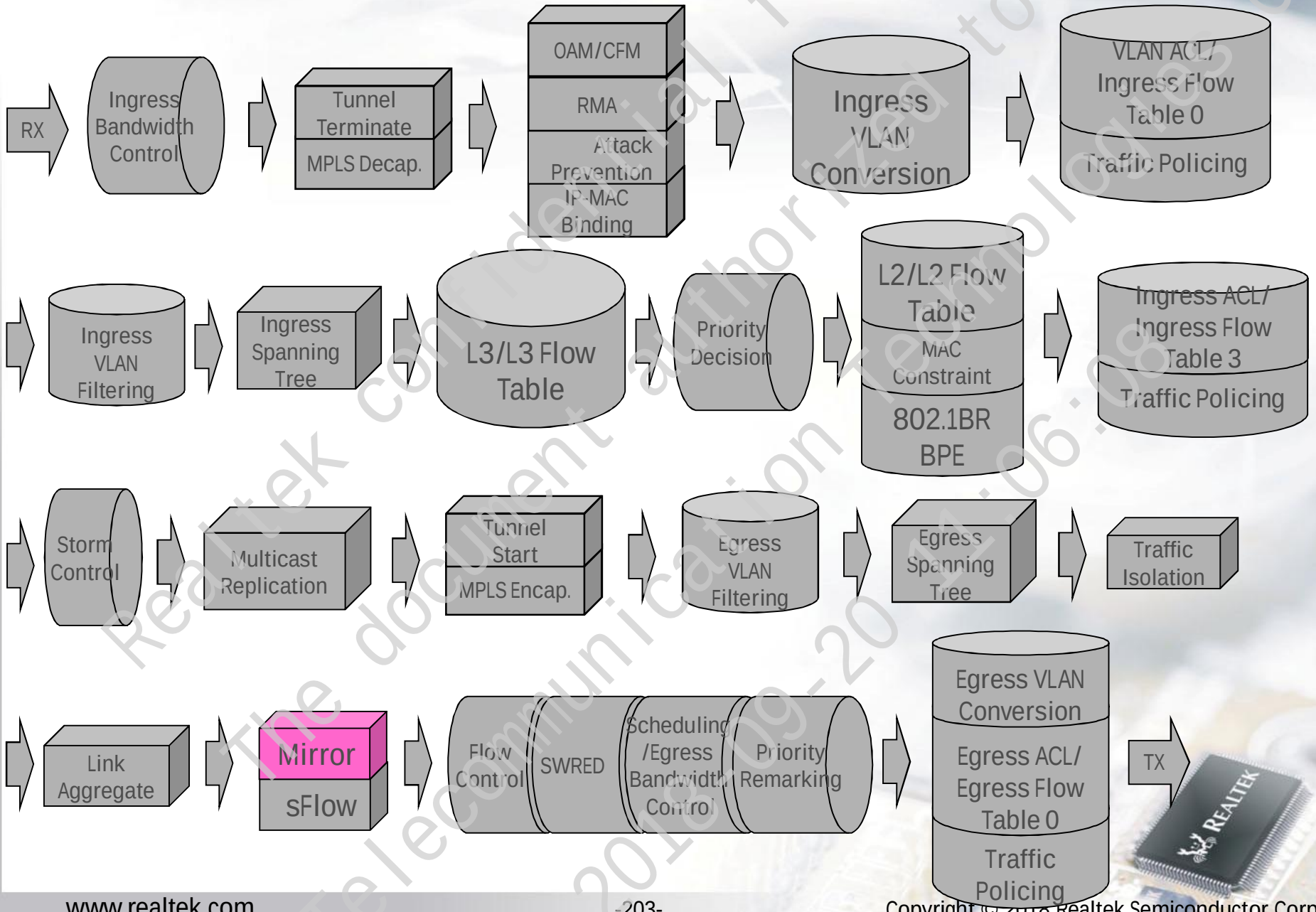
- Reduce stacking port bandwidth
- Option: Avoid Load Sharing to TX Congest
- Option: Avoid Load Sharing to TX Link Down Port



Realtek Semiconductor Corp.

Mirroring

Packet Processing Pipeline

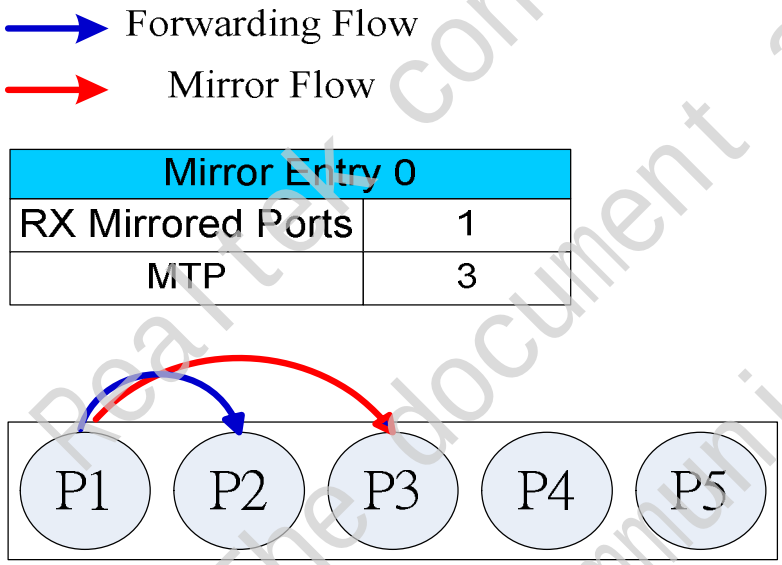


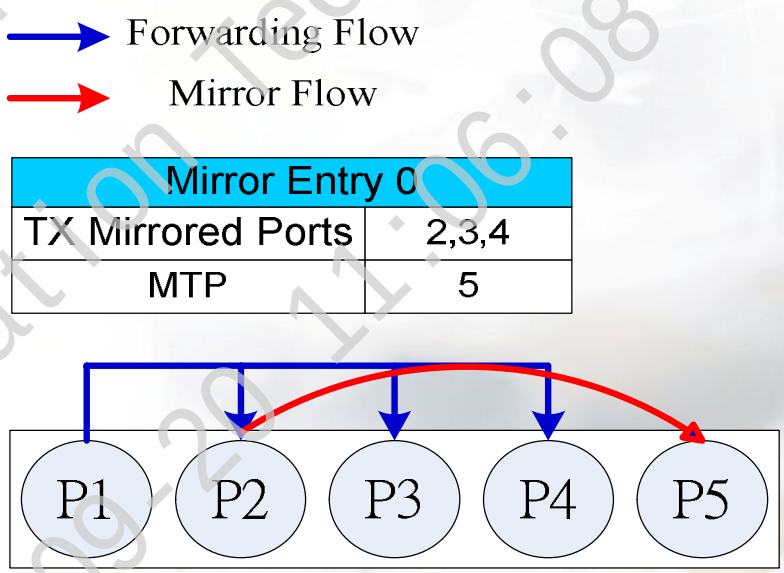
Mirror Overview

- Total 4 Entries Supported, per entry setting
  - Mirror types
    - Port-based
    - Flow-based
    - RSPAN
  - Mirrored port(RX port , TX port) and Mirror to Port (MTP)
  - Mirror Operation
    - AND operation
    - Mirror direction selection
    - MTP isolation
    - Duplication filter
    - Self filter
    - VLAN mode
    - Mirror Queue
- Mirror Traffic Sampling
  - Sample one packet after N packets, per entry setting.

RX And TX Mirror

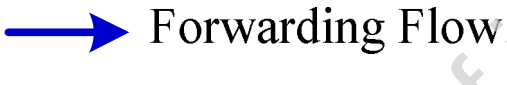
- RX mirror: original packet is mirrored
- TX mirror: modified packet is mirrored
  - Lowest port would be mirrored for a multi-target packet that hits multiple TX mirrored port





Mirror To Multiple MTP

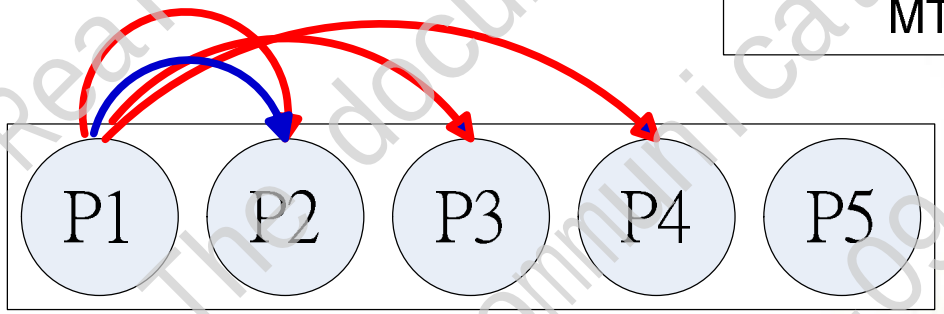
- One packet can be mirrored to multiple MTPs
  - Increase MTP port redundancy



Mirror Entry 0	
RX Mirrored Ports	1
MTP	2

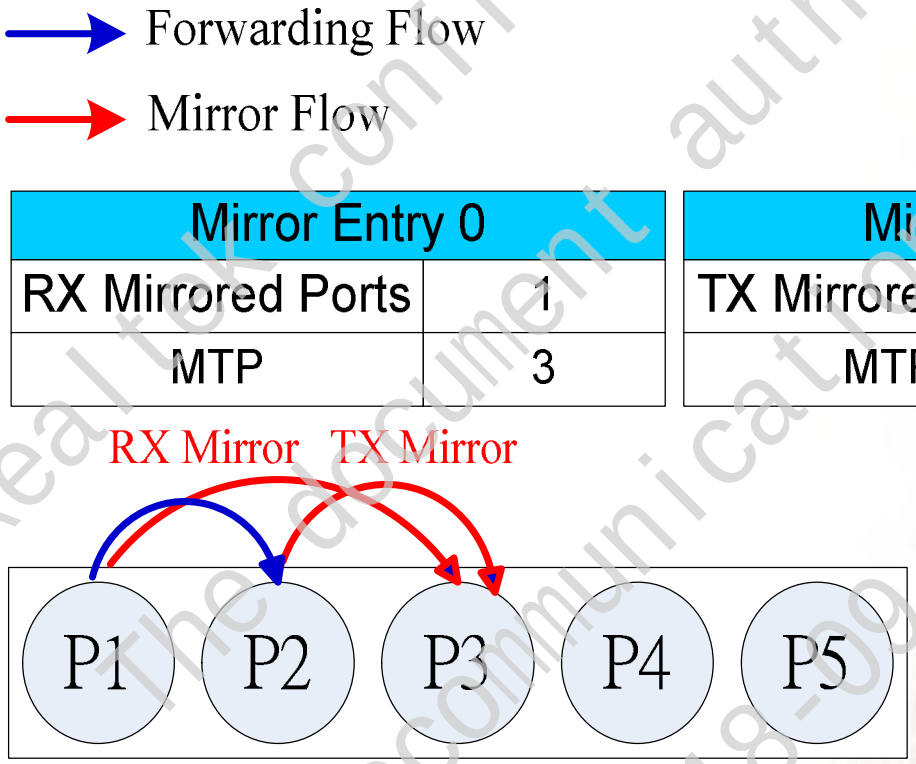
Mirror Entry 1	
RX Mirrored Ports	1
MTP	3

Mirror Entry 2	
RX Mirrored Ports	1
MTP	4



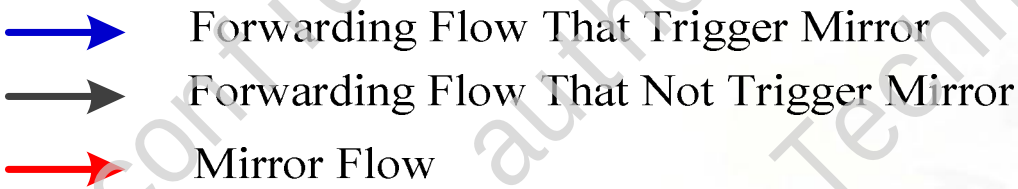
Multiple Mirror Hit To Same MTP

- MTP of mirror entries could be the same port
  - All hit entries would produce mirror entries
  - Both RX and TX path of one packet could be sniffered

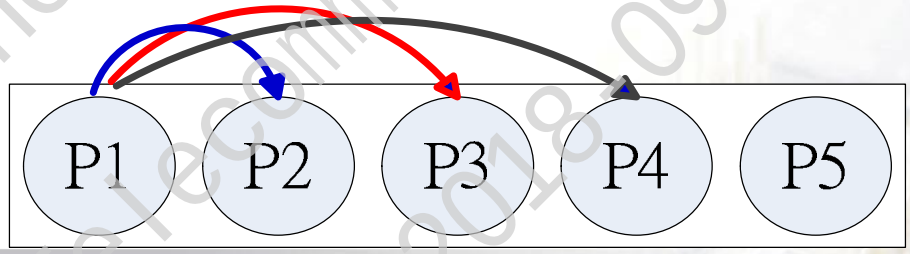


AND Operation and Mirror Direction Selection

- AND operation enabled: Only packets that hit both RX and TX mirrored ports would be mirrored
- Mirror direction select: For packet that hit both RX and TX mirror ports, this option select which direction would be mirrored

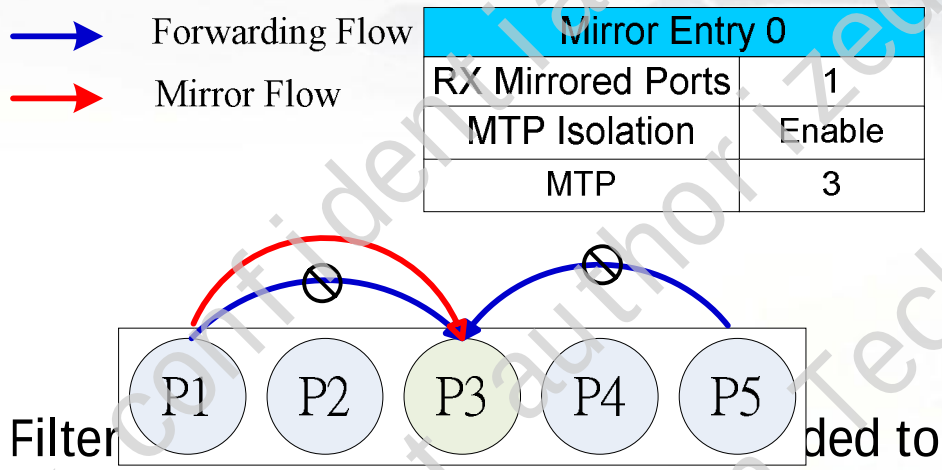


Mirror Entry 0	
RX Mirrored Ports	1
TX Mirrored Ports	2
AND Operation	Enable
Direction Select	RX
MTP	3

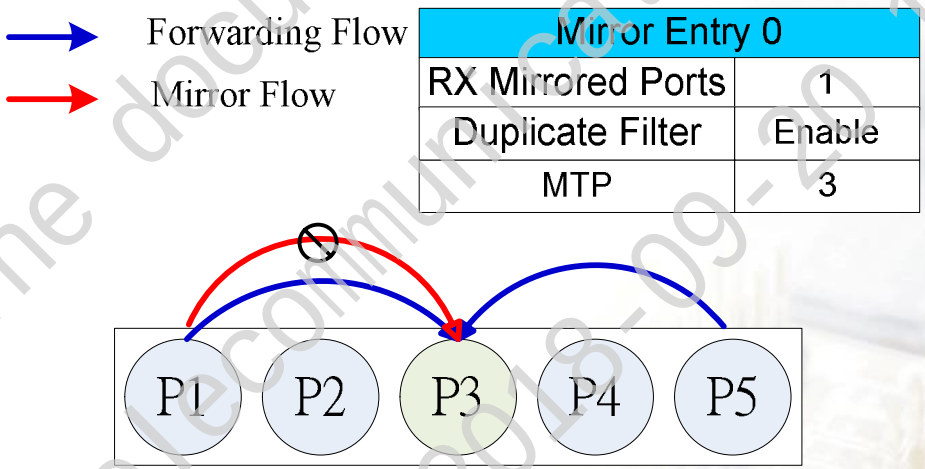


MTP Isolation And Duplicate Filter

- MTP isolation enabled: Only mirrored packet would be sent to MTP



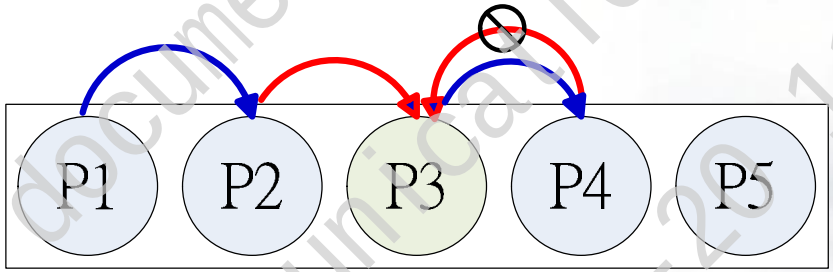
- Duplicate Filter ... ded to MTP would not be mirrored



MTP Self Filter

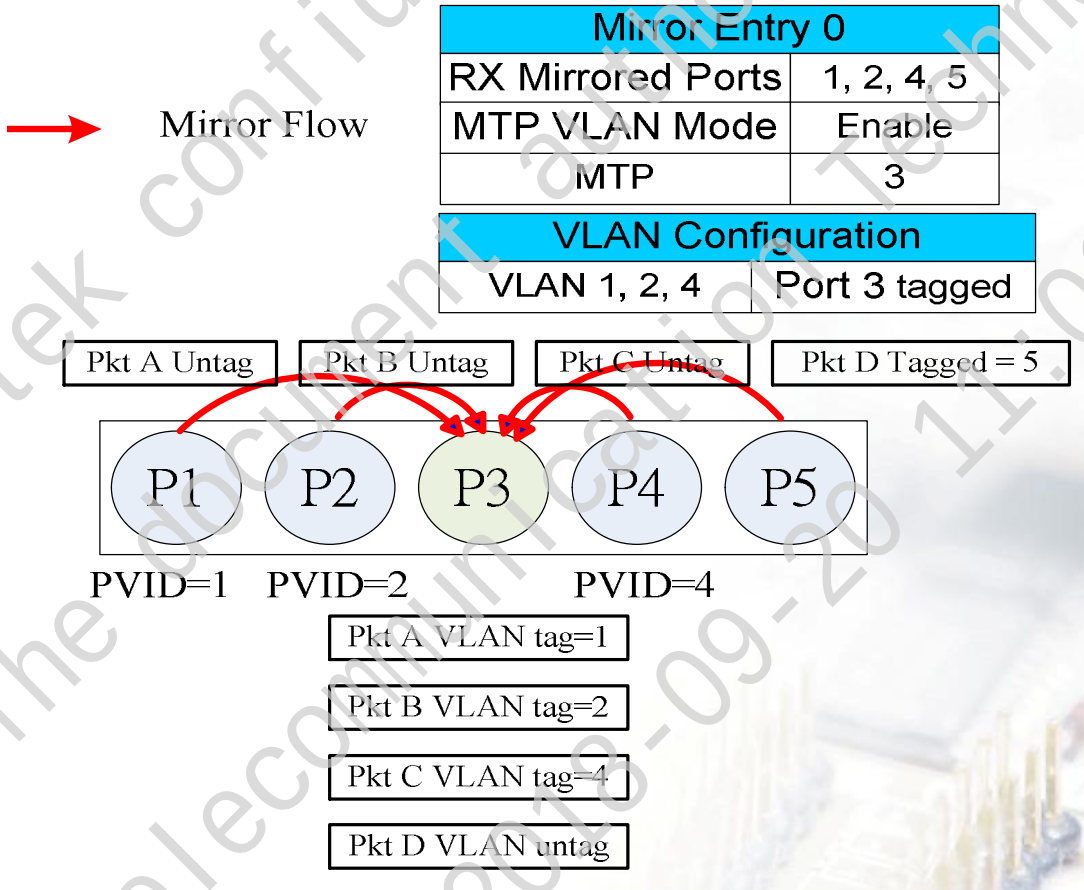
- MTP Self Filter Enabled: Packet received from MTP port would not be mirrored
  - So traffic sent by traffic analyzer could avoid mirroring

Mirror Entry 0	
TX Mirrored Ports	2, 4
Self Filter	Enable
MTP	3



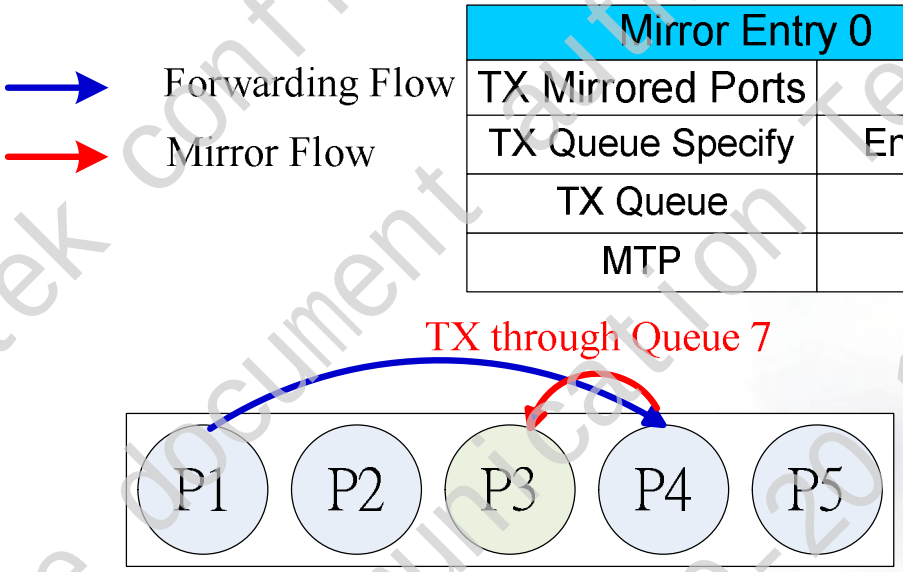
MTP VLAN Mode

- MTP VLAN Mode Enabled: Mirrored packet would follow MTP configuration to update VLAN tag status
  - For special usage such as verify RX mirrored port by VLAN ID



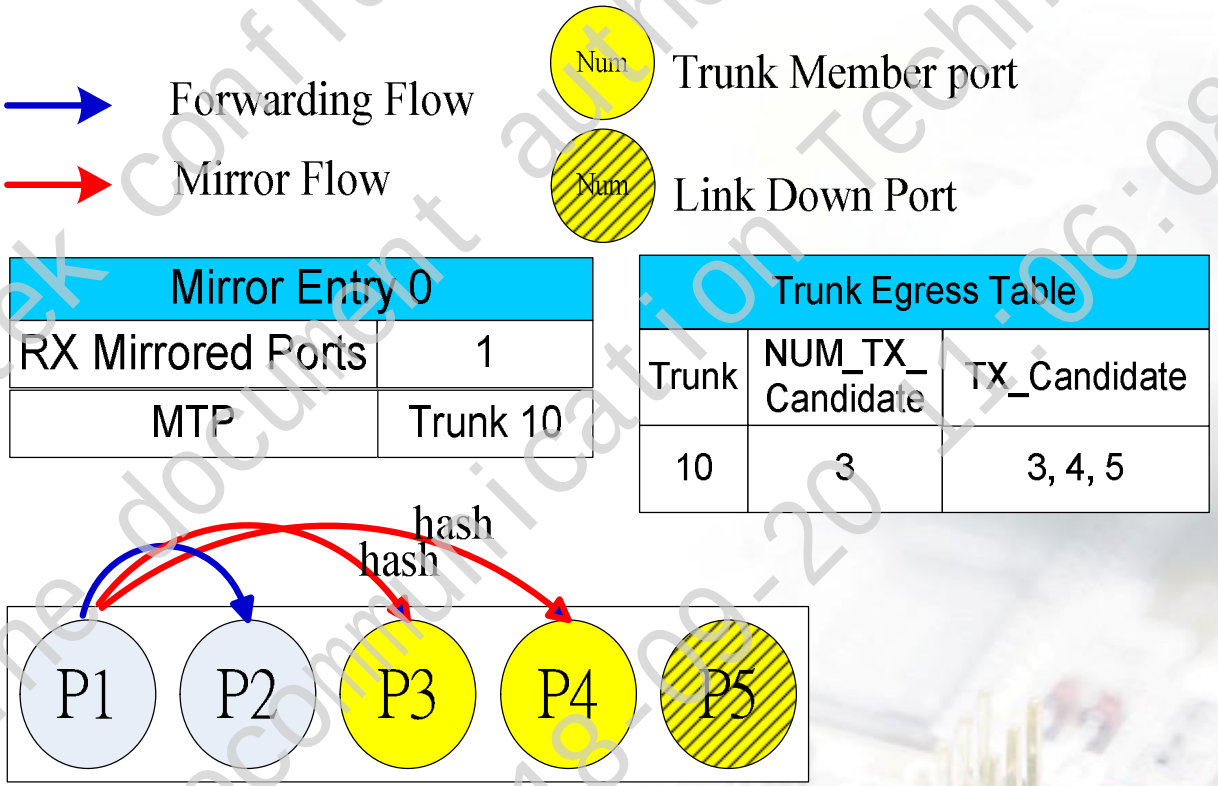
TX QID Configuration

- TX queue of mirrored packet could be specified
  - Give mirrored packet a high or low priority queue depending on analyzer port usage



Trunk Group MTP

- MTP could be a trunk group
  - Traffic is load balanced to all member ports
- Link down auto fail-over is hardware supported



Mirror Entry 0	
RX Mirrored Ports	1
MTP	Trunk 10

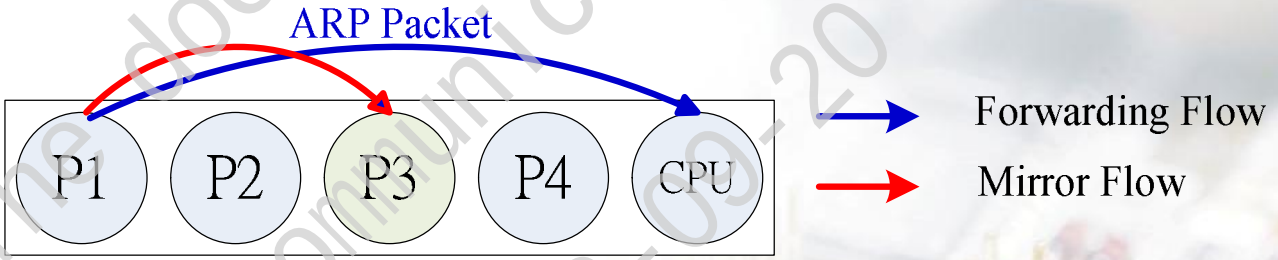
Trunk Egress Table		
Trunk	NUM_TX_Candidate	TX_Candidate
10	3	3, 4, 5

Flow Based Mirror

- Flow based mirror is supported by VLAN/Ingress ACL mirror action
  - Enable ACL mirror action and configure the mirror entry ID and direction

Ingress ACL Entry 0	
ACL Key	Eth=ARP
Action	Mirror
Mirror ID	0
Mirror Direction	RX

Mirror Entry 0	
Mirror Type	Flow Based
MTP	3

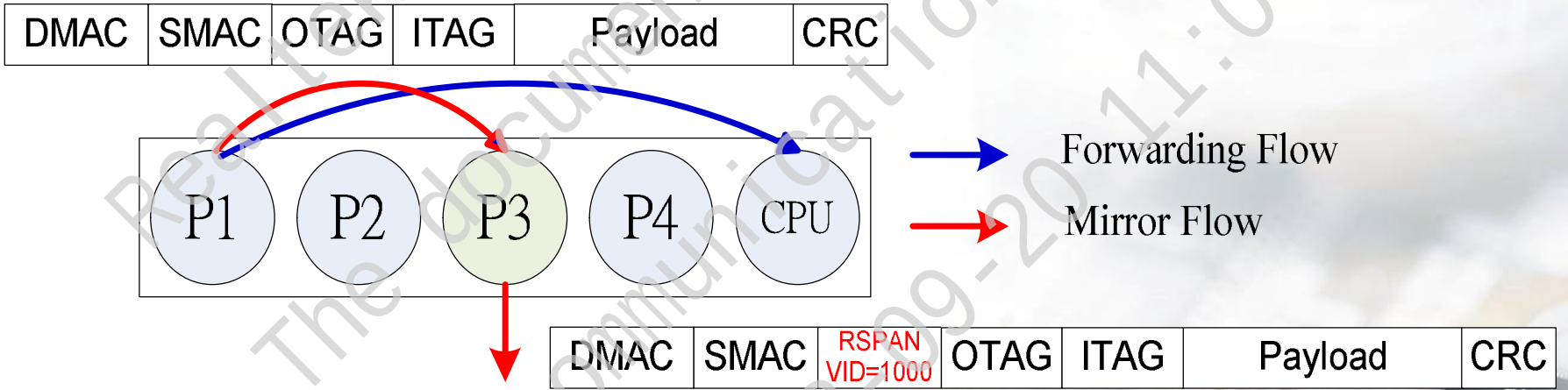


RSPAN Support – Source Switch

- Enable RSPAN tag add function to insert RSPAN tag for mirrored packet

Mirror Entry 0	
Mirror Type	Flow/ Port Based
RX Mirrored Ports	1
RSPAN ADD	Enabled
MTP	3

RSPAN Tag Configuration 0	
OTPID Idx	0
CFI	0
PRI	0
VID	1000

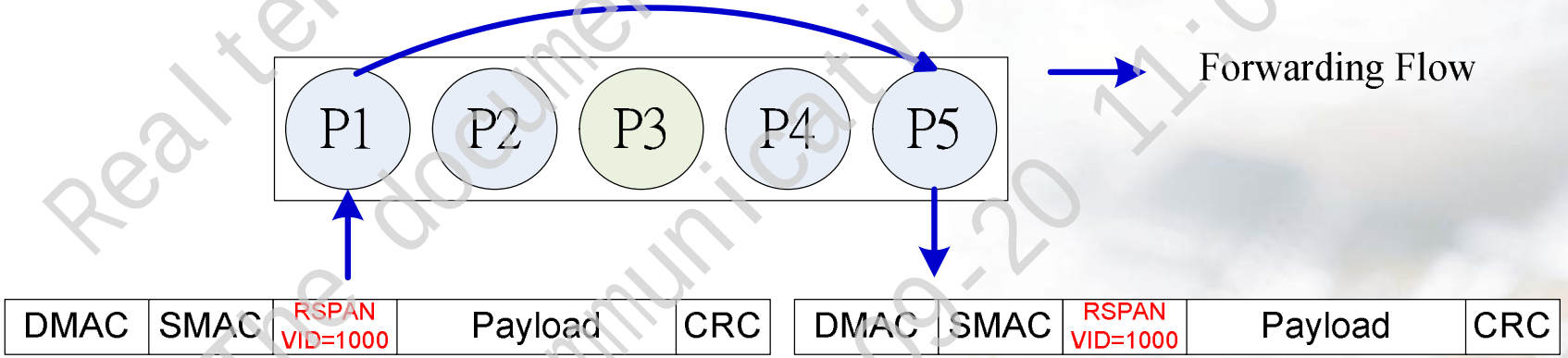


RSPAN Support – Intermediate Switch

- Configure VLAN member table to forward the RSPAN packet
- Configure VLAN profile to disable L2 learning

VLAN 1000 Configuration	
Member Ports	1, 5
Tagged Ports	1, 5
VLAN Profile ID	1

VLAN Profile 1	
L2 Learn	Disabled



RSPAN Support – Destination Switch

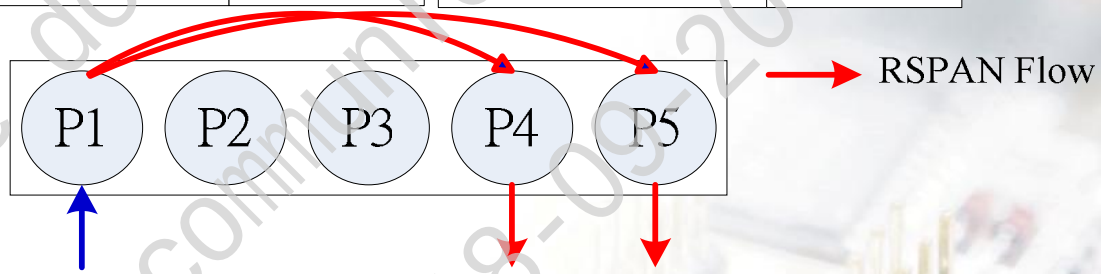
- Set mirror entry type as RSPAN to enable RSPAN mirror
  - RX mirrored ports decide RSPAN detecting ports
  - RSPAN VID should be matched
  - Enable RSPAN tag remove function
  - L2 learning is auto disabled
- RSPAN multiple hit is supported

Mirror Entry 0	
Mirror Type	RSPAN
RX Mirrored Ports	1
RSPAN Remove	Enable
MTP	4

Mirror Entry 1	
Mirror Type	RSPAN
RX Mirrored Ports	1
RSPAN Remove	Enable
MTP	5

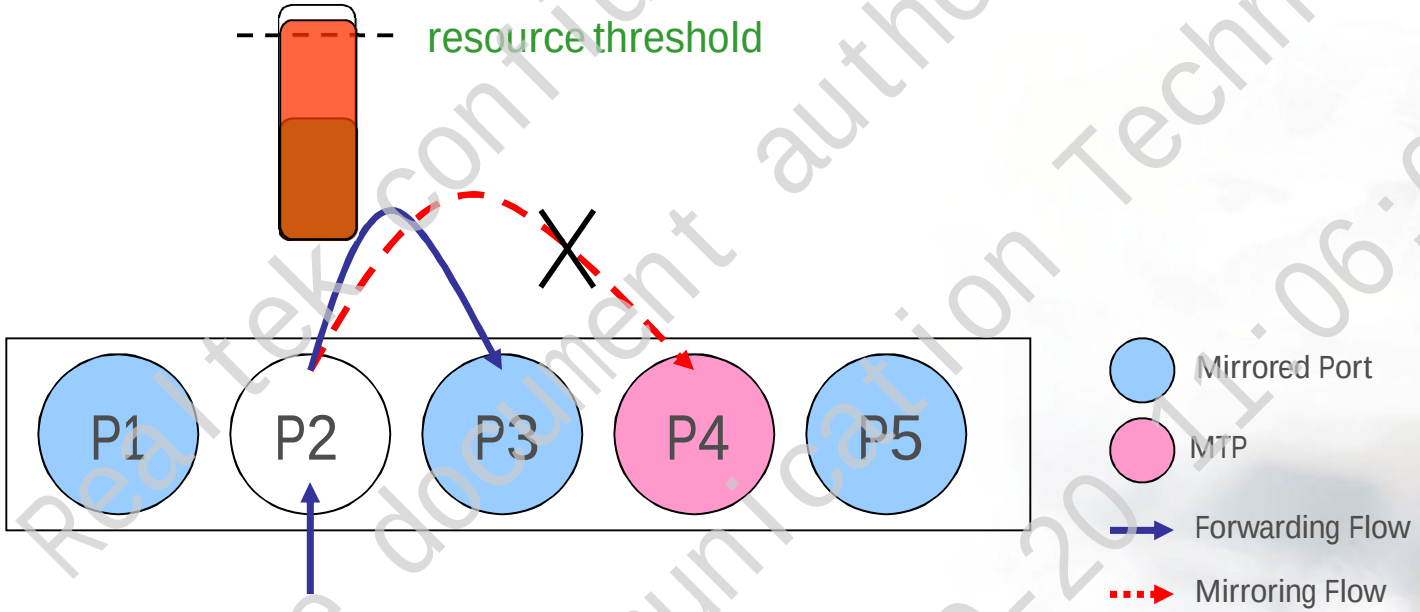
RSPAN Tag Configuration 0	
VID	1000

RSPAN Tag Configuration 1	
VID	1000



Mirroring Congestion

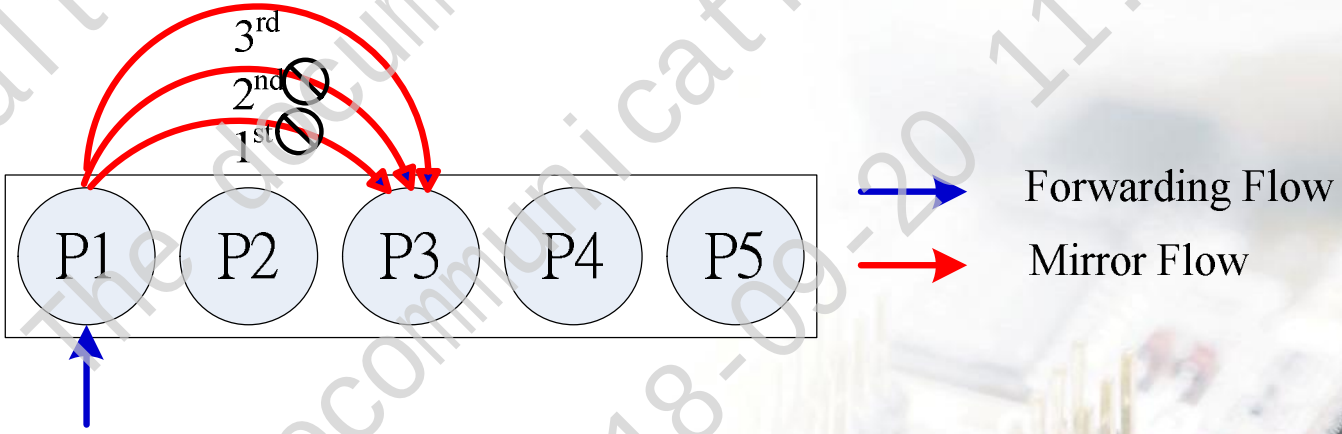
- Mirrored traffic would not trigger flow control
  - The mirroring would auto stop if port is congested
  - Ex: MTP links in 100M, mirrored port links in 1000M



Mirroring – Sample Rate

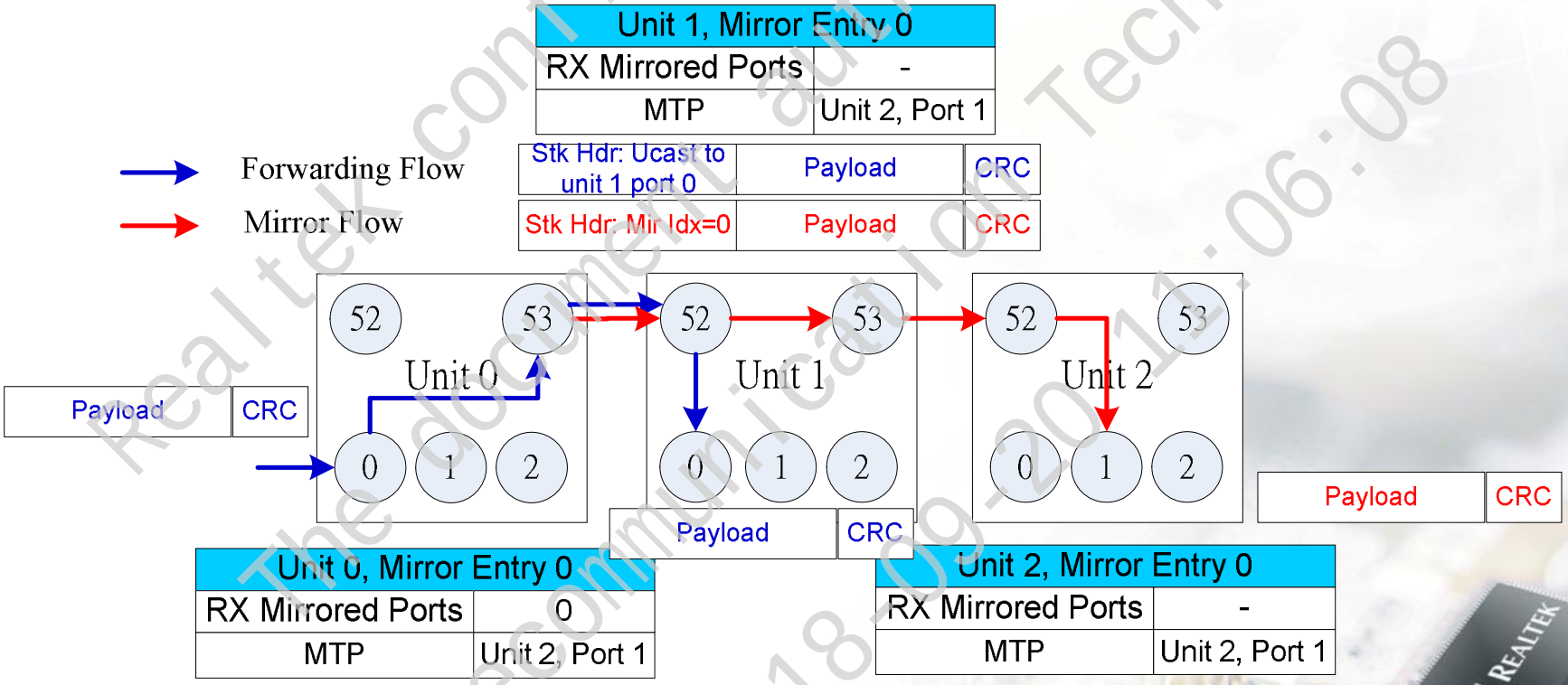
- Each mirroring set can configure sampling rate (unit: packet)
  - Ex: Configure sampling rate to 2, for every 3 packets, system would skip 2 packets and mirror the 3rd packet
  - Application example: IPFIX

Mirror Entry 0	
Mirror Type	Any
RX Mirrored Ports	1
Sample Rate	2
MTP	3



Mirroring In Stacking System

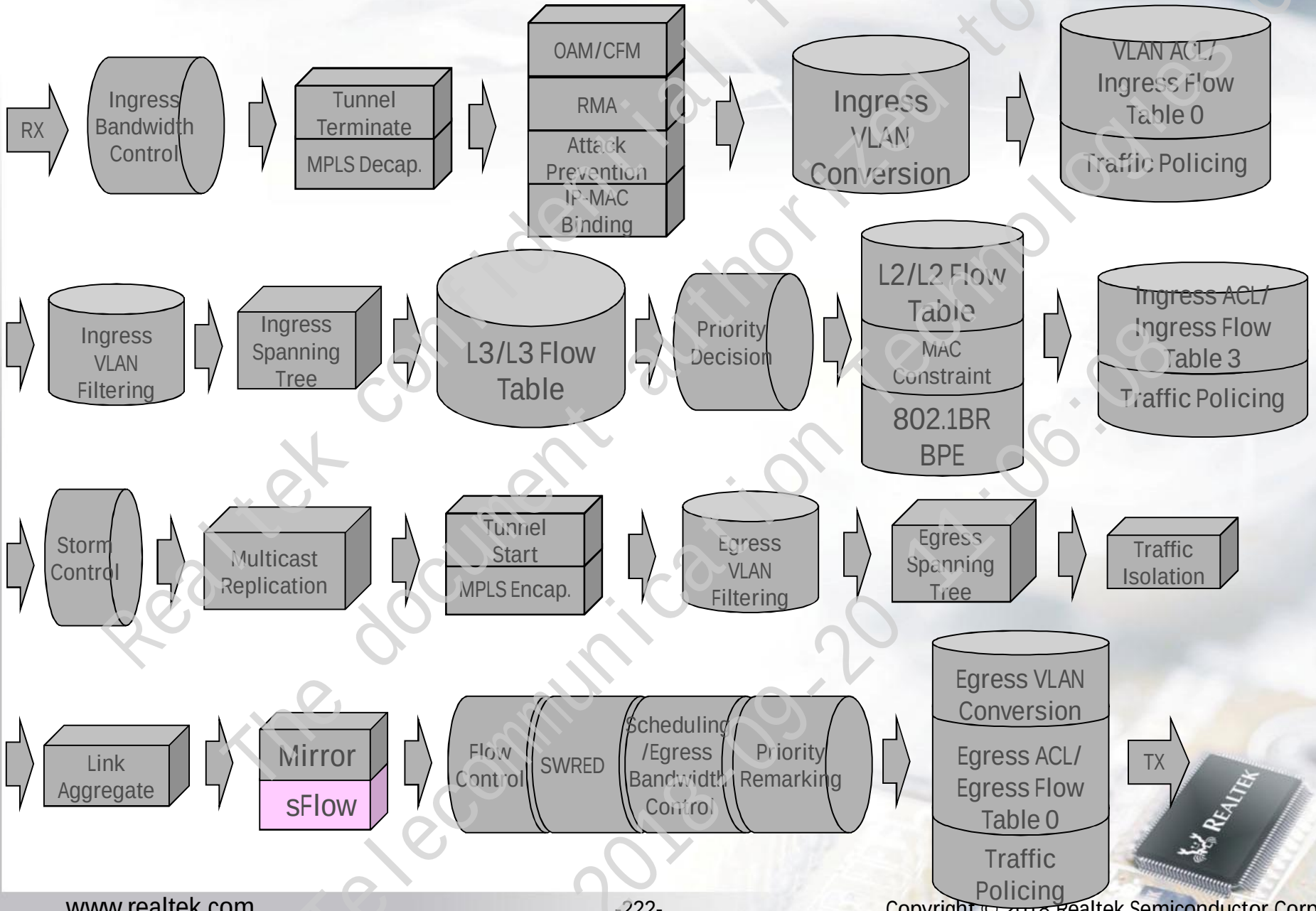
- Specify unit ID of MTP port
- Mirror packet is generated by switch that mirror is triggered
- Mirror index is attached in stacking header
  - Received switch use mirror index to look up MTP to decide TX port



Realtek Semiconductor Corp.

sFlow

Packet Processing Pipeline



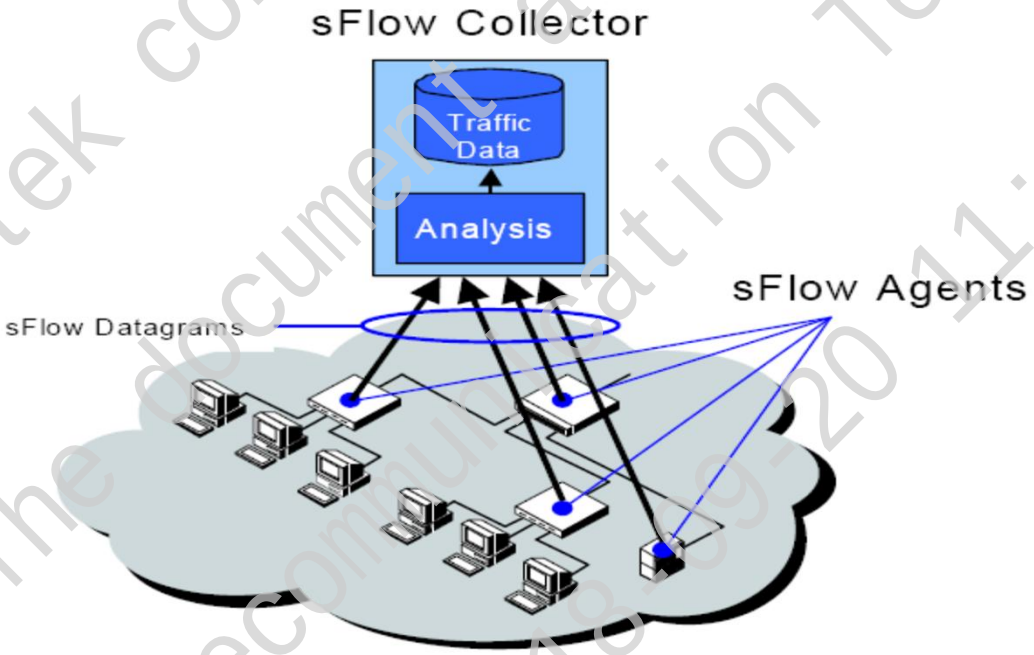
sFlow Overview

- Per-Port configuration
  - Sample rate
  - Ingress sampling
  - Egress sampling

- Flow-based sampling
  - ACL + Mirror entry
  - Per Mirror entry define sample rate

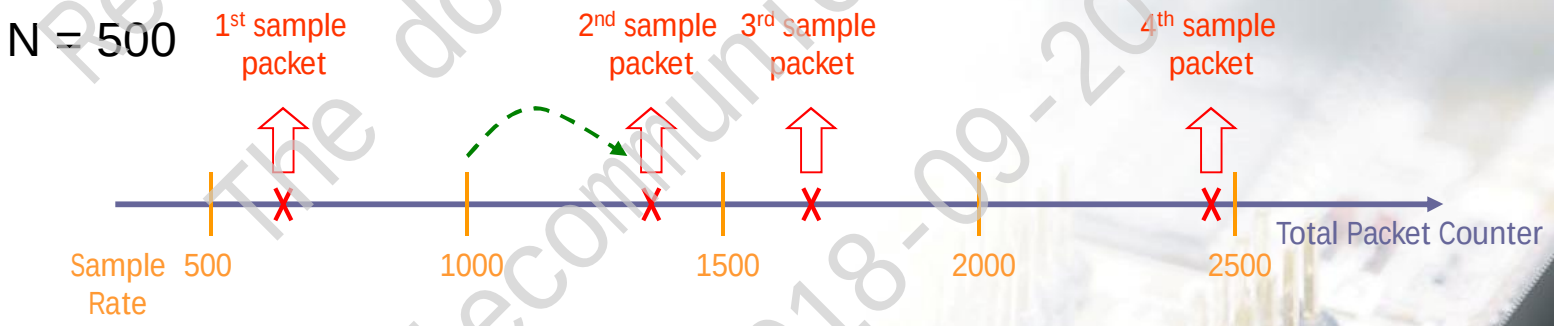
sFlow

- sFlow is a sampling technology embedded in switches and routers for monitoring network. These devices are sFlow agent.
- sFlow agent pack sampled packet as sFlow datagram, then send it to sFlow collector for analyzing.
- Base on sample rate (N), an average of 1 out of N packets is randomly sampled.



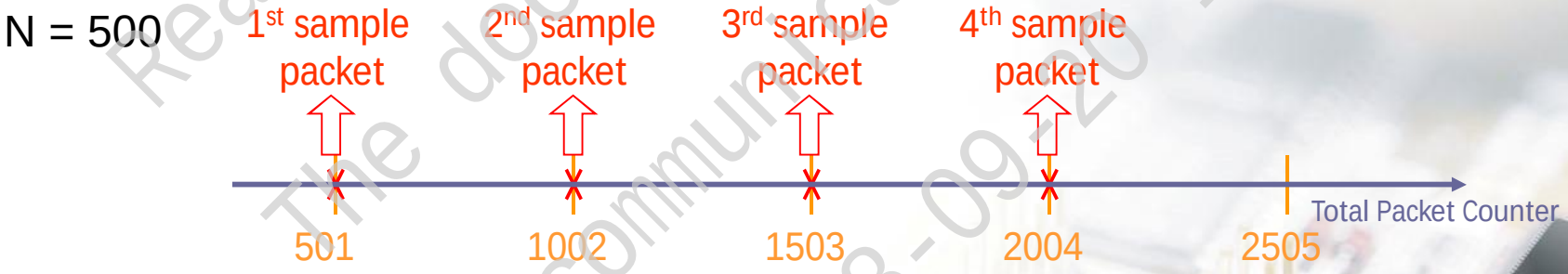
sFlow – counter sample

- Sampled packet by counter sample is trapped to CPU
- Per port configuration
  - Ingress sample rate (0 ~ 65535)
  - Egress sample rate (0 ~ 65535)
- Global configuration
  - Sample selector
    - Prefer ingress or egress when sampled the same packet
- Sample Rate
  - Rate is N means sample 1 of the N packets randomly in the monitored flow
  - Rate is 0 => disable sampling function



sFlow – flow sample

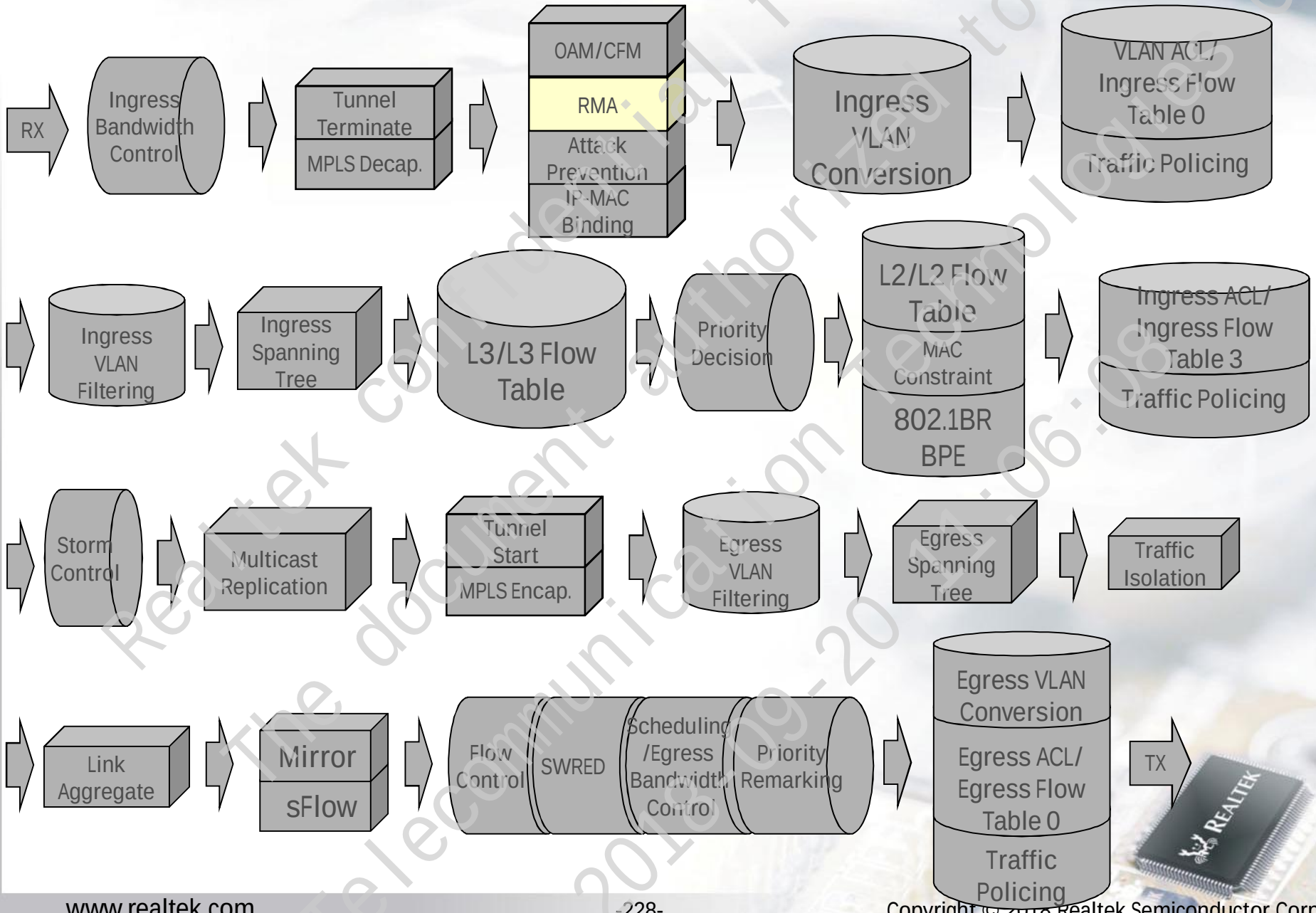
- Support flow-based sFlow by ACL and mirror
- Qualify the specific flow by ACL, then bind with mirror action
- Configure mirror entry's sample rate and Rx/Tx sampled ports
- When sample rate is N, mirror entry every skip N packets then sample one packet
- Sampled packet by mirror entry is send to destination port



Realtek Semiconductor Corp.

RMA

Packet Processing Pipeline



L2 RMA (Reserved Multicast Address) Action

High

Pri

Low

ID	Address	Configure	Action			
			Forward	Drop	Trap	Flood
1	XX-XX-XX-XX-XX-XX (User defined 4 sets)	Global	Yes	Yes	Yes	Yes
2	Ether Type = 0x88CC (LLDP)	Per Port	Yes	Yes	Yes	Yes
3	01-80-C2-00-00-00 (BPDU)	Per Port	Yes	Yes	Yes	Yes
4	01-80-C2-00-00-02 (Slow Protocol, excluding OAM)	Global	Yes	Yes	Yes	--
5	Ether Type = 0x88F7 (PTP)	Per Port	Yes	Yes	Yes	--
6	Ether Type = 0x888E(EAPOL)	Per Port	Yes	Yes	Yes	Yes
7	01-80-C2-00-00-01,01-80-C2-00-00-03~01-80-C2-00-00-1F	Global	Yes	Yes	Yes	--
8	01-80-C2-00-00-20 ~ 01-80-C2-00-00-2F	Global	Yes	Yes	Yes	--
9	01-80-C2-00-00-30 ~ 01-80-C2-00-00-3F	Global	Yes	Yes	Yes	--

- Lower ID with higher priority for multiple hit
- L2-table lookup will be performed when action is Forward/Flood

- L2-table lookup miss flooding portmask
  - Forward: RMA_FLD_PMSK
  - Flood (bypass egress VLAN filter)
    RMA_USR_DEF_FLD_PMSK
    RMA_LLDP_FLD_PMSK
    RMA_BPDU_FLD_PMSK
    RMA_EAPOL_FLD_PMSK

L2 RMA Bypass STP/VLAN

- When action is Trap, the following are the RMA traffic which can bypass STP/VLAN filter
- Each user defined RMA can be configured as bypass STP/VLAN filter

ID	Address	Bypass	
		STP *	VLAN
1	XX-XX-XX-XX-XX-XX (User defined 4 sets)	Configurable	Configurable
2	Ether Type = 0x88CC (LLDP)	Yes	Yes
3	01-80-C2-00-00-00 (BPDU)	--	Yes
4	01-80-C2-00-00-02 (Slow Protocol, excluding OAM)	Yes	Yes
5	Ether Type = 0x88F7 (PTP)	Yes	--
6	Ether Type = 0x888E(EAPOL)	Yes	Yes
7	01-80-C2-00-00-01,01-80-C2-00-00-03~01-80-C2-00-00-1F	--	--
8	01-80-C2-00-00-20 ~ 01-80-C2-00-00-2F	Yes	Yes
9	01-80-C2-00-00-30 ~ 01-80-C2-00-00-3F	--	--

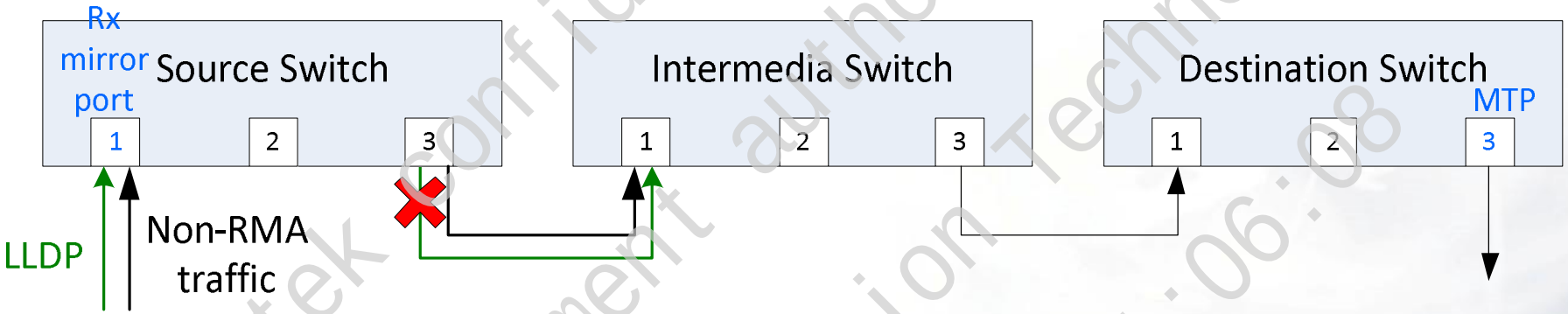
* STP "Disable" state cannot be bypassed

L2 RMA Learning

- When action is Drop, address will NOT be learnt
- When action is not Drop, learning will be performed if below cases all pass:
  - STP state is in "Learning" or "Forwarding"
  - No violation to ingress VLAN filter
  - RMA SA learning is configured to learn (default)
    - Each RMA (01-80-C2-00-00-00~01-80-C2-00-00-2F) has SA leaning ability
    - For specific application has SA leaning ability
      - PTP, LLDP, EAPOL

L2 RMA Mirror

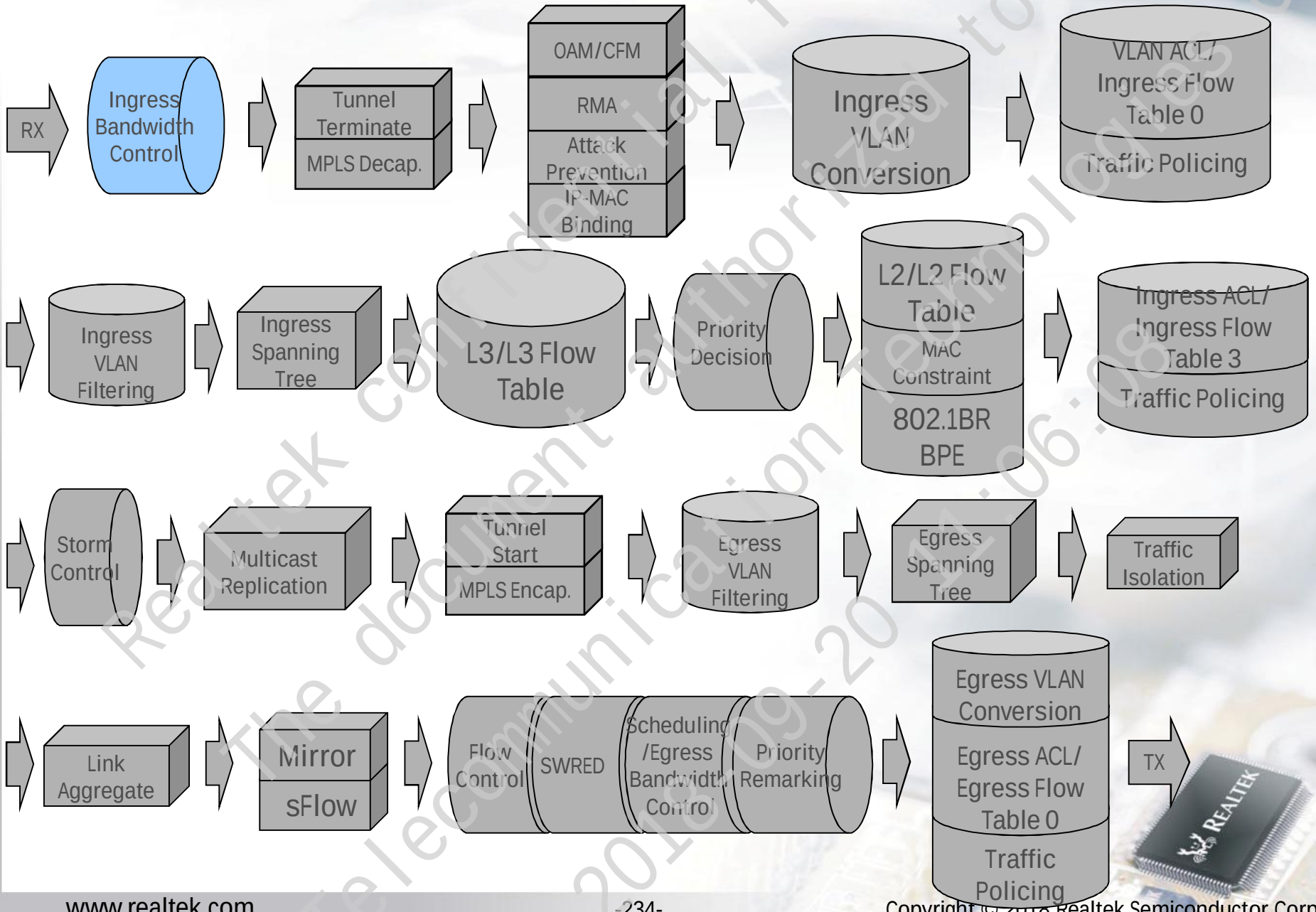
- System can be configured as deny mirror for all RMA traffic.
- Example: Do NOT mirror LLDPDU to remote destination at RSPAN.



Realtek Semiconductor Corp.

Ingress Bandwidth Control

Packet Processing Pipeline

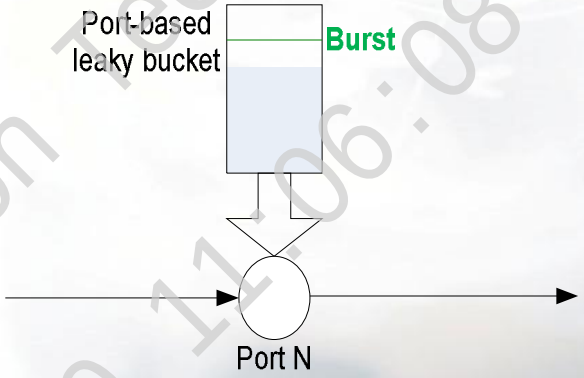


Ingress Rate Limit Configuration (1/2)

- Global configuration
  - Include/Exclude IFG
  - Bypass control packet configuration
    - ARP/NS(Neighbor Solicitation)
    - RMA
    - BPDU
    - IGMP
    - RTK control packet (Ether type 0x8899)
    - DHCP
    - RIP & OSPF
  - Include/Exclude bypassed control packet for bandwidth consumption

Ingress Rate Limit Configuration (2/2)

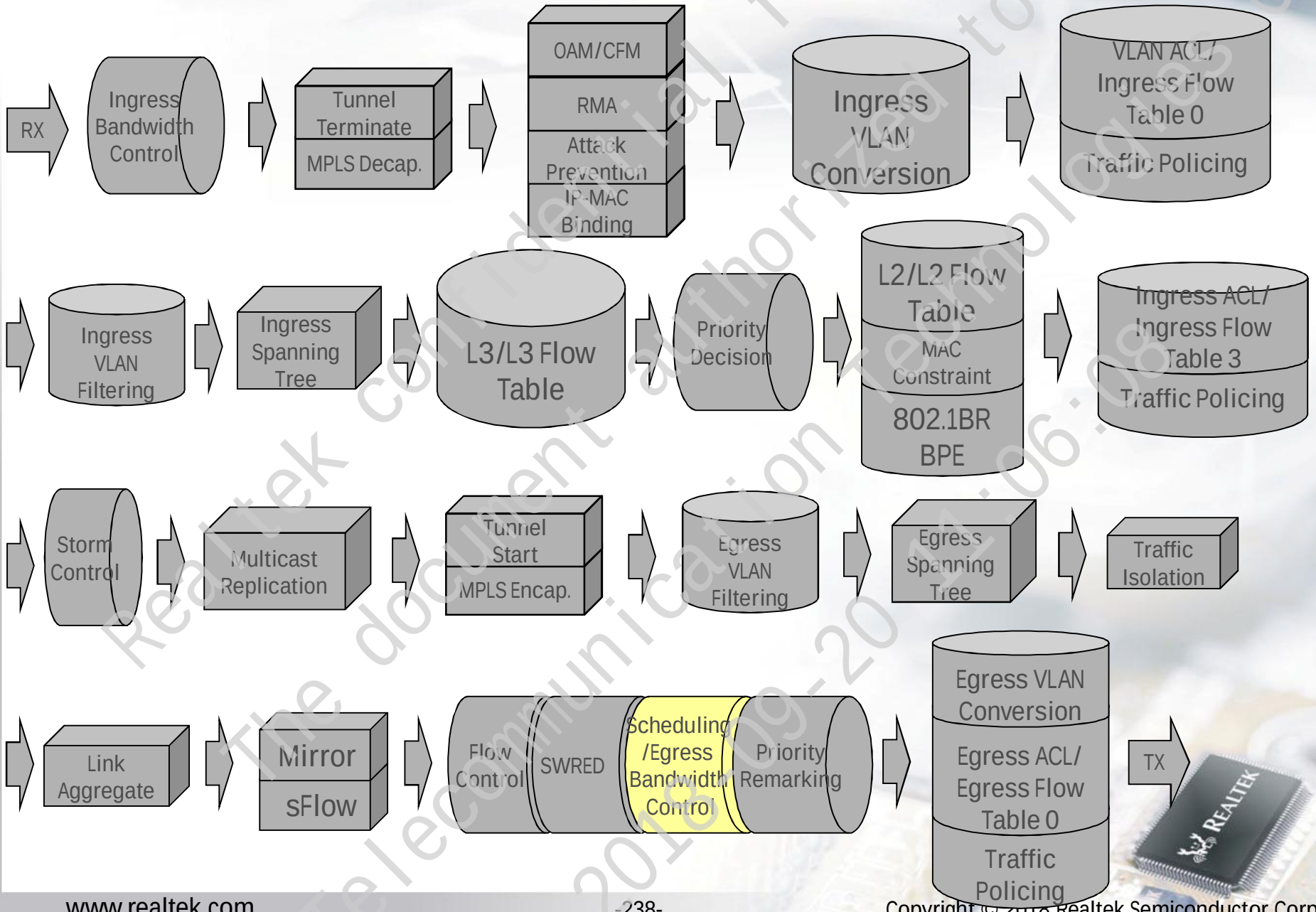
- Port-based Configuration
  - Rate Configuration
    - Enable/Disable
    - Rate (16 kbps granularity up to 10G, value 0 for blocking)
    - Burst size (MAX to 64KB)
  - Flow control configuration
    - Enable/Disable
  - Indication flag for exceed state



Realtek Semiconductor Corp.

Egress Bandwidth Control

Packet Processing Pipeline



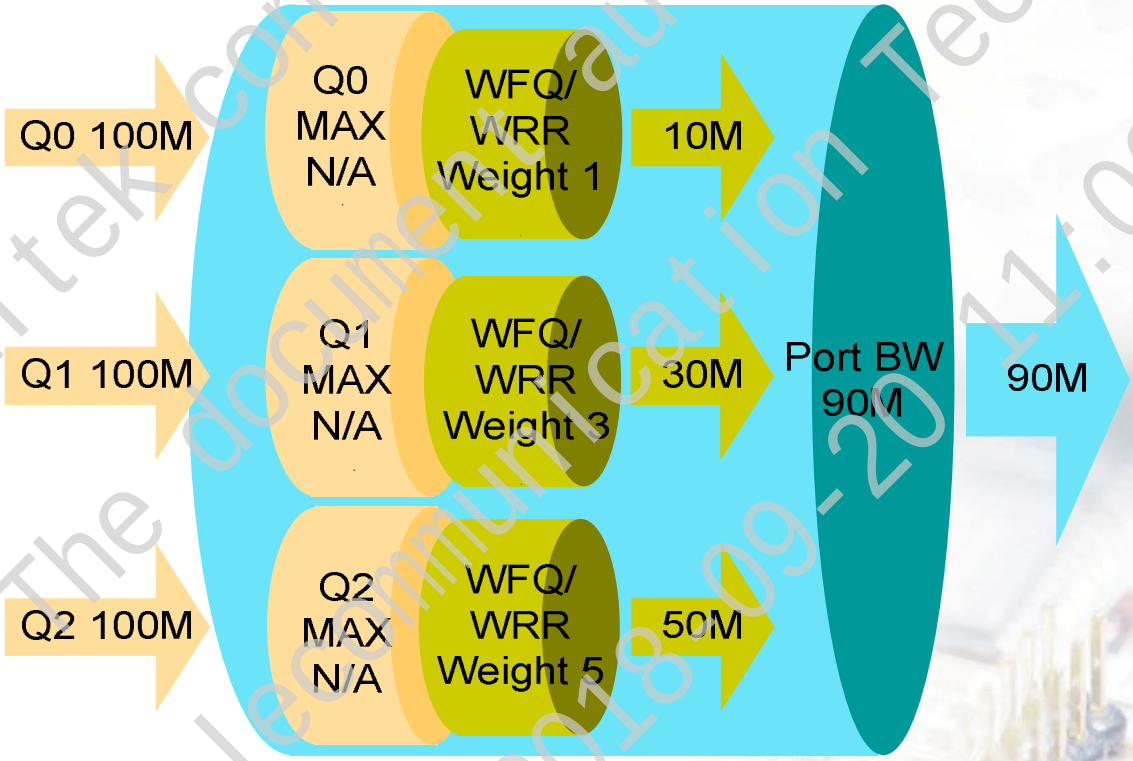
Egress Bandwidth Control Overview

- Port-based Configuration
  - Normal Port: Byte count (unit: 16kbps)
  - CPU Port: Byte count (unit: 16kbps) or Packet count (unit: 1pps)
- Queue-based Configuration
  - Normal Port: Byte count (unit: 16kbps)
    - Maximum bandwidth
    - Assured bandwidth
  - CPU Port: Byte count (unit: 16kbps) or Packet count (unit: 1pps)
    - Maximum bandwidth
- Bandwidth Allocation Priority
  - Assured > Strict > WFQ/WRR
- Assured Bandwidth Mode
  - Share mode: Remain bandwidth can be borrowed
  - Fixed mode: Remain bandwidth is wasted

Egress Bandwidth with WFQ/WRR

- Example: WFQ/WRR(same packet length)

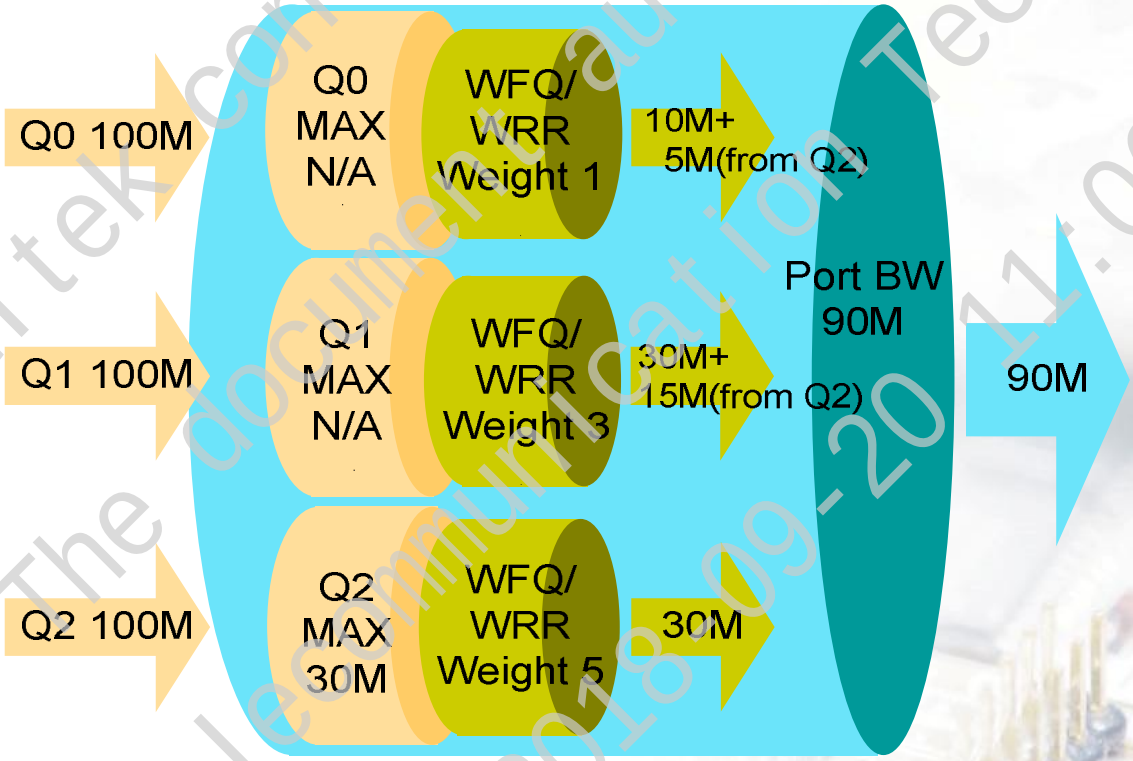
Configuration						
Port BW	Queue ID	Weight	Strict	Assured BW	Assured Mode	Maximum BW
90M	0	1	N/A	N/A	N/A	N/A
	1	3	N/A	N/A	N/A	N/A
	2	5	N/A	N/A	N/A	N/A



Queue Maximum Bandwidth

- Example: WFQ/WRR(same packet length)

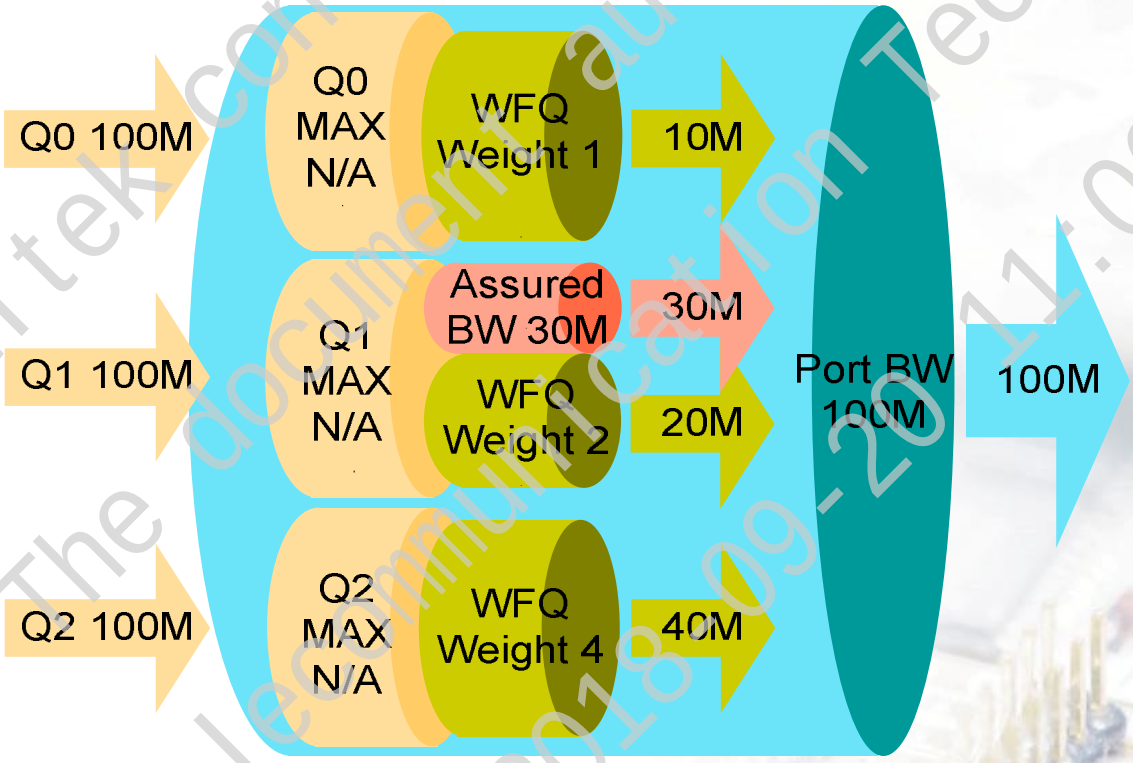
Configuration						
Port BW	Queue ID	Weight	Strict	Assured BW	Assured Mode	Maximum BW
90M	0	1	N/A	N/A	N/A	N/A
	1	3	N/A	N/A	N/A	N/A
	2	5	N/A	N/A	N/A	30M



Assured Bandwidth (1/3)

- Assured Bandwidth in share mode
  - Remain bandwidth can be borrowed

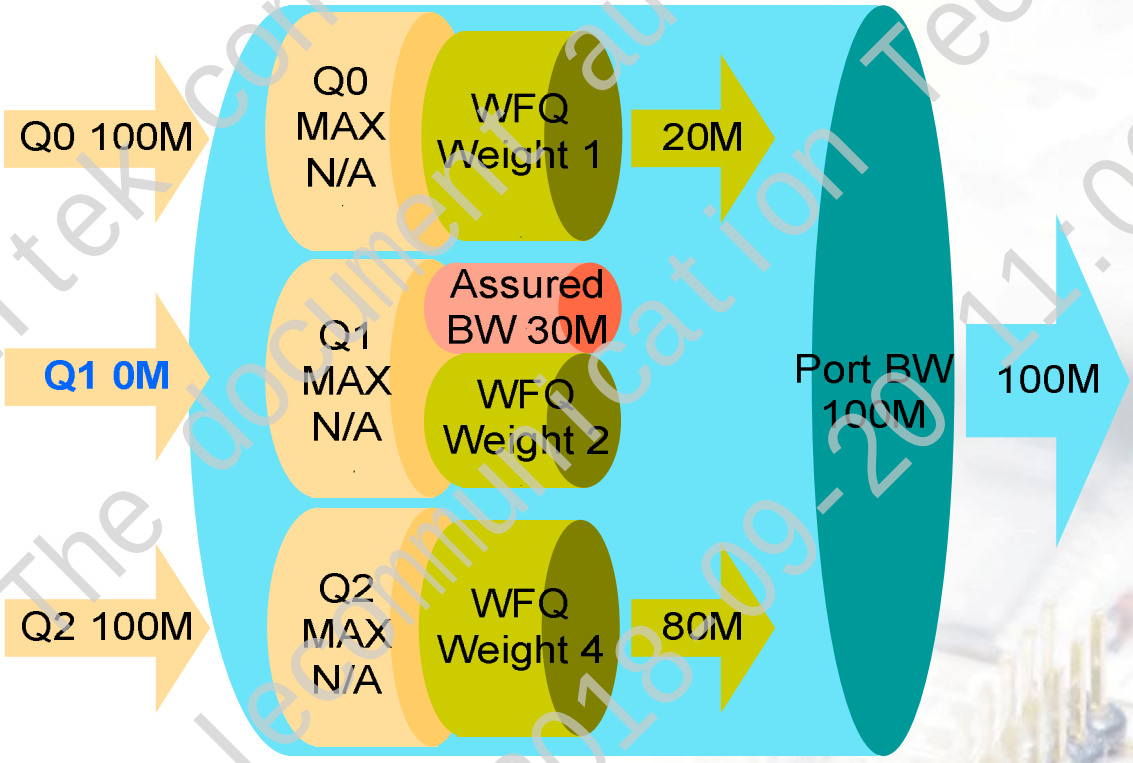
Configuration						
Port BW	Queue ID	WFQ Weight	Strict	Assured BW	Assured Mode	Maximum BW
100M	0	1	N/A	N/A	N/A	N/A
	1	2	N/A	30M	Share	N/A
	2	4	N/A	N/A	N/A	N/A



Assured Bandwidth (2/3)

- Assured Bandwidth in share mode
  - Remain bandwidth can be borrowed

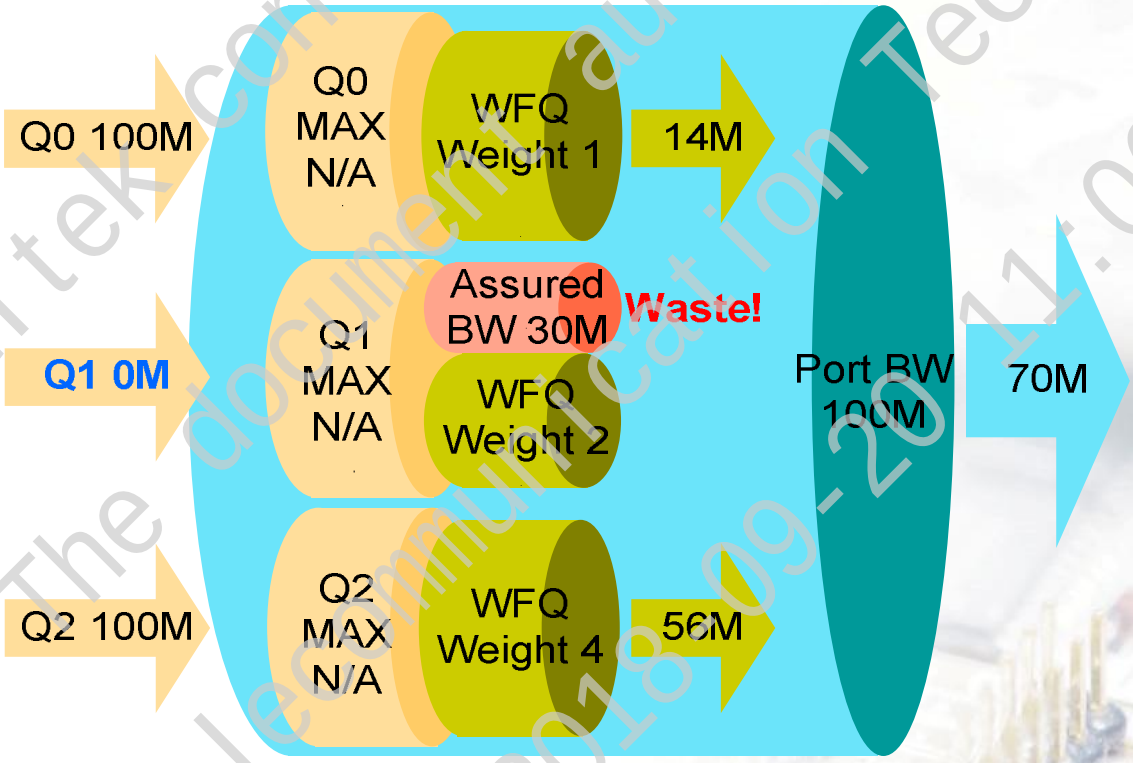
Configuration						
Port BW	Queue ID	WFQ Weight	Strict	Assured BW	Assured Mode	Maximum BW
100M	0	1	N/A	N/A	N/A	N/A
	1	2	N/A	30M	Share	N/A
	2	4	N/A	N/A	N/A	N/A



Assured Bandwidth (3/3)

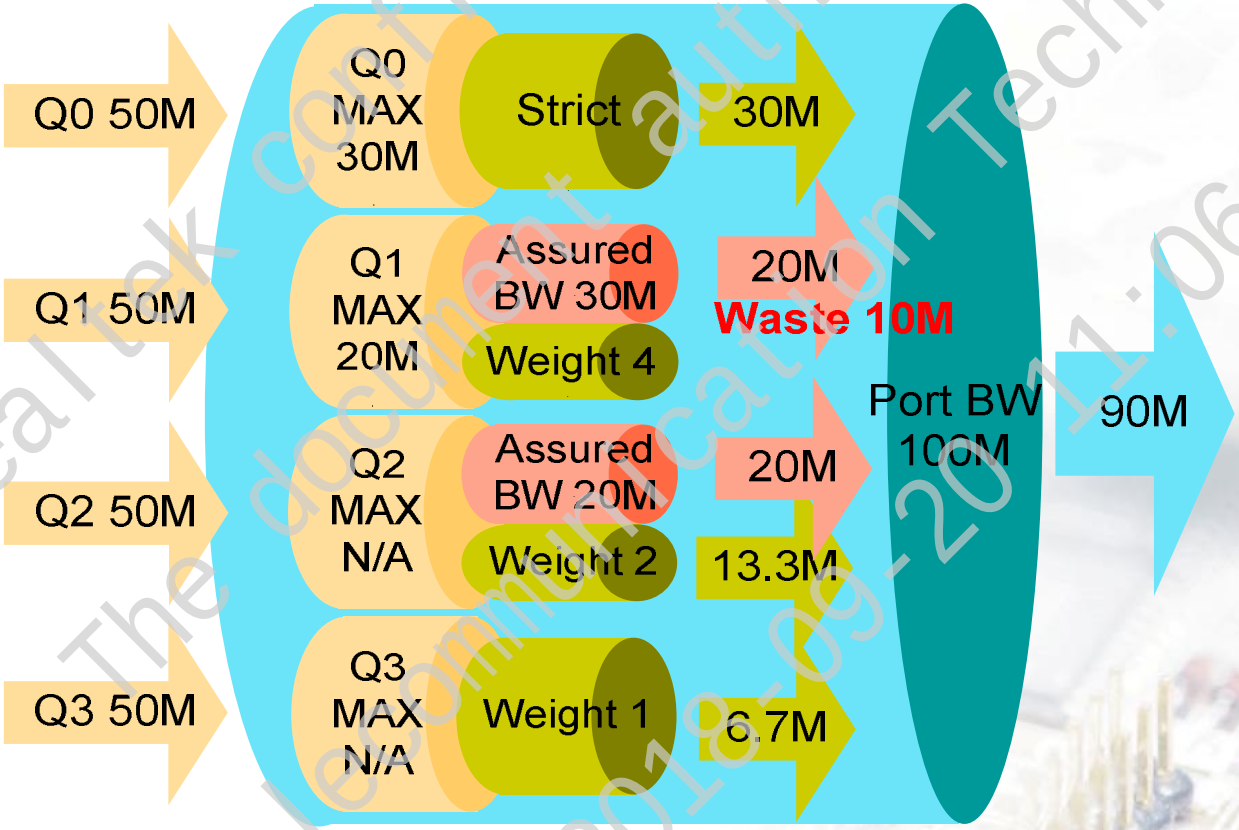
- Assured Bandwidth in fixed mode
  - Remain bandwidth is wasted

Configuration						
Port BW	Queue ID	WFQ Weight	Strict	Assured BW	Assured Mode	Maximum BW
100M	0	1	N/A	N/A	N/A	N/A
	1	2	N/A	30M	Fixed	N/A
	2	4	N/A	N/A	N/A	N/A



Mixed Example

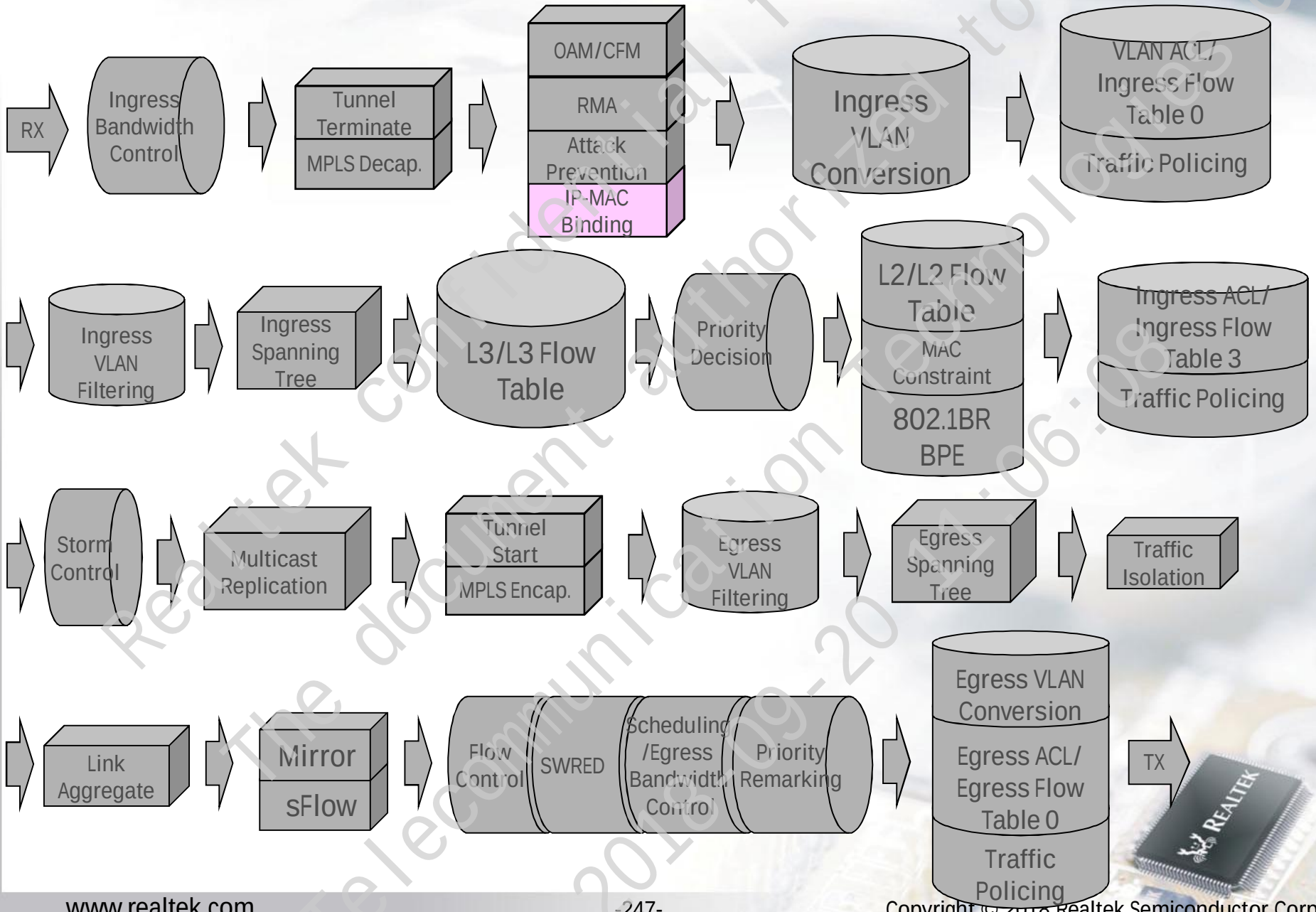
Configuration						
Port BW	Queue ID	WFQ Weight	Strict	Assured BW	Assured Mode	Maximum BW
100M	0	1	Yes	N/A	N/A	30M
	1	4	N/A	30M	Fixed	20M
	2	2	N/A	20M	Share	N/A
	3	1	N/A	N/A	N/A	N/A



Realtek Semiconductor Corp.

IP MAC Binding

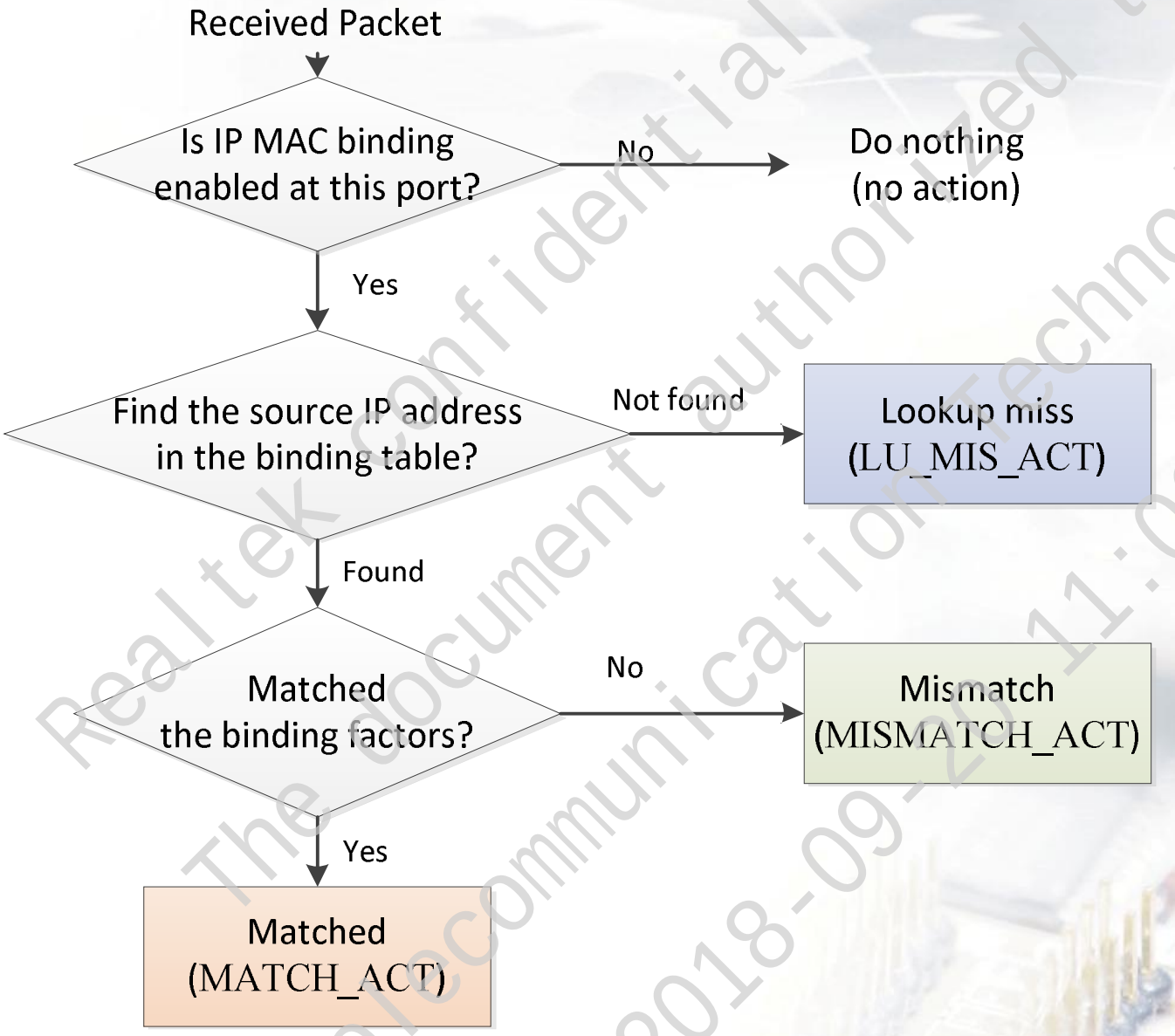
Packet Processing Pipeline



IP MAC Binding Overview

- IP MAC binding table
  - 1024 entries + 64 entries for hash collision
  - Extra 2048 entries (configurable table for MPLS, BPE, IP MAC binding)
- Each entry chooses the binding factors
  - IP + MAC
  - IP + MAC + PORT
  - IP + MAC + PORT + VLAN ID
- Action for lookup miss/ mismatched/ matched
  - Drop/Forward/Trap/Copy
- Per port enables IP MAC binding by
  - ARP packet
  - IP packet (excluding ARP)

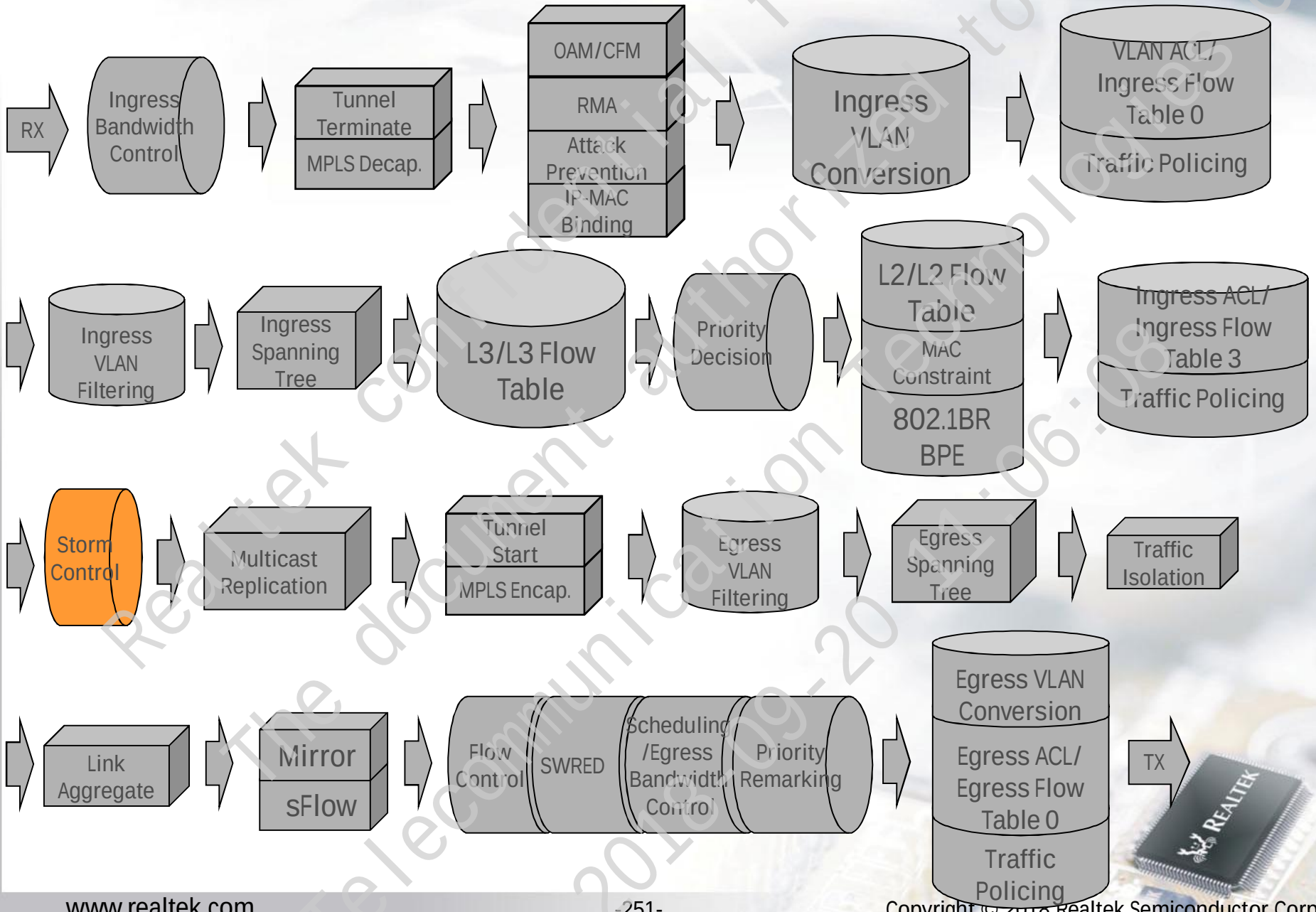
IP MAC Binding Process



Realtek Semiconductor Corp.

Storm Control

Packet Processing Pipeline



Storm Control (1/2)

- Global configuration
  - Include/Exclude IFG
  - Include/Exclude bypassed control packet
  - Bypass control packet
    - ARP/NS(Neighbor Solicitation)
    - RMA
    - BPDU
    - IGMP
    - RTK control packet (Ether type 0x8899)
    - DHCP
    - RIP & OSPF

Storm Control (2/2)

- Per port configuration
  - Flow type selection
    - Unicast: all or unknown only
    - Multicast: all or unknown only
    - Broadcast
  - Rate mode
    - PPS
    - BPS (unit 16 kbps)

- Per port per flow type configuration
  - Enable/Disable
  - Rate
  - Burst size
    - MAX to 64KBytes
  - Exceed flag

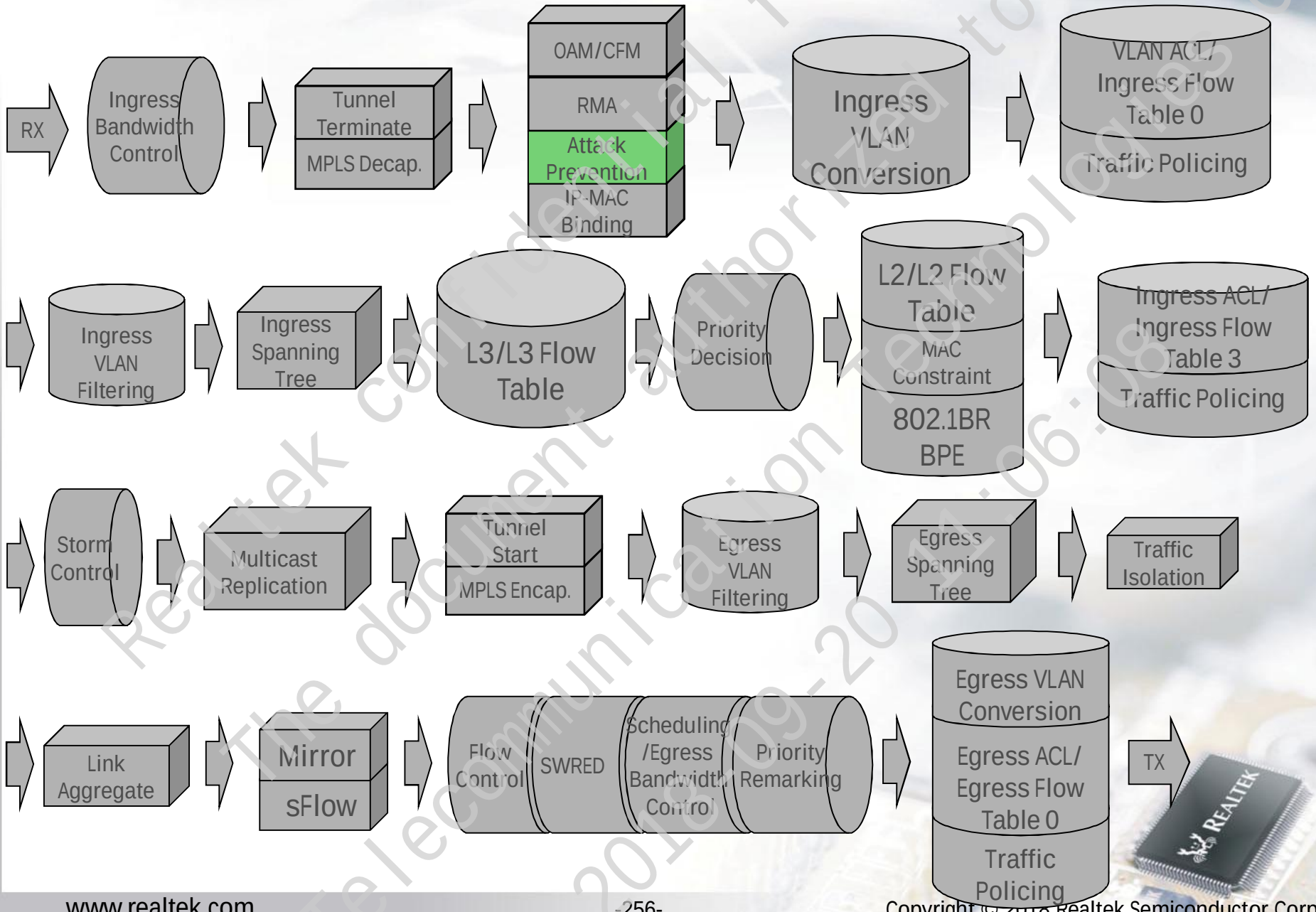
Protocol Storm Control

- Supported protocol type
  - DHCP
  - ARP Request/NS
  - BPDU
  - IGMP/MLD
- PPS rate mode only
- Per port per protocol configuration
  - Enable/Disable
  - Rate (0~511)
  - Burst size
  - Exceed flag
- VLAN constraint
  - Protocol storm control limits the rate for the traffic belonging to the activated VLAN group.
  - Global configuration
    - ARP
    - DHCP

Realtek Semiconductor Corp.

Attack Prevention

Packet Processing Pipeline



Attack Prevention

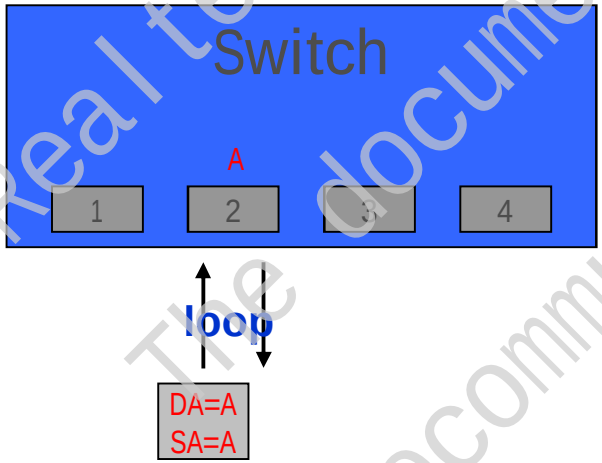
- Per-port enable/disable Attack-Prevention feature
- Globally configured the forwarding action for each Attacking Type
  - Drop / Trap / Forward

- Per-port configures ARP-Validation forwarding action
  - Drop / Trap / Forward

- SA is not learnt if the traffic is dropped/trapped by Attack-Prevention.

Attack Prevention

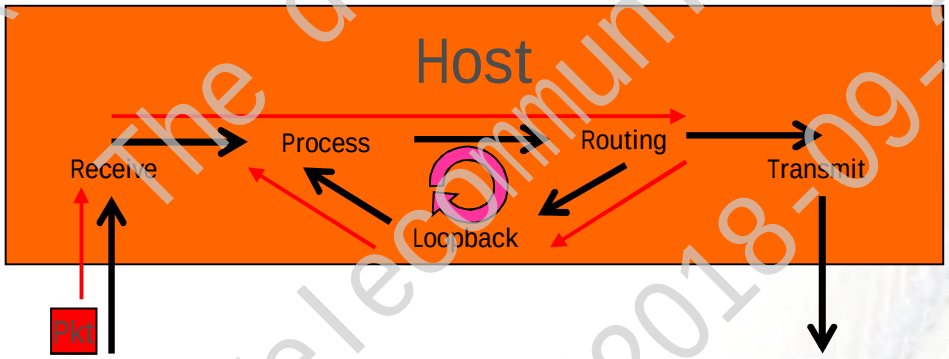
- Type: DA_EQUAL_SA
- Description:
  Both the source and the destination MAC addresses are the same.
- Issue:
  1. It is not reasonable, each MAC entity should have own unique address.
  2. It will consume the redundant bandwidth on the received port normally.
     (1) Switch learns its source address into L2 table, when received this packet.
     (2) Switch looks up the destination port with the destination address,
     It will send it out on the same port that learnt before.



1. An attack packet is received from Port 2.
2. Switch learns the source address (A) on Port 2.
3. Switch forwards the packet to Port 2 with the result of address lookup.
4. Cause a loop.

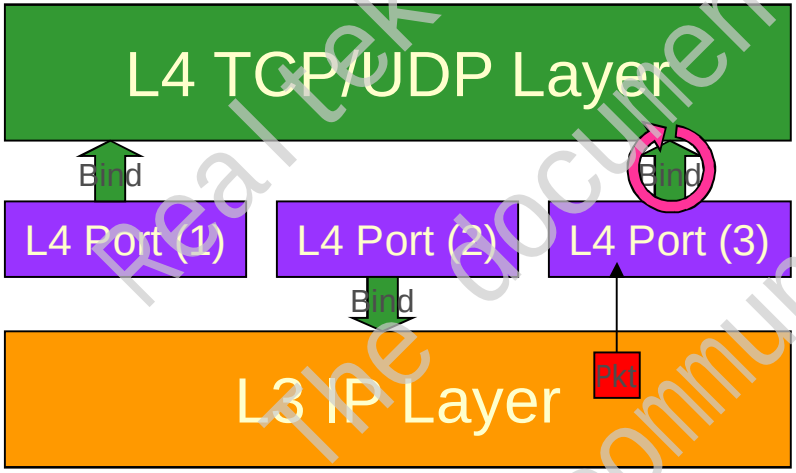
Attack Prevention

- Type: LAND
- Description:
  Both the source and the destination IPv4/IPv6 addresses are the same.
- Issue:
  1. Switch should not receive this type of packet, and it shouldn't appear outside of the host.
  2. It may be a malicious packet to attack a victim.
     (1) The attacker use the victim's IP address to manufacture a packet as his address, and send it to the victim.
     (2) If the victim doesn't check the rationality of the source address, it may cause a loop on the host of victim.
     (3) The victim may attempt to reply a large number packets from itself.



Attack Prevention

- Type: UDP_BLAT / TCP_BLAT (Blat Attack)
- Description:
  Both the source and the destination UDP/TCP port are the same.
- Issue:
  1. The attacks result from sending a specially crafted packet to a machine where the source host port is the same as the destination host port. The system may attempt to reply to itself, resulting in system lockup.

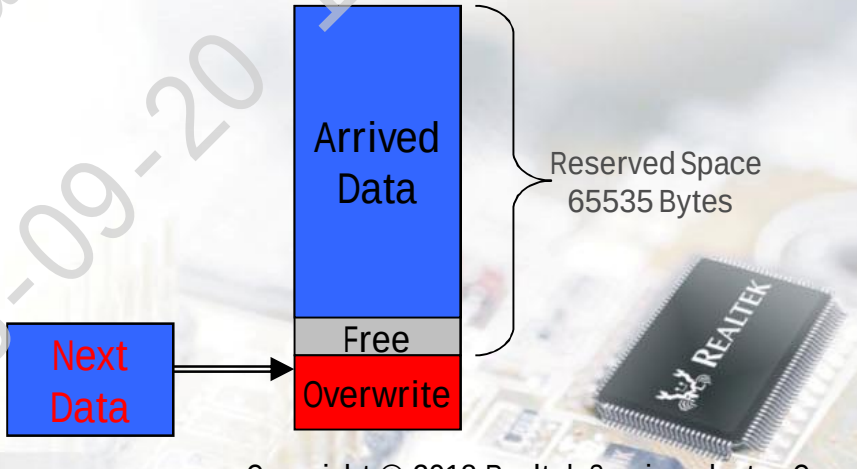


Simple Protocol Stack Model (Legacy)

1. Some traditional protocol stack are implemented simply (L4 port number bind to direction).
2. An attack packet is received from L3 IP.
3. According to the source port (3) to transfer the packet to TCP/UDP module.
4. The TCP/UDP module will reply a response packet to the destination port (3).
5. Because the port (3) already has bound to L4 TCP/UDP module, it will loopback to them.

Attack Prevention

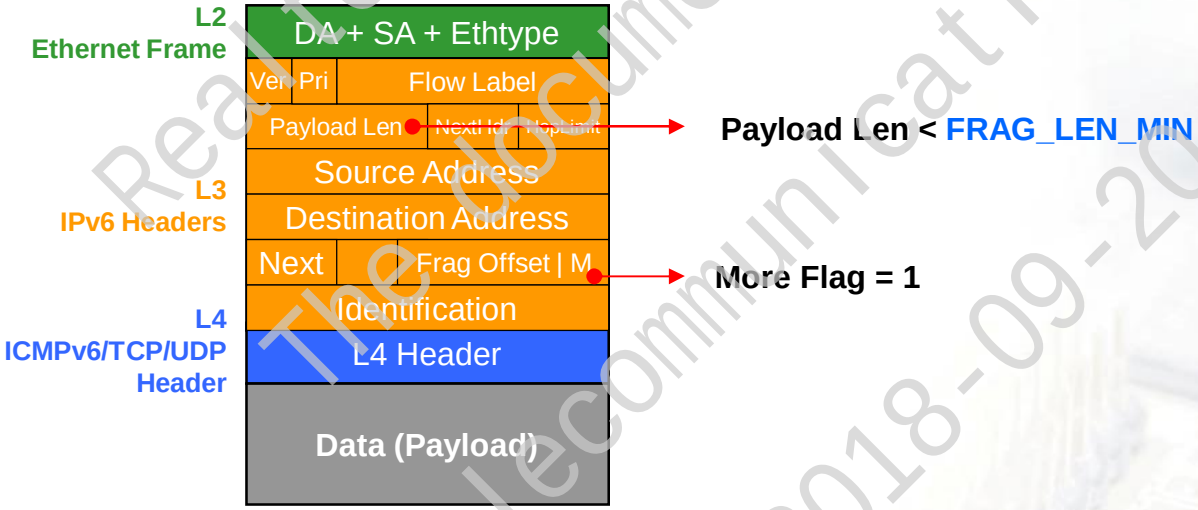
- Type: POD (Ping of Death)
- Description:
  Ping packets that length are larger than 65535 bytes.
- Issue:
  1. The Ping of Death attack relied on a bug in the Berkeley TCP/IP stack which also existed on most systems which copied the Berkeley network code.
  2. The attacker will send ping packets larger than 65535 bytes to the victim.
  3. The most protocol stacks reserve only 65535 bytes space to receive the sending data (It is the reasonable space to be used enough).
     If the receiver don't check the used length carefully, it may overwrite the other space that it should not use.



Attack Prevention

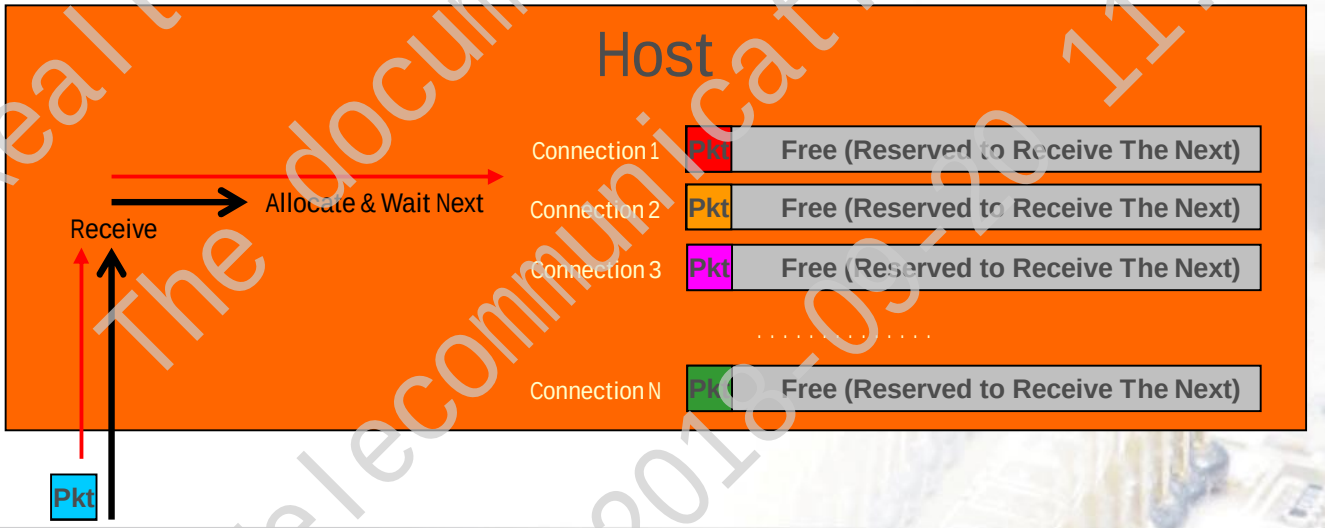
- Type: IPV6_FRAG_LEN_MIN
- Description:
  IPv6 fragmented packets (not including the last one) that payload length less than 1240 bytes.

  (ATK_PRVNT_IPV6_CTRL.FRAG_LEN_MIN, default: 1240, Range: 0 ~ 65535)
- Issue:
  1. Since the minimum MTU size that is to be supported for IPv6 is 1280 bytes.
     Note: MTU (1280 B) -- IPv6 Header (40 B) = IPv6 Payload (1240 Bytes)
  2. IPv6 requires that every link in the internet have an MTU of 1280 octets or greater. (Refer to RFC 2460, Section 5.)



Attack Prevention

- Type: ICMP_FRAG_PKT
- Description:
  Fragmented ICMP packets.
- Issue:
  1. There is a mass of resources to handle a reassembling the fragmented packets to be a completed original packet.
  2. Attack may send many fragmented ICMP packets with different connections, The receiver will consume and occupy a lot of memory to process it.
  3. This is a common way of denial-of-service attack.

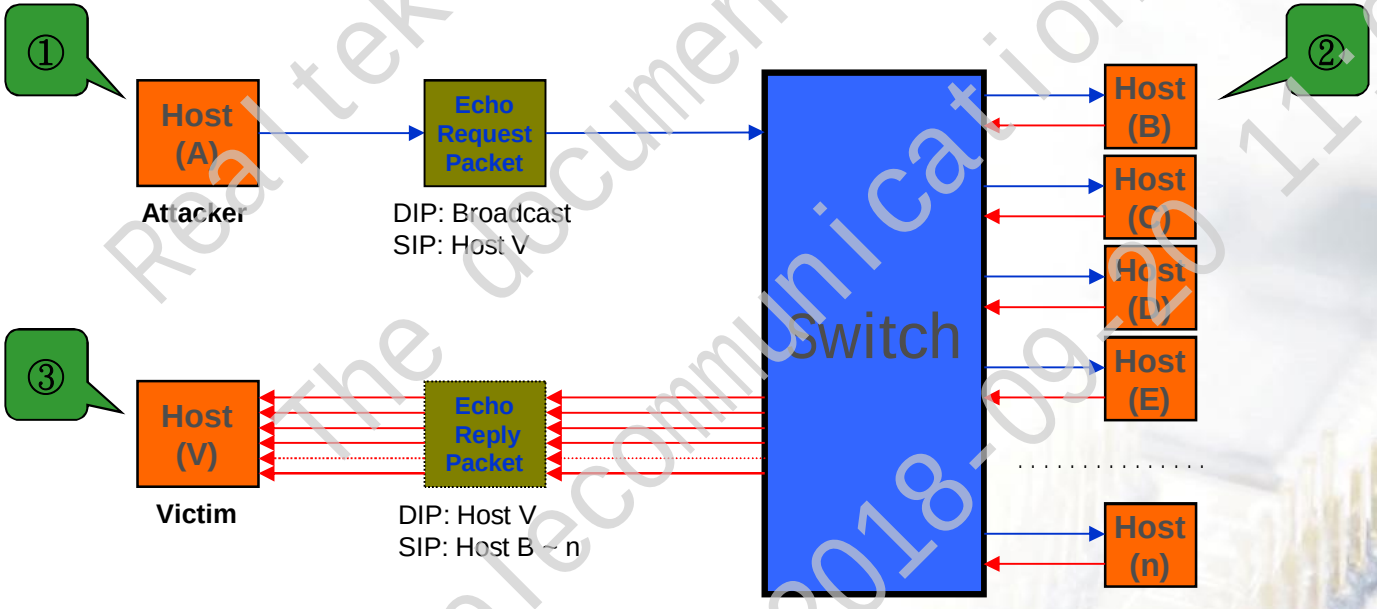


Attack Prevention

- Type: ICMPV4_PING_MAX / ICMPV6_PING_MAX
- Description:
  PING packet with the length over register configured.
  (ATK_PRVNT_ICMP_CTRL.PKT_LEN_MAX, default: 512 Bytes, Range: 0 ~ 65535)
- Issue:
  1. Generally, The ping packet should be small.
  2. This attack is similar to Ping of Death, but more strict.
  3. The length can be configured from 0 to 65535 bytes.
  4. It can filter the ping packet of ECHO request or reply that length is longer than the pre-configured value.

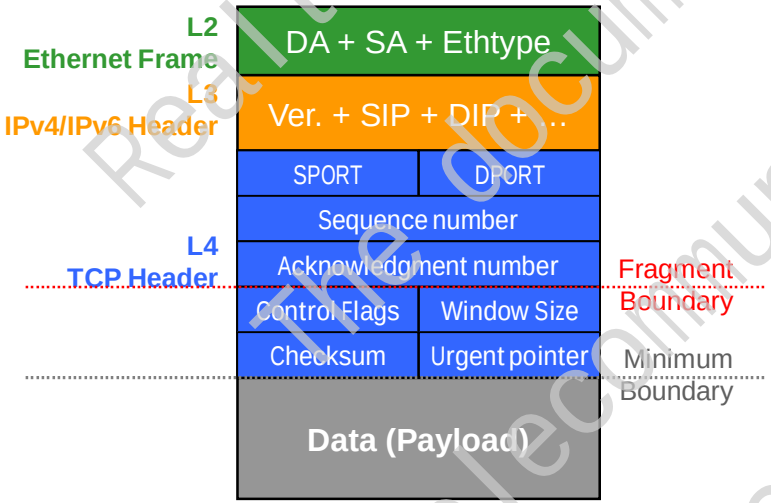
Attack Prevention

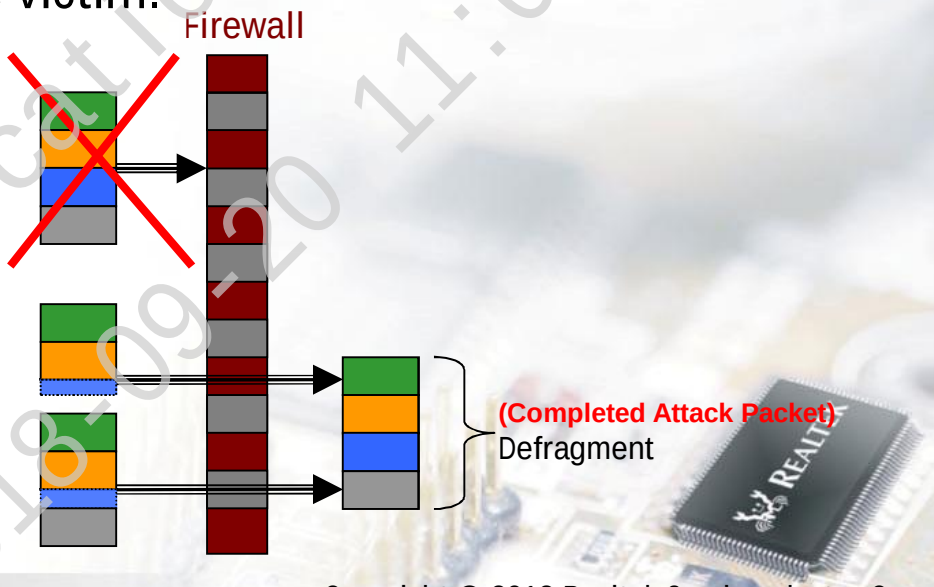
- Type: SMURF
- Description:
  ICMP echo request packet that destination IPv4 address is broadcast address.
  (ATK_PRVNT_SMURF_CTRL.NETMASK, default: 0.0.0.0, Range: 0.0.0.0 ~ 255.255.255.255)
- Issue:
  1. This is a type of denial-of-service attack that floods a target system via spoofed broadcast ping messages.
  2. This attack relies on a perpetrator sending a large amount of ICMP echo request (ping) traffic to IP broadcast addresses, all of which have a spoofed source IP address of the intended victim.



Attack Prevention

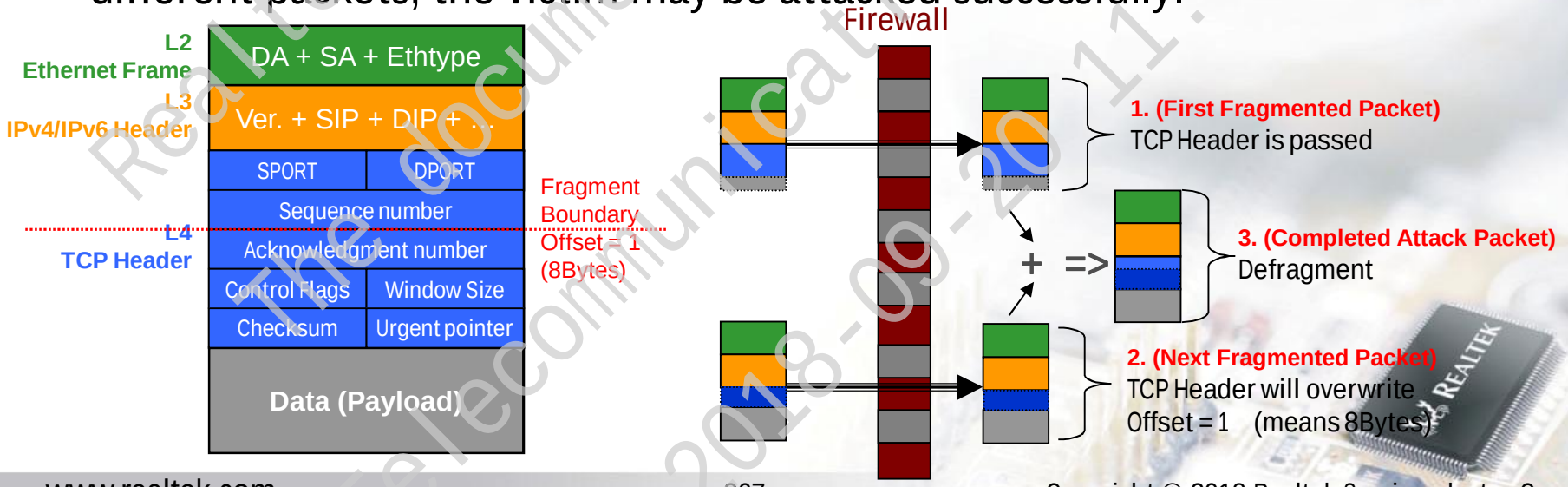
- Type: TCP_HDR_LEN_MIN
- Description:
  TCP packet that header length is less than the register configured.
  (ATK_PRVNT_TCP_CTRL.HDR_LEN_MIN, default: 20B, Range: 0~31)
- Issue:
  1. The length of a completed TCP packet should be at least 20 bytes.
  2. If the TCP header length is less than 20 bytes, the switch/firewall cannot check the main fields (likes port number, control flags) to detect the attack.
  3. Some attacker can make an attack packet using this way to avoid the security checking, but still affect the victim.





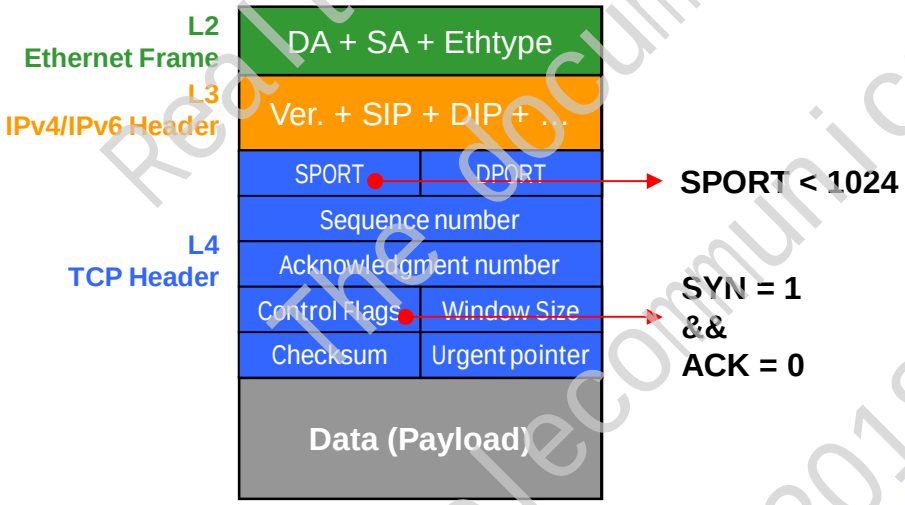
Attack Prevention

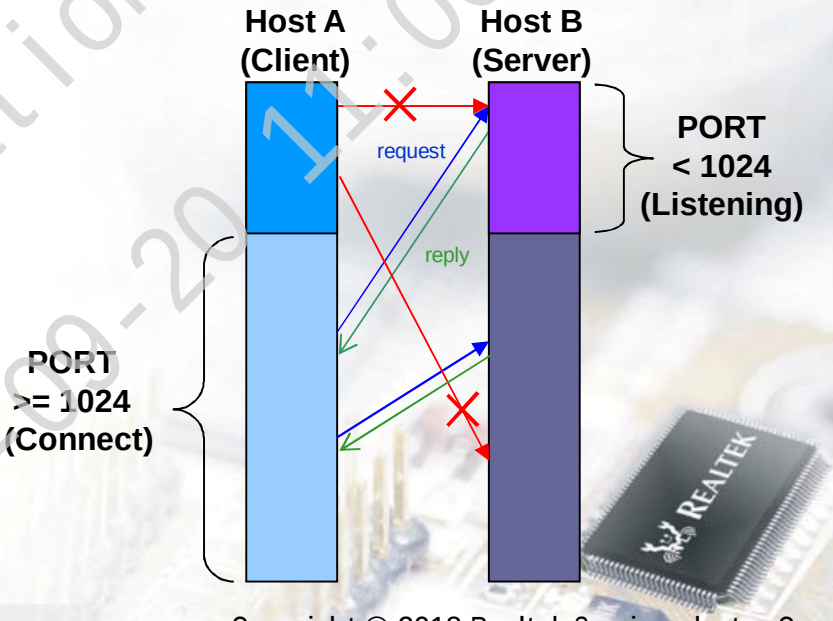
- Type: TCP_FRAG_OFF_MIN
- Description:
  Fragmented TCP packets with offset = 1 (means 8 bytes).
- Issue:
  1. The attacker send an attack packet with this way to hide the TCP control flags info, this can avoid the detection of attack to check the control flags.
  2. Because the most firewall just check the first fragmented packet, it should have the completed TCP header.
  3. By this way, the attacker will make the TCP header to be fragmented in different packets, the victim may be attacked successfully.



Attack Prevention

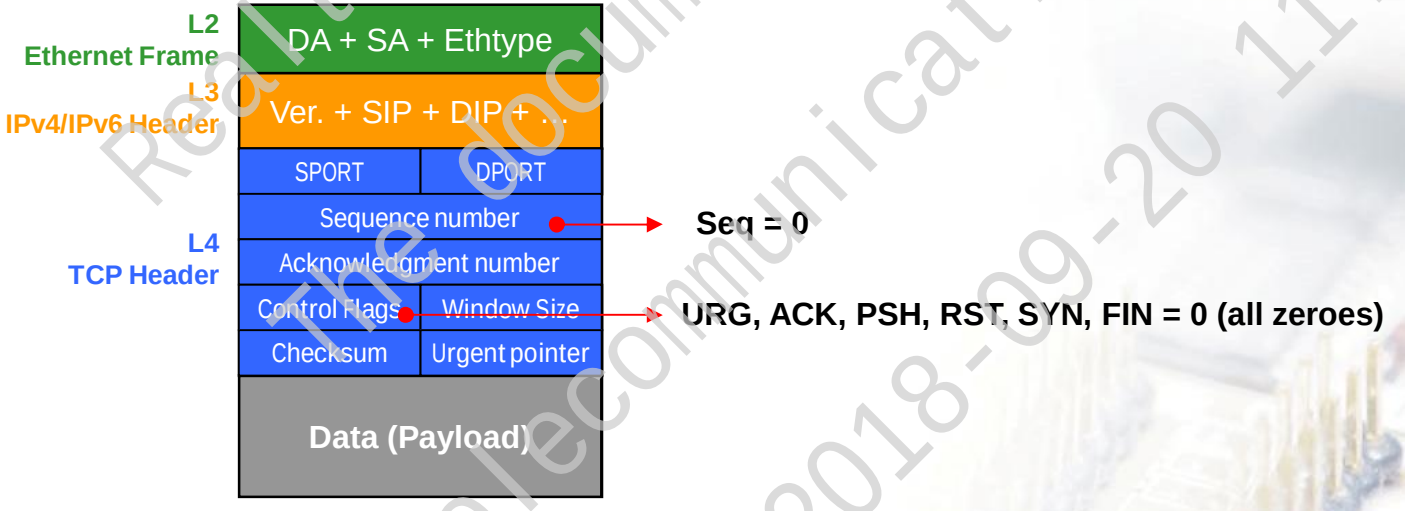
- Type: SYN_SPORT_LESS_1024
- Description:
  TCP SYN packets with source port less than 1024.
- Issue:
  1. Generally, the port number less than 1024 is the server port.
  2. Server port should be listening state, will not connect someone actively.
  3. It is not reasonable to appear on the TCP connection normally, so it could be the counterfeit packet.





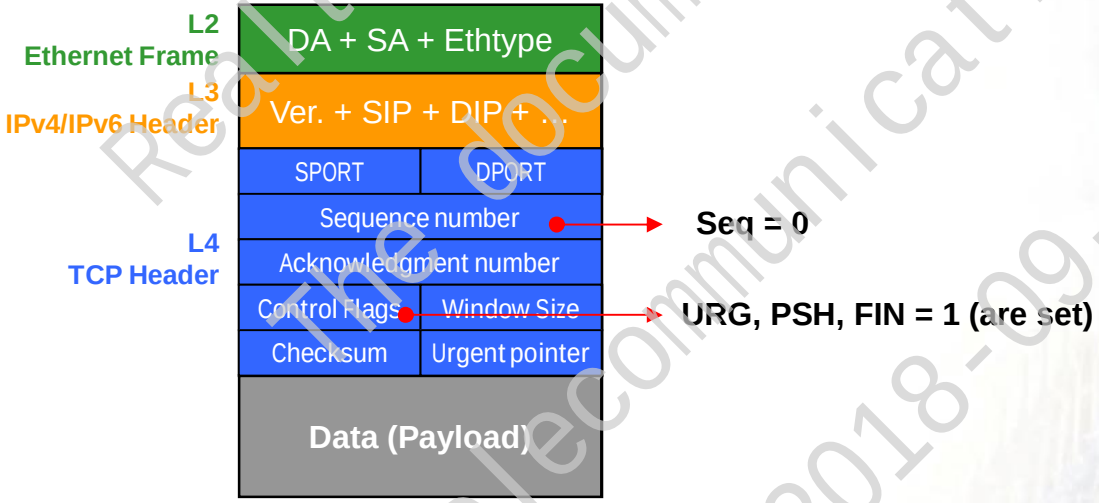
Attack Prevention

- Type: NULL_SCAN (Scan Attack)
- Description:
  TCP sequence number is zero, and all control flags are zeroes.
- Issue:
  1. Generally, at least one flag has been set.
  2. SYN flag should be set at beginning, and ACK flag should be set except the first packet of the connection.
  3. In some protocol stack, they may not handle this kind of packet correctly.



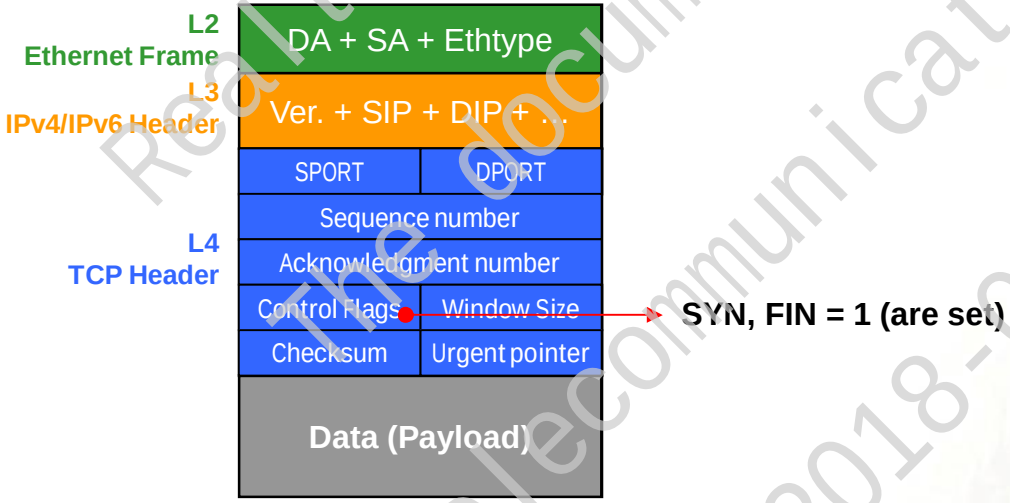
Attack Prevention

- Type: XMAS (Scan Attack)
- Description:
  TCP sequence number is zero, and the FIN/URG/PSH flags are set.
- Issue:
  1. The FIN URG and PSH flags should never be seen together in normal TCP traffic.
  2. Typically, a closed port will respond with an ACK RST packet, whereas an open port may not respond at all. (It can be used to scan ports)



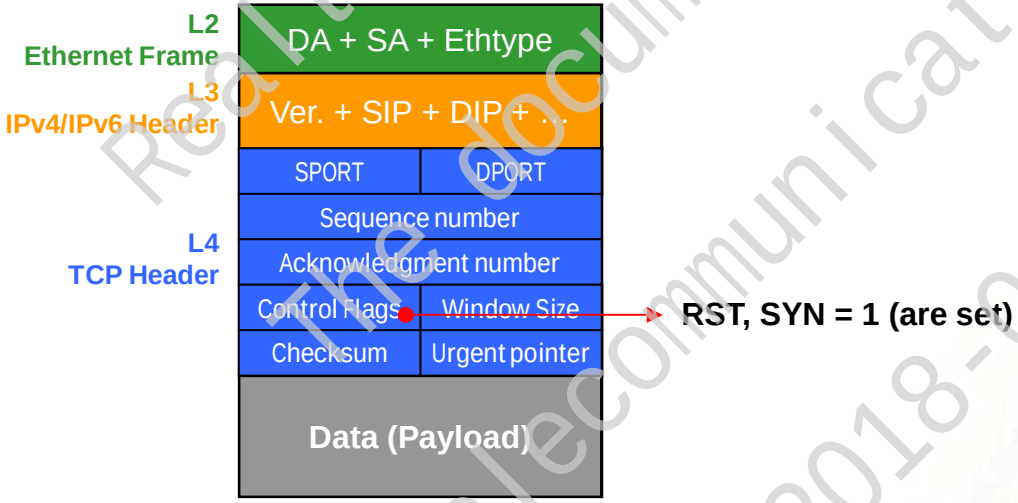
Attack Prevention

- Type: SYN_FIN
- Description:
  A TCP packet with the SYN and FIN flags set.
- Issue:
  1. SYN flag means trying to create a new connection, and FIN flag means trying to close this connection.
  2. It's not reasonable that SYN/FIN flags are set at the same time.
  3. This situation hasn't been defined in RFC document, For each protocol stack has different implementation or flaw.



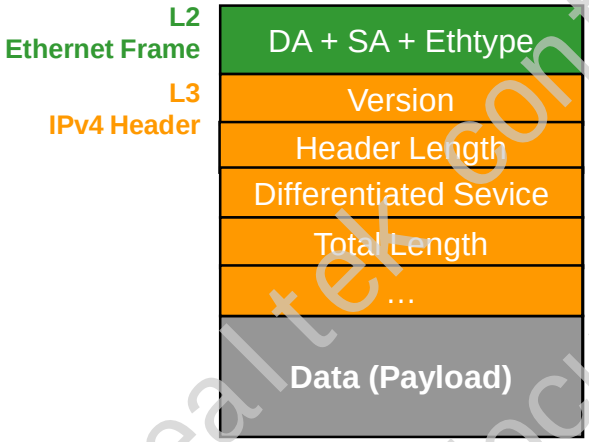
Attack Prevention

- Type: SYN_RST
- Description:
  A TCP packet with the SYN and RST flags set.
- Issue:
  1. SYN flag means trying to create a new connection, and RST flag means trying to reset this connection (reply to notify that this TCP port is not open).
  2. It's not reasonable that SYN/RST flags are set at the same time.
  3. This situation hasn't been defined in RFC document, For each protocol stack has different implementation or flaw.



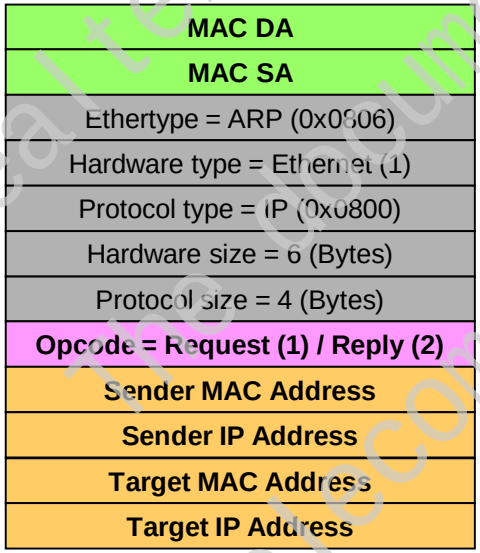
Attack Prevention

- Type: IPV4_INVLD(INVALID_LEN)
- Description:
  Drop/forward/trap packet with header length is greater than total length.
- Note:
  1. When Header Length * 5 > Total Length, the attack happens.



Attack Prevention

- Type: ARP_VALIDATION
- Description:
  Invalid (IPv4) ARP packet that has any one of the following conditions:
  1. For Request Packet:
     (1) MAC SA != ARP Sender MAC Address.
     (2) ARP Sender IP = all-zero or multicast or broadcast (255.255.255.255) IP address.
  2. For Reply Packet:
     (1) MAC SA != ARP Sender MAC Address.
     (2) ARP Target MAC Address = all-zero or multicast address.
     (3) ARP Target MAC Address != MAC DA.
     (4) ARP Sender IP = all-zero or multicast or broadcast (255.255.255.255) IP address.
     (5) ARP Target IP = all-zero or multicast or broadcast (255.255.255.255) IP address.
- Note:
  1. Independent configuration: ARP validation type can set the action (Forward/Drop/Trap) for each port, it's different from other types.



Attack Prevention Precedence

ID	Type
1	DA_EQUAL_SA
2	LAND
3	UDP_BLAT
4	TCP_BLAT
5	POD (Ping of Death)
6	IPV6_FRAG_LEN_MIN
7	ICMP_FRAG_PKT
8	ICMPV4_PING_MAX
9	ICMPV6_PING_MAX
10	SMURF
11	TCP_HDR_LEN_MIN
12	SYN_SPORT_LESS_1024
13	NULL_SCAN
14	XMAS
15	SYN_FIN
16	SYN_RST
17	TCP_FRAG_OFF_MIN
18	IPV4_INVLD(INVALID_LEN)
19	ARP-Validation

High

Low

1. The action priority is (Drop > Trap):
   if multiple types are confirmed, but the action of one is drop and another one is trap.
   => the packet will be dropped.

2. The checking priority is (Lower id > Higher ID):
   if multiple type are confirmed, and their actions are all traps. => the attack type will be identify to the lowest ID of the all confirmed types.

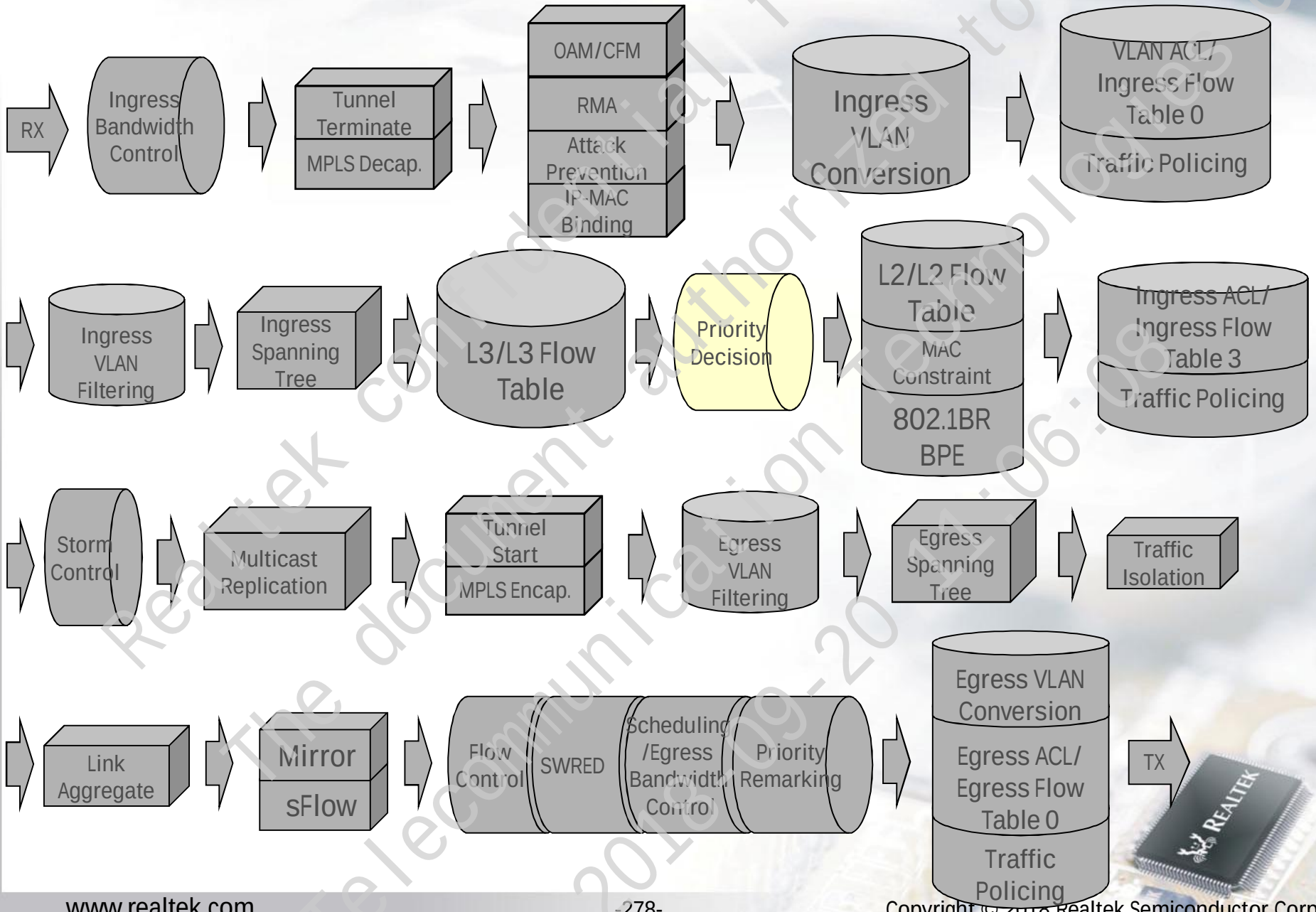
Attack Prevention Review

Type	Criteria
DA_EQUAL_SA	MAC.DA = MAC.SA in an Ethernet frame.
ARP-Validation	Invalid ARP packet. (Note: it has its own independent settings to enable or disable by per port)
IPV4_INVLD(INVALID_LEN)	Drop/forward/trap packet with header length is greater than total length
LAND	SIP = DIP in an IPv4/IPv6 datagram.
POD (Ping of Death)	For an IPv4/IPv6, (IP payload length + fragment offset) > 65535.
IPV6_FRAG_LEN_MIN	Fragment size of IPv6 datagram < IPV6.FRAG_LEN_MIN (Range: 0~65535)
ICMP_FRAG_PKT	ICMP(v4/v6) protocol data are carried in a fragmented IP(v4/v6) datagram.
ICMPV4_PING_MAX	(Total length field – header length) > ICMP.PKT_LEN_MAX (Range: 0~65535)
ICMPV6_PING_MAX	Payload length > ICMP.PKT_LEN_MAX (Range: 0~65535)
SMURF	ICMPv4 echo request packet with broadcast address.
UDP_BLAT	UDP SPORT = DPORT of an IPv4/IPv6 packet.
TCP_BLAT	TCP SPORT = DPORT of an IPv4/IPv6 packet.
TCP_HDR_LEN_MIN	TCP header length < TCP.HDR_LEN_MIN (Range: 0~31)
TCP_FRAG_OFF_MIN	TCP packet with IP fragment offset = 1 (means 8 Bytes)
SYN_SPORT_LESS_1024	TCP Control flag SYN = 1 & ACK = 0 & TCP SPORT < 1024.
NULL_SCAN	TCP Seq_Num = 0, All control flags are zeroes.
XMAS	TCP Seq_Num = 0, Control flag FIN = 1 & URG = 1 & PSH = 1
SYN_FIN	TCP Control flag SYN = 1 & FIN = 1
SYN_RST	TCP Control flag SYN = 1 & RST = 1

Realtek Semiconductor Corp.

Priority Decision

Packet Processing Pipeline

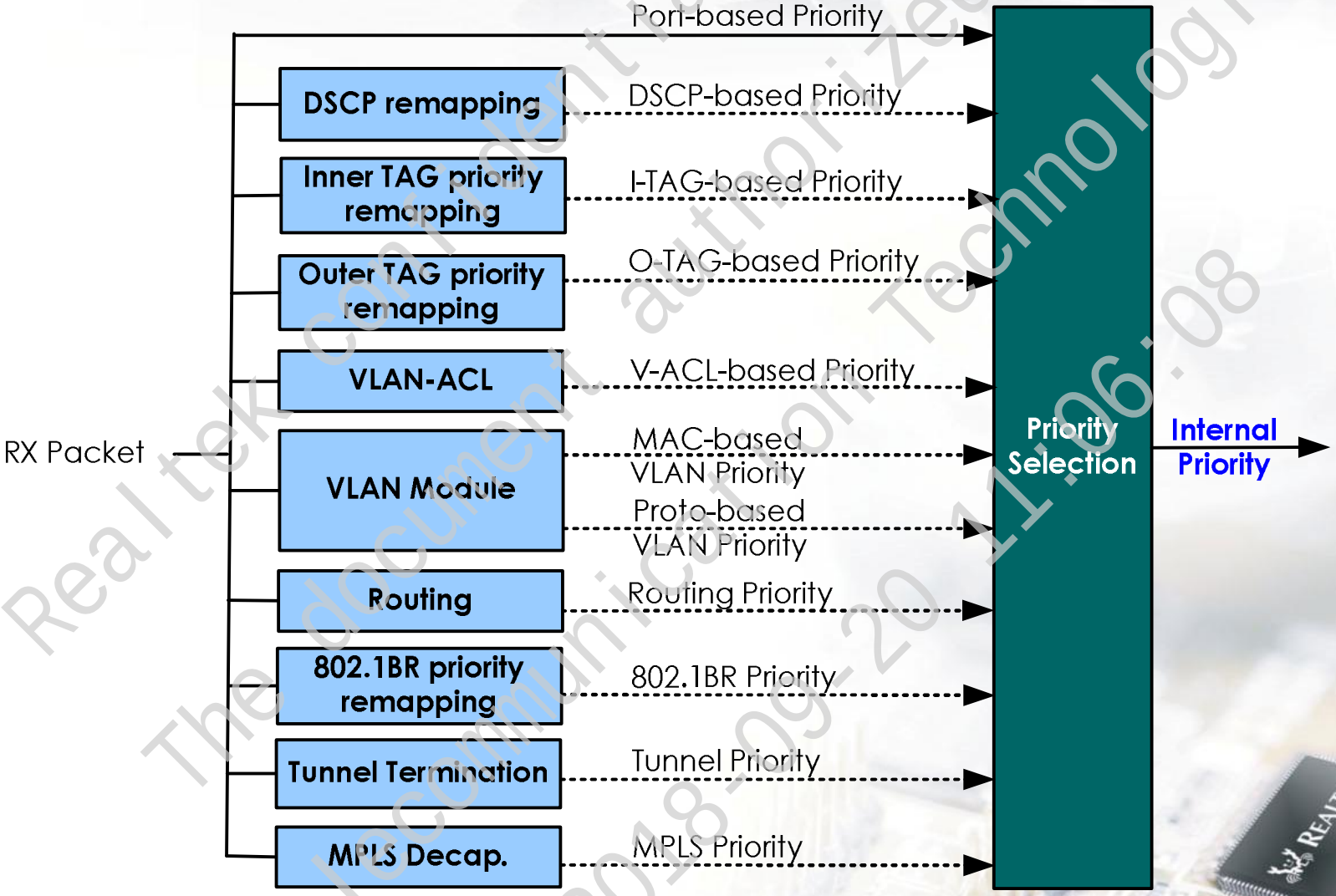


Priority Assignment Overview (1/2)

- 11 priority sources and each source can be specified a weight
  - Port-based
  - DSCP field
  - Inner 802.1p priority
  - Outer 802.1p priority
  - VLAN ACL (V-ACL)
  - MAC-based/IP-subnet-based VLAN
  - Protocol-and-port-based VLAN
  - Routing
  - 802.1BR
  - Tunnel
  - MPLS
- System decides an internal priority from the 11 sources for mapping egress queue
- 4 priority selection tables
  - Per port index to 1 of these 4 tables

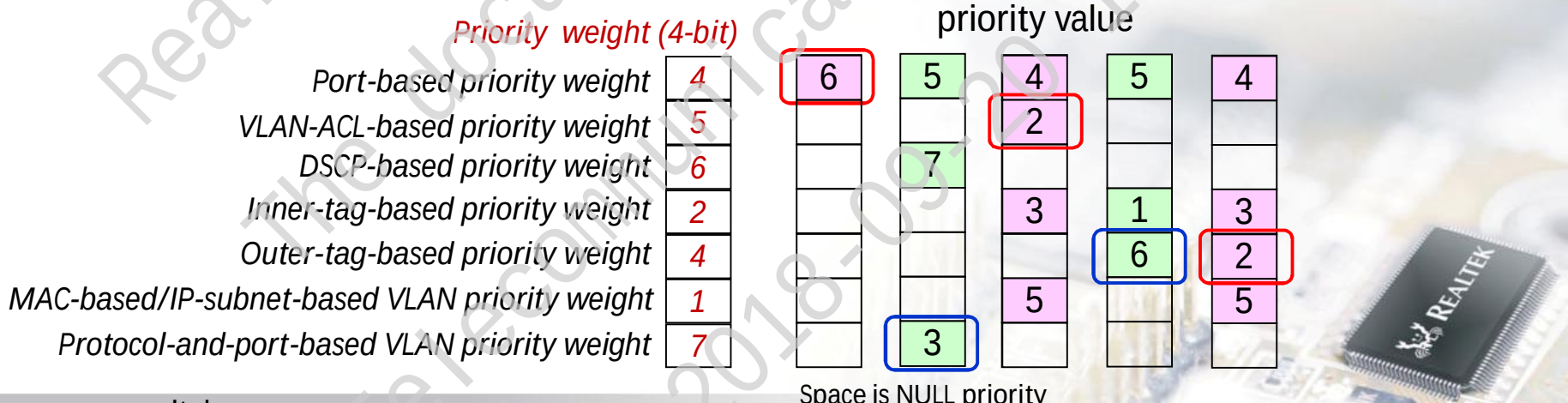
Priority Assignment Overview (2/2)

- Priority selection module



Arbitration Weight

- Per-port selects 1 of 4 global selection tables
- Priority weight for each source:
  - 1~12 (12 is highest)
  - 0: ignore the priority source. If all sources are ignored, Internal Priority will be 0.
- Priority value of particular source may be NULL(ignore) except port-based priority
- Precedence for the same weight
  - VLAN ACL-based > Routing > MAC-based(IP-subnet-based) VLAN > Proto-based VLAN > DSCP-based > OTAG-based > ITAG-based > 802.1BR-based > Tunnel-based > MPLS-based > Port-based
  - Example :



Port-based & DSCP-based

- Port-based
  - Per-port priority configuration

Port	Port-based Priority
0	0
...	...
10	7
...	...
56	1

Input

Output

- DSCP-based
  - Per-DSCP configure
  - Global Invalid DSCP configuration
    - If packet's DSCP hits the configured value, the DSCP-based priority is treated as NULL.

DSCP	DSCP-based Priority
0	0
...	...
23	3
...	...
63	1

Input

Output

I-TAG-based & O-TAG-based

- I-TAG-based
  - Global table for I-DEI 0 and 1
  - Per-priority configuration

I-DEI	I-Priority	I-TAG-based Priority
0	0	1
0	1	0
...	...	...
0	7	0
1	0	3
1	...	...
1	7	7

Input

Output

- O-TAG-based
  - Global table for O-DEI 0 and 1
  - Per-priority configuration

O-DEI	O-Priority	O-TAG-based Priority
0	0	1
...	...	...
0	7	0
1	0	3
1	...	...
1	7	7

Output

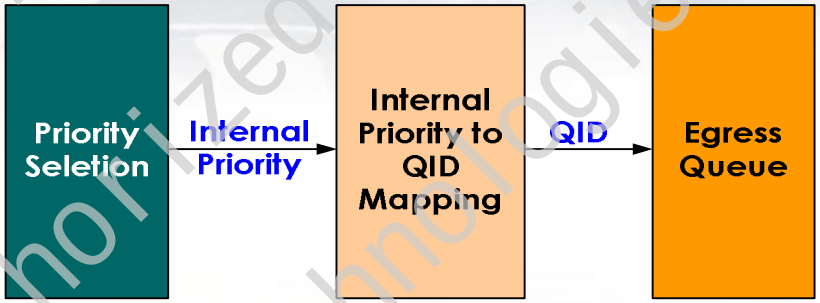
Input

Other Source

- VLAN-ACL-based
  - Rule with assigning priority action
- Routing
  - Packet hits routing table
- MAC-based(IP-subnet-based) VLAN
  - Packet hits MAC-based VLAN rule
- Protocol-and-port-based VLAN
  - Packet hits Protocol-and-port-based VLAN rule
- 802.1BR-based
  - Packet with E-Tag
- Tunnel-based
  - Packet hits Tunnel Termination
- MPLS-based
  - Packet hits MPLS decapsulation

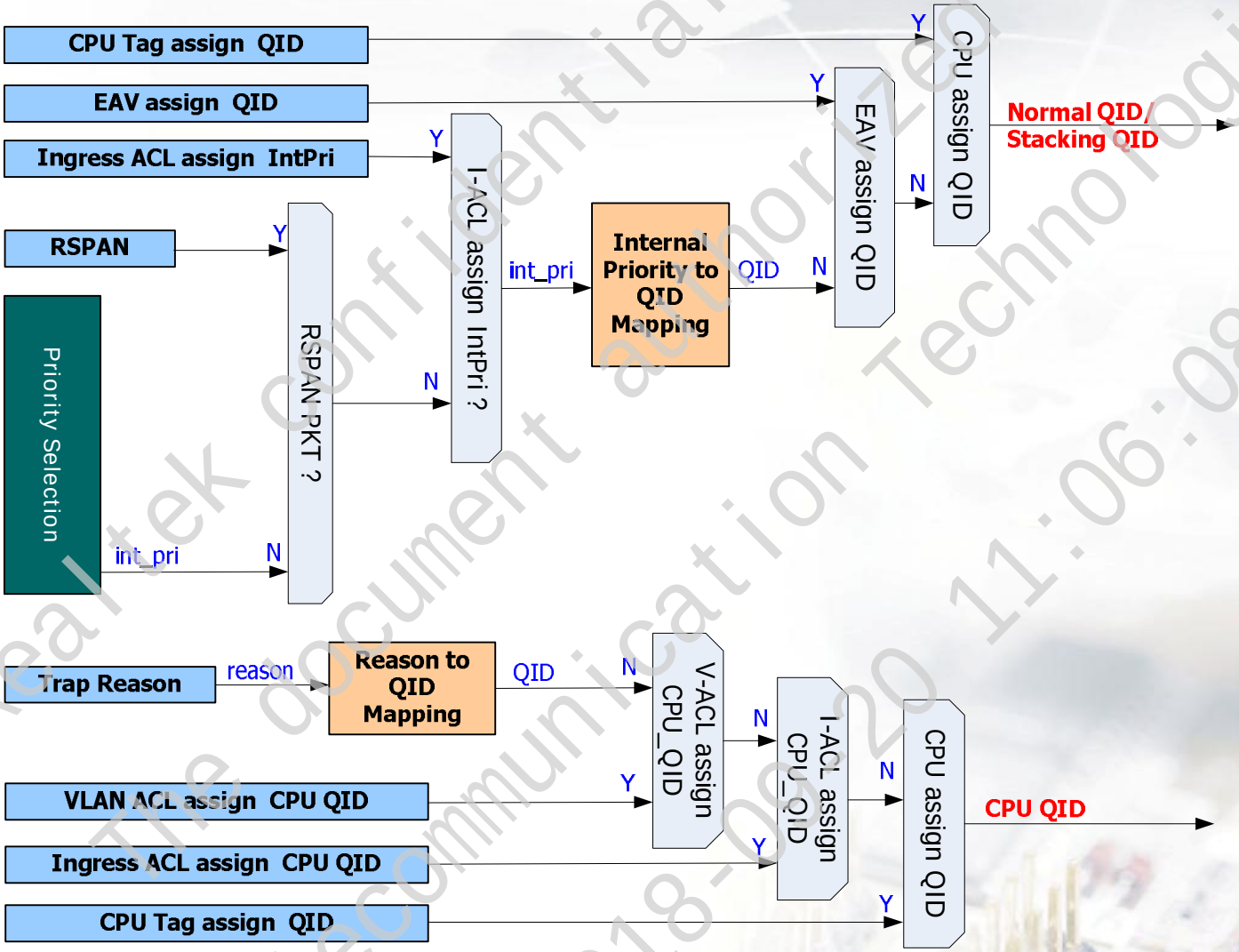
Egress Queue Decision

- Queue Number
  - Normal Port – 8
  - Stacking Port – 12
  - CPU Port – 32
- Internal Priority to Queue Mapping Table
  - 8 Priority → 8 Queue ID
- Packets to CPU Queue are decided by Trap Reason



- For Stacking Application
  - 32 CPU Queue to 8 Queue Mapping Table
    - CPU to CPU communication through non-uplink stacking port
  - 32 CPU Queue to 12 Queue Mapping Table
    - CPU to CPU communication through uplink stacking port

Internal Priority & Queue Decision



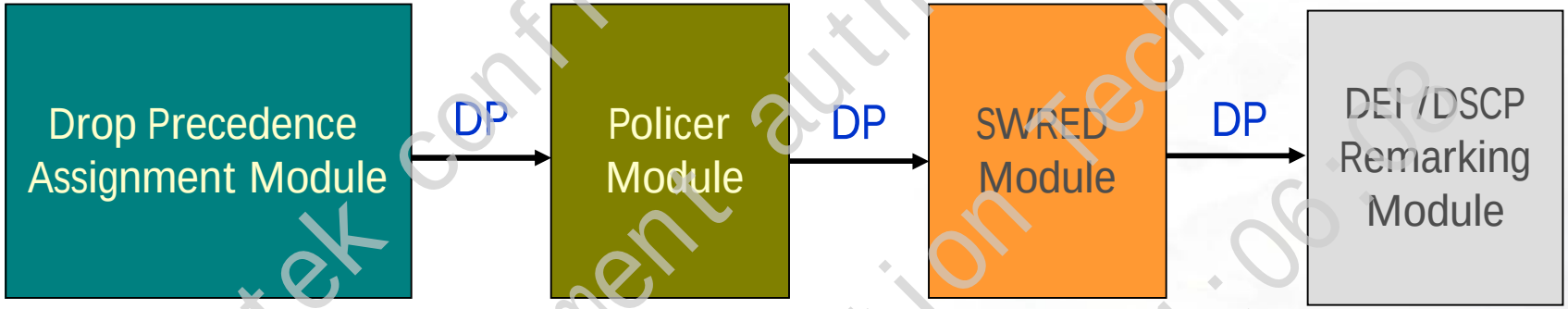
Realtek Semiconductor Corp.

Drop Precedence

Drop Precedence Overview

- 3 DP
  - 0: Green
  - 1: Yellow
  - 2: Red
- 4 drop precedence(DP) sources in system configuration
  - Inner Tag
  - Outer Tag
  - DSCP-based
  - MPLS-based
- DP may be remarked by ACL Meter
- SWRED utilizes DP to determine the drop probability and threshold
- DEI/DSCP remarking also utilizes DP to be remarking table source

Drop Precedence Overview



I-TAG-based & O-TAG-based

- I-TAG-based
  - Global table for I-DEI 0 and 1
  - Per-priority configuration

I-DEI	I-Priority	I-TAG-based DP
0	0	1
0	1	0
...	...	...
0	7	0
1	0	2
1	...	...
1	7	2

Input

Output

- O-TAG-based
  - Global table for O-DEI 0 and 1
  - Per-priority configuration

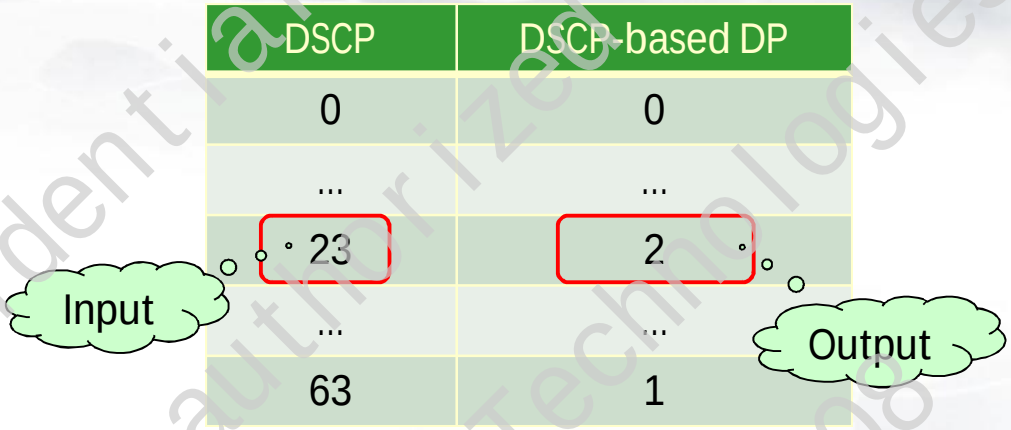
O-DEI	O-Priority	O-TAG-based DP
0	0	1
...	...	...
0	7	0
1	0	2
1	...	...
1	7	2

Output

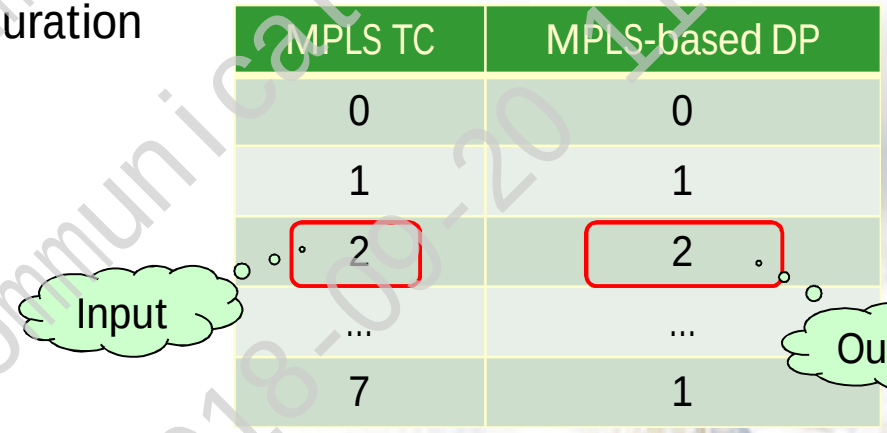
Input

DSCP-based & MPLS-based

- DSCP-based
  - Per-DSCP configuration

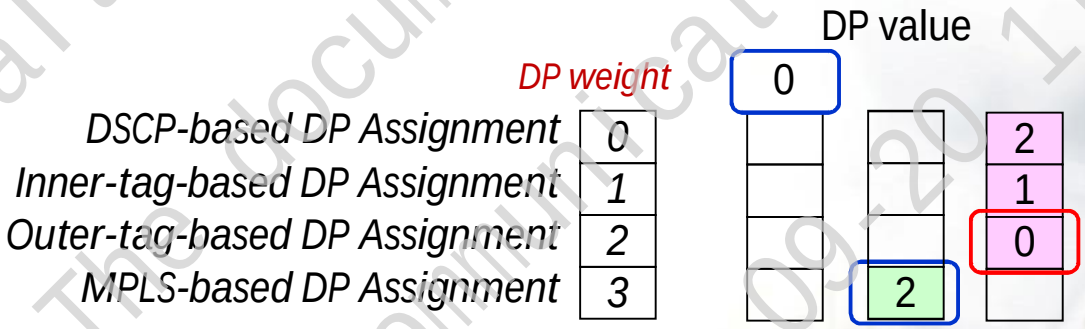


- MPLS-based
  - Per-Traffic Class(TC) configuration



DP Source Decision

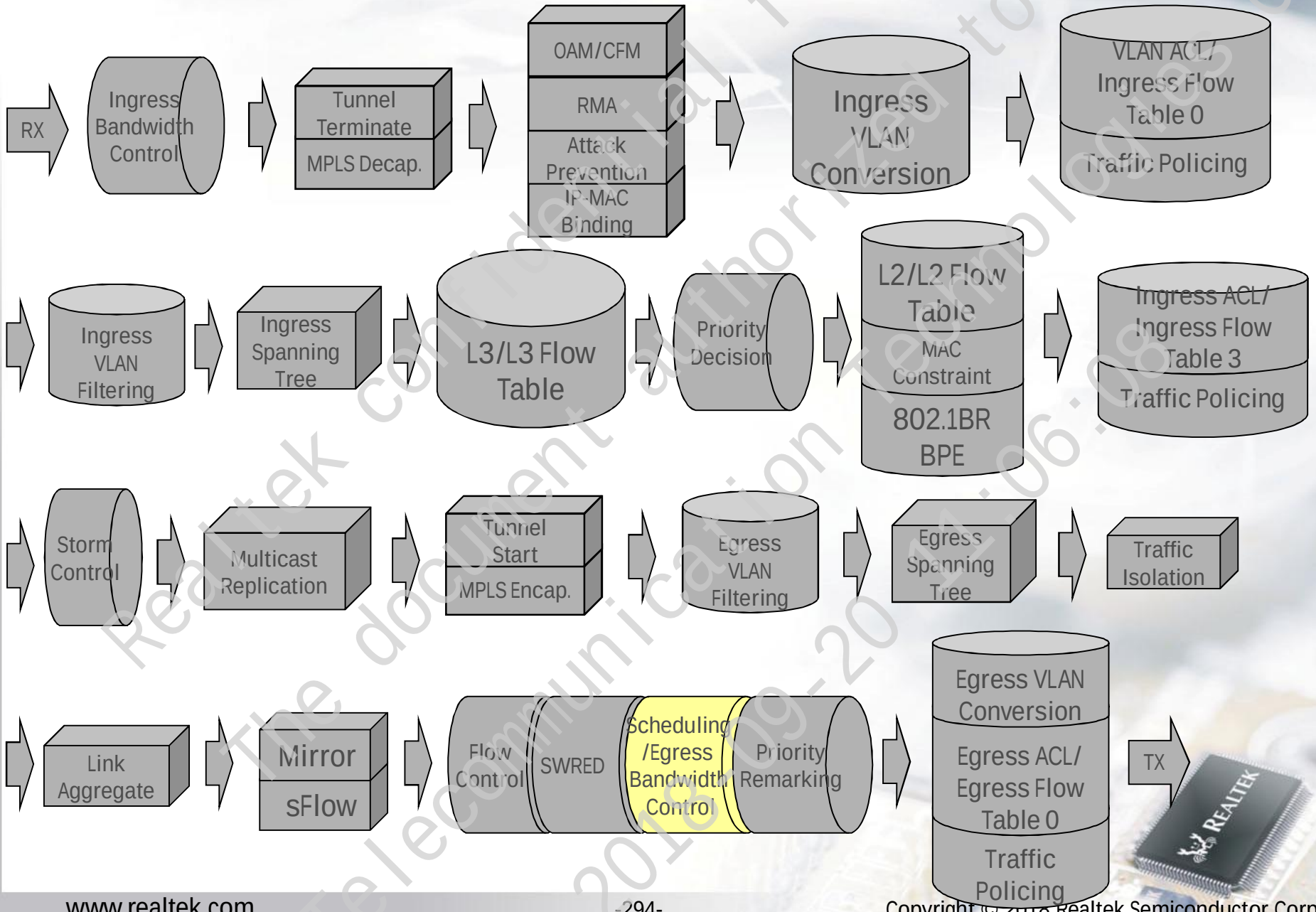
- DP Weight Table
  - Per Ingress Port configured
  - Take the MAX weight source as DP result
  - Each DP weight:
    - 1~3: DP weight, 3 is highest
    - 0: ignore the DP source
  - If all sources are ignored, DP will be 0
  - If weights of all sources are the same, DSCP > Outer > Inner > MPLS



REALTEK
Realtek Semiconductor Corp.

Scheduling

Packet Processing Pipeline



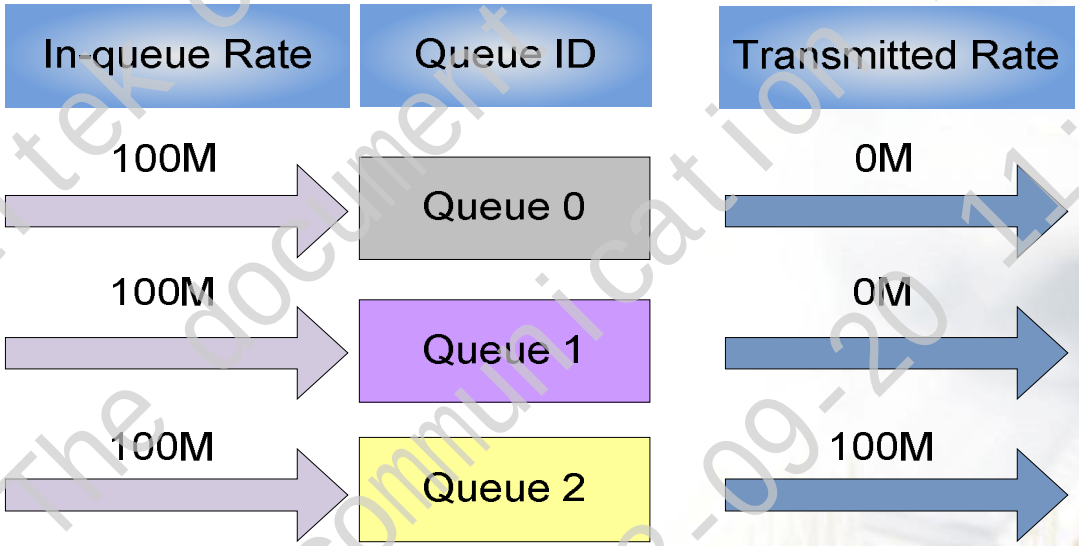
Scheduling Overview

- Scheduling Algorithms
  - Strict
  - Weighted Round Robin(WRR): Packet count
  - Weighted Fair Queue(WFQ): Byte count
- Queue Weight
  - Weight 0~127
  - Weight 0 is blocking
- Scheduling Max Queue Number
  - 8 queues for Normal Port
  - 12 queues for Stacking Port
  - 32 queues for CPU Port

Strict

- High queue is scheduled firstly when configure multiple strict queues

Configuration		
Port BW	Queue ID	Strict
100M	0	Yes
	1	Yes
	2	Yes

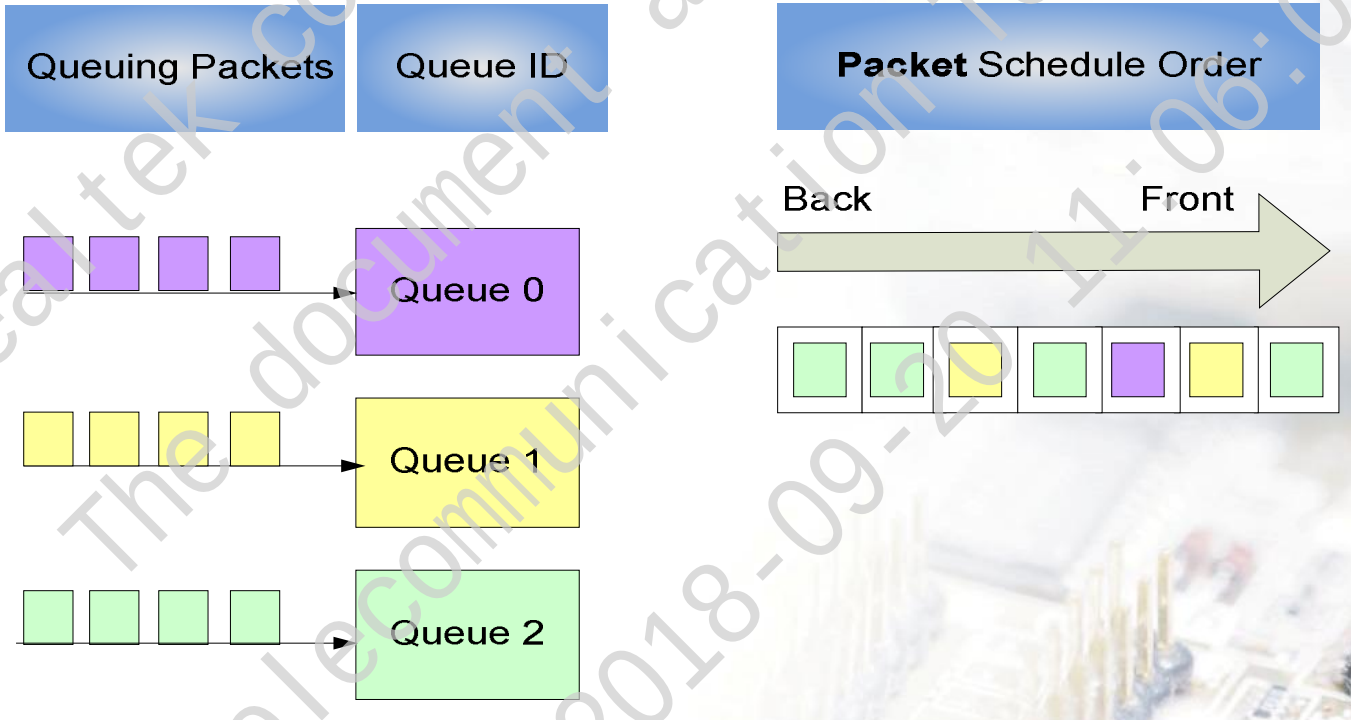


WRR

- Scheduled with Packet weight

Configuration	
Queue ID	Weight
0	1
1	2
2	4

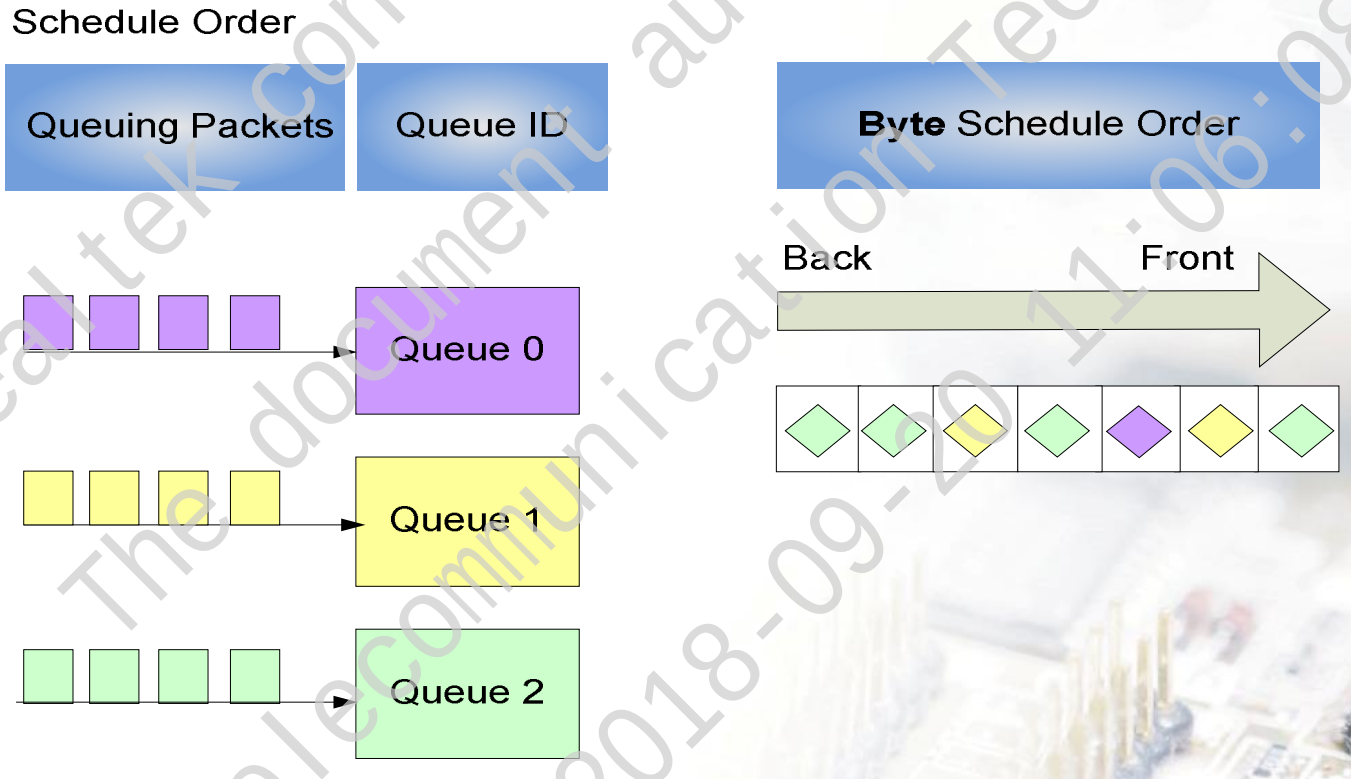
Schedule Order



WFQ

- Scheduled with Byte weight

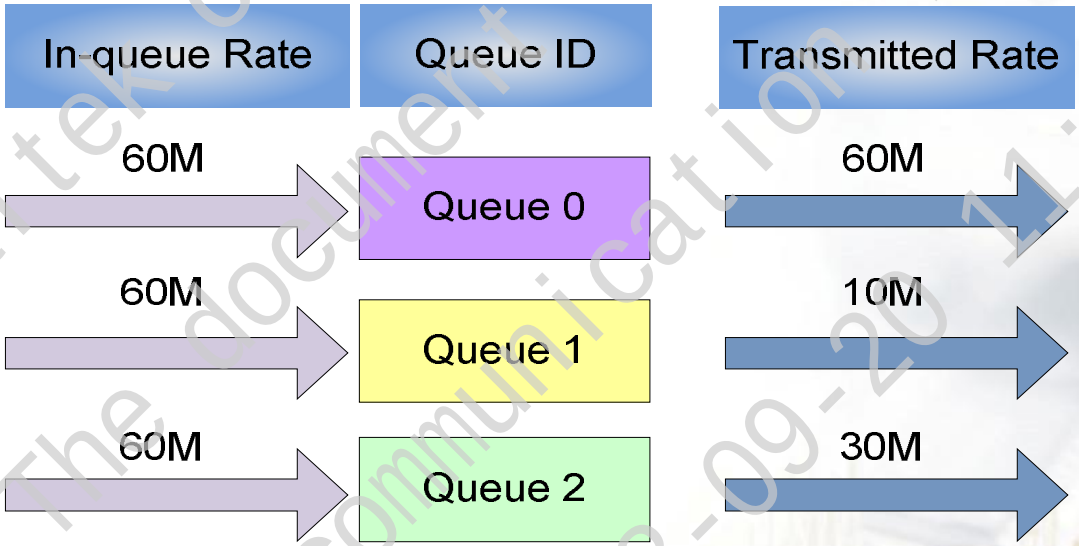
Configuration	
Queue ID	Weight
0	1
1	2
2	4



Strict + WFQ

- Strict queue is scheduled firstly , then WFQ/WRR queue

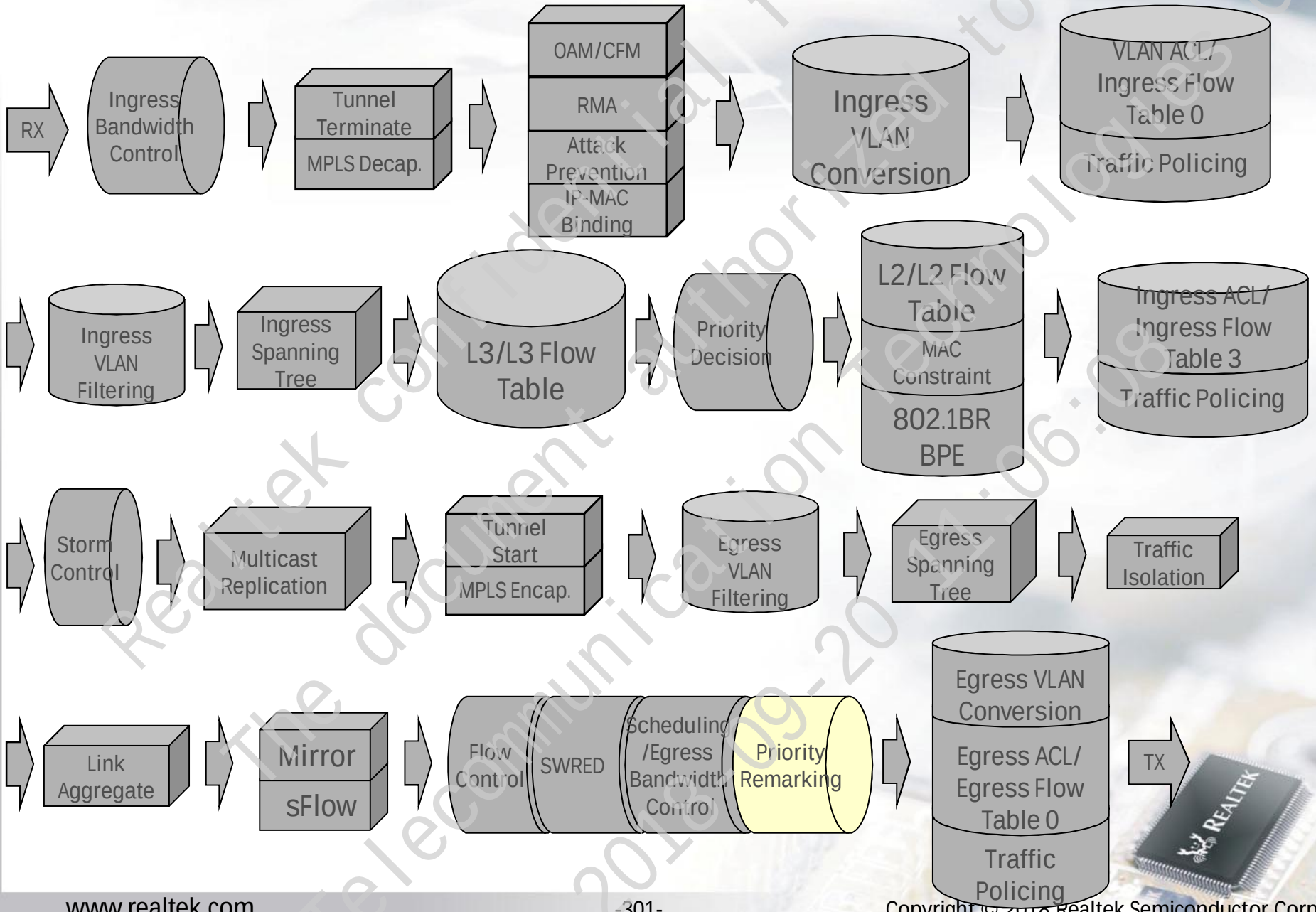
Configuration			
Port BW	Queue ID	Weight	Strict
100M	0	1	Yes
	1	1	N/A
	2	3	N/A



Realtek Semiconductor Corp.

Egress Remarking

Packet Processing Pipeline



Remarking Overview

- Per Egress Port Configuration
  - Inner 802.1p priority remarking
  - Outer 8021p priority remarking
  - DSCP remarking
  - DEI remarking

Outer-priority/Inner-priority Remarking (1/2)

- Global Independent Remarking Source
  - Internal Priority
  - Inner-priority
  - Outer-priority
  - DSCP
- Each source has a self remarking table
  - Example

Internal Priority to Inner-priority Remarking Table

Internal Priority	Remarking Outer-priority
0	0 ~ 7
1	0 ~ 7
2	0 ~ 7
...	...
7	0 ~ 7

DSCP to Outer-priority Remarking Table

DSCP	Remarking Outer-priority
0	0 ~ 7
1	0 ~ 7
2	0 ~ 7
...	...
63	0 ~ 7

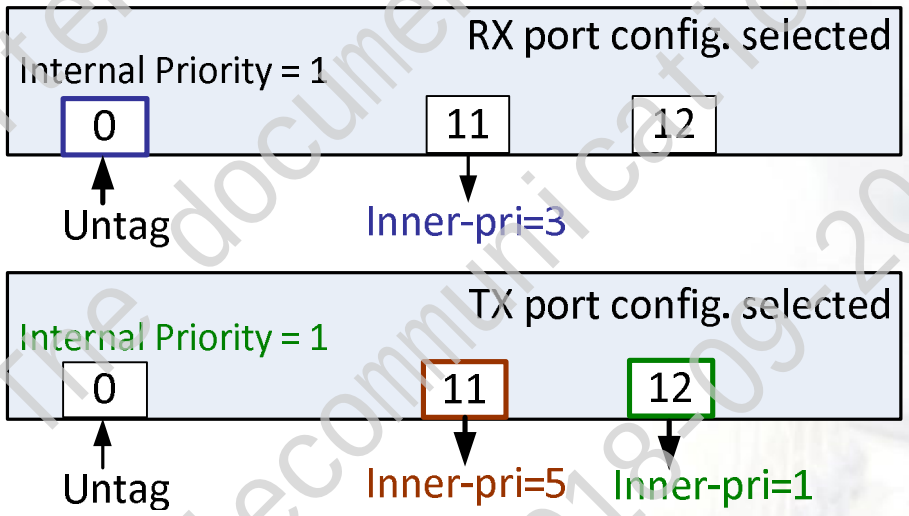
Outer-priority/Inner-priority Remarking (2/2)

Port Remarking State	Remarking Source	Incoming Packet	TX Inner-priority (Outer-priority)
Enable	Internal Priority	ANY	Internal Priority to Inner-priority (Outer-priority) Remarking Table
	Inner-priority	I-TAG	Inner-priority to Inner-priority (Outer-priority) Remarking Table
		I-UNTAG	Default Inner-priority (Outer-priority) Arbitration
	Outer-priority	O-TAG	Outer-priority to Inner-priority (Outer-priority) Remarking Table
		O-UNTAG	Default Inner-priority (Outer-priority) Arbitration
	DSCP	IP	DSCP to Inner-priority (Outer-priority) Remarking Table
		Non-IP	Default Inner-priority (Outer-priority) Arbitration
Disable	ANY	I-TAG	Original Inner-priority (Outer-priority)
		I-UNTAG	Default Inner-priority (Outer-priority) Arbitration

Default Inner Priority Arbitration

- Global selects RX or TX port's configuration
- Port-based Default Inner-priority Source
  - Copy from Internal-priority
  - Per-port default inner-priority

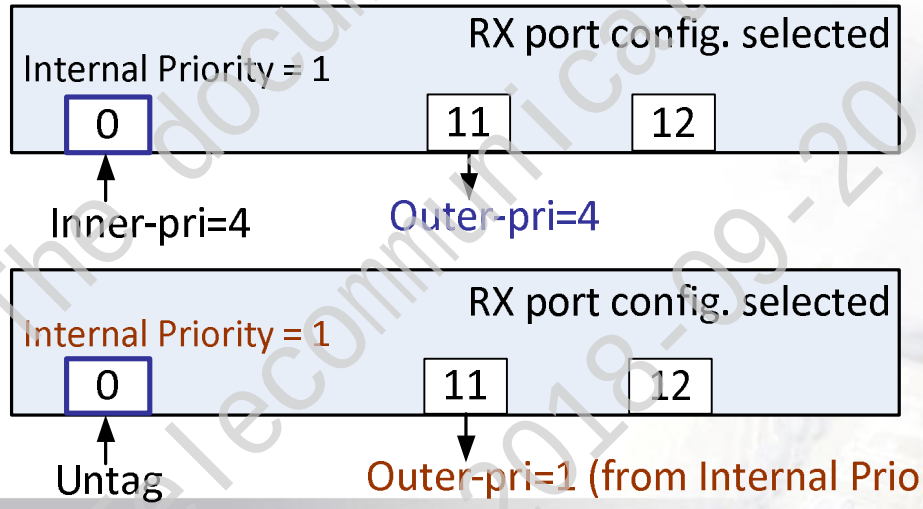
	Default Inner-priority Source	Default Inner-priority
Port 0	Port default Inner-priority	3
Port 11	Port default Inner-priority	5
Port 12	Copy Internal Priority	7



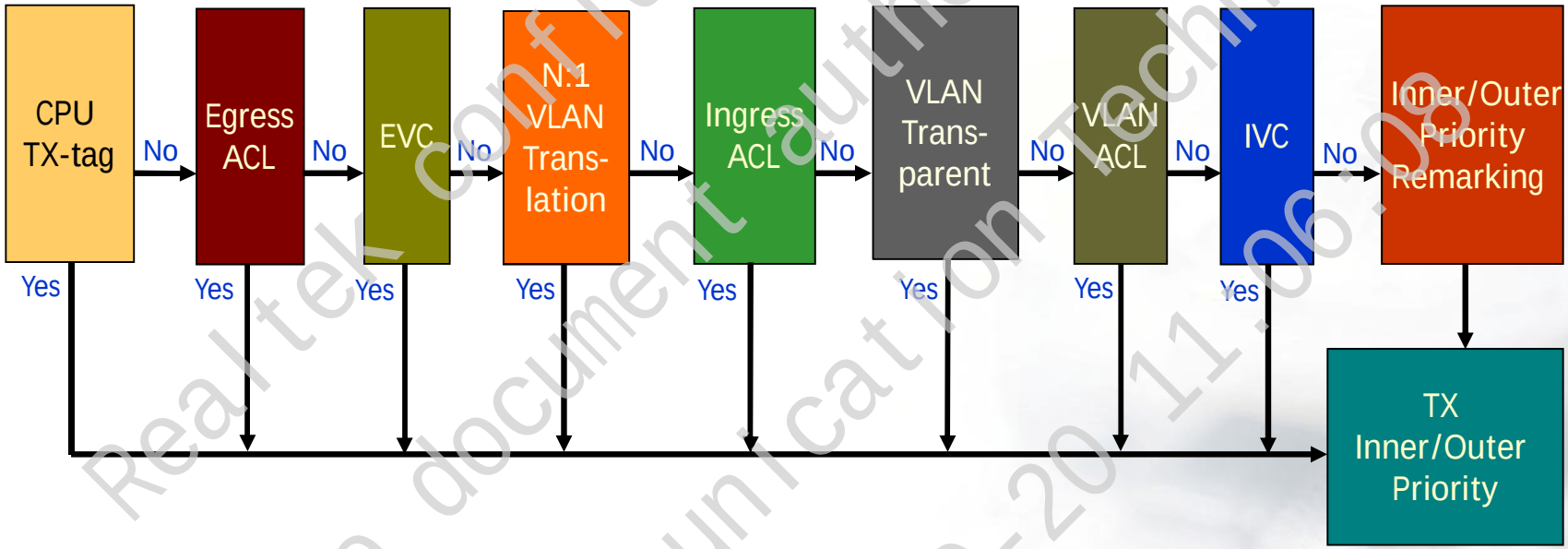
Default Outer Priority Arbitration

- Global selects RX or TX port's configuration
- Port-based Default Outer-priority Source
  - Copy from Internal-priority
  - Per-port default outer-priority
  - Copy from Inner-priority
    - Take Internal Priority if inner-priority is not available

	Default Outer-priority Source	Default Outer-priority
Port 0	Copy from Inner-priority	3
Port 11	Copy from Inner-priority	5
Port 12	Copy Internal Priority	7



TX Inner/Outer Priority Precedence



DSCP Remarking

- Global DSCP Remarking Source
  - Internal Priority
  - Inner-tag Priority
  - Outer-tag Priority
  - DSCP
  - Drop precedence(DP): According to RFC 2475 “DiffServ”
  - DP and Internal Priority

- Each source excluding DP has a self remarking table

DSCP Remarking Tables

Internal/Inner/Outer Priority to DSCP Remarking Table

Internal/Inner/Outer Priority	Remarking DSCP
0	0 ~ 63
1	0 ~ 63
...	...
7	0 ~ 63

DSCP to DSCP Remarking Table

DSCP	Remarking DSCP
0	0 ~ 63
1	0 ~ 63
...	...
63	0 ~ 63

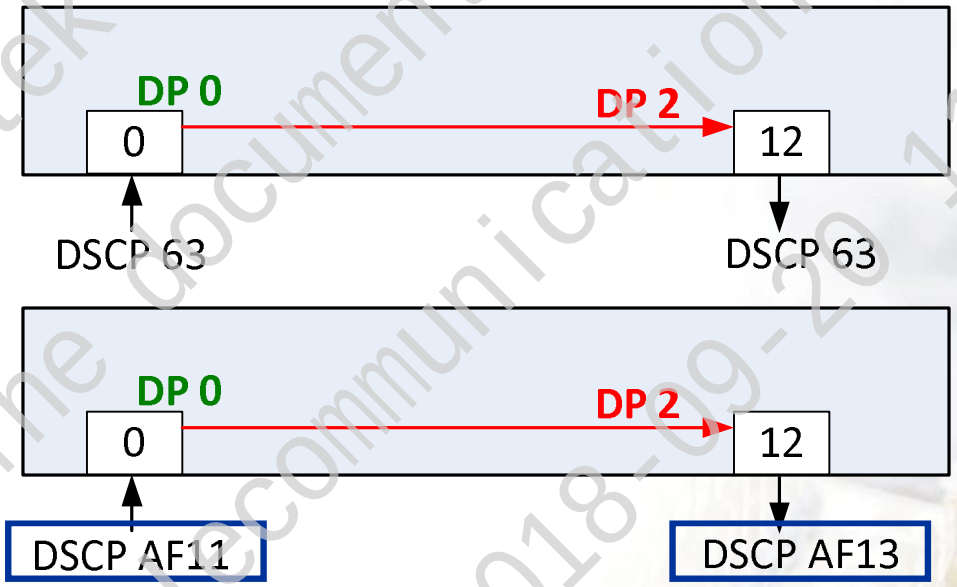
DP and Internal Priority to DSCP Remarking Table

DP	Internal Priority	Remarking DSCP
0	0	0 ~ 63
	1	0 ~ 63
	...	...
	7	0 ~ 63
1	0	0 ~ 63
	1	0 ~ 63
	...	...
	7	0 ~ 63
2	0	0 ~ 63
	1	0 ~ 63
	...	...
	7	0 ~ 63

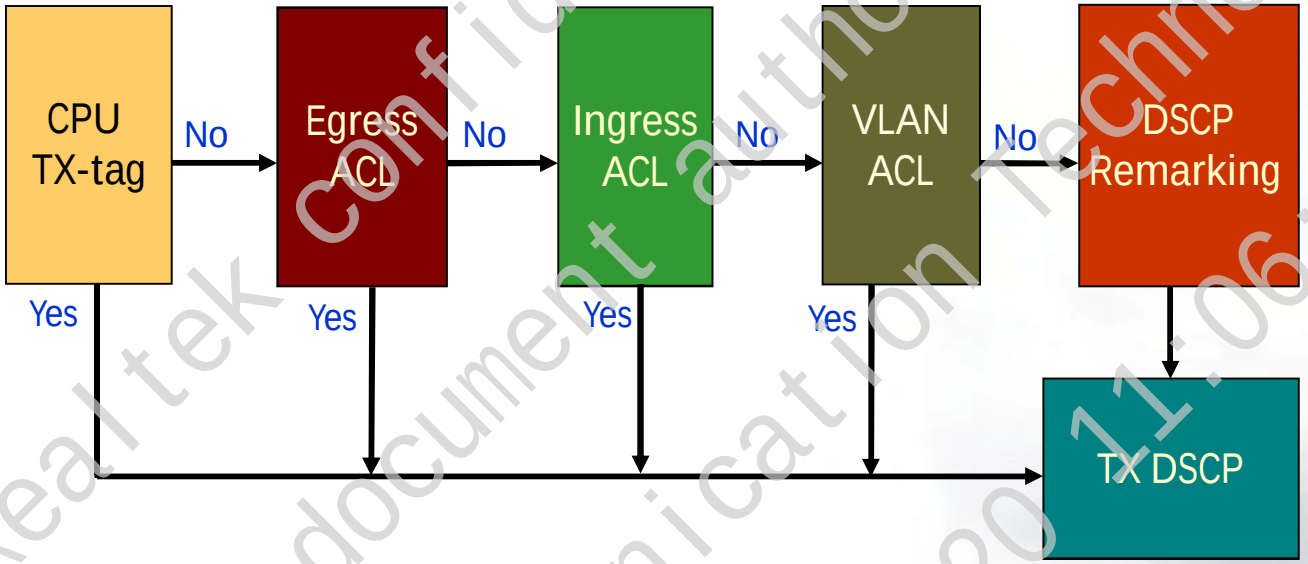
DSCP Remarking – DP Remarking Source

- According to RFC 2475 “DiffServ”

Assured Forwarding (AF) Behavior Group				
DP	Class 1	Class 2	Class 3	Class 4
DP 0 (Green)	AF11(001010)	AF21(010010)	AF31(011010)	AF41(100010)
DP 1 (Yellow)	AF12(001100)	AF22(010100)	AF32(011100)	AF42(100100)
DP 2 (Red)	AF13(001110)	AF23(010110)	AF33(011110)	AF43(100110)



DSCP Remarking Precedence



DEI Remarking

- Per egress port selects DEI remarking target
  - Inner Tag
  - Outer Tag
- Global DEI Remarking Source
  - DP
  - Internal Priority

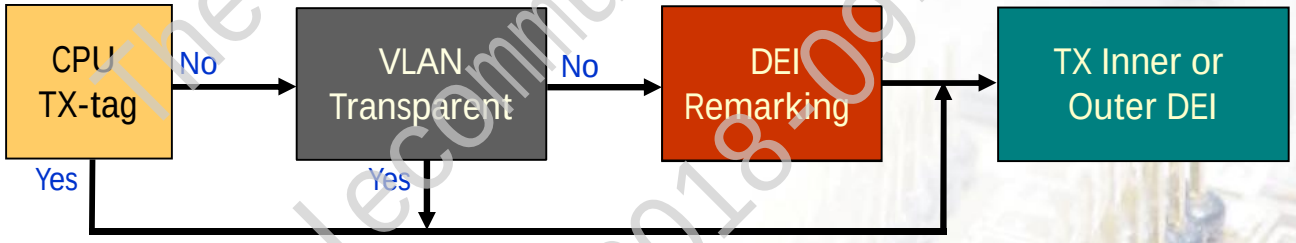
DP to DEI Remarking Table

DP	Remarking DEI
0	0、1
1	0、1
2	0、1

Internal Priority to DEI Remarking Table

Internal Priority	Remarking DEI
0	0、1
1	0、1
...	...
7	0、1

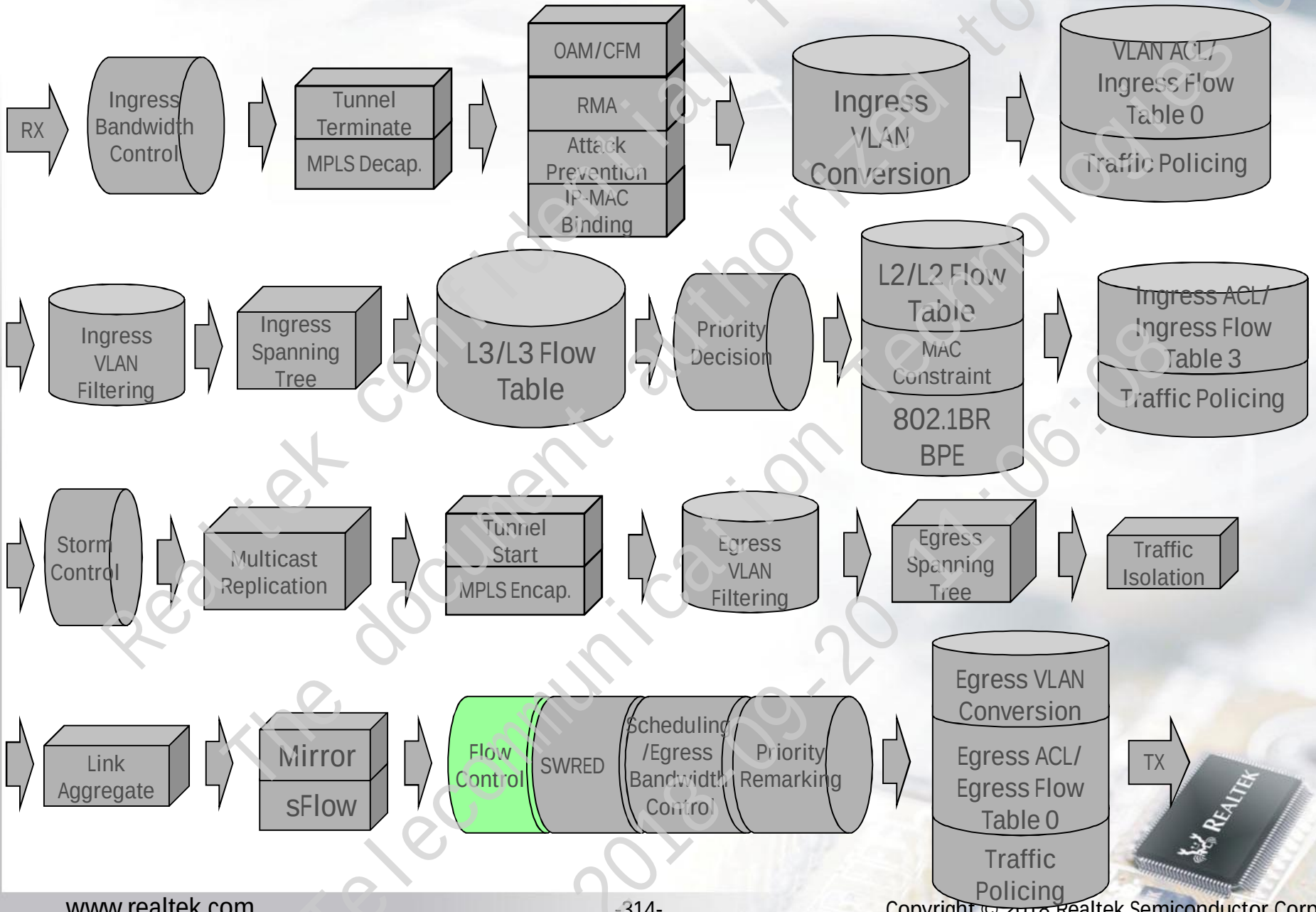
- TX DEI Precedence



REALTEK
Realtek Semiconductor Corp.

Flow Control

Packet Processing Pipeline

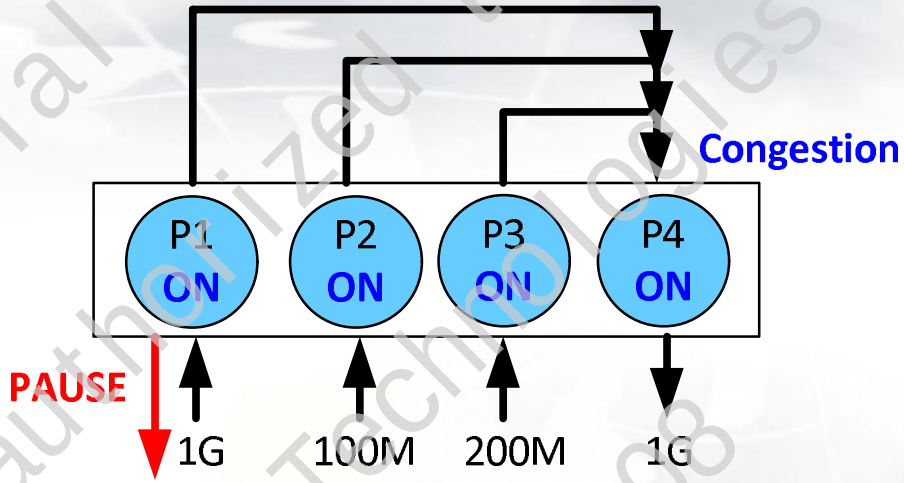


Flow Control Overview (1/2)

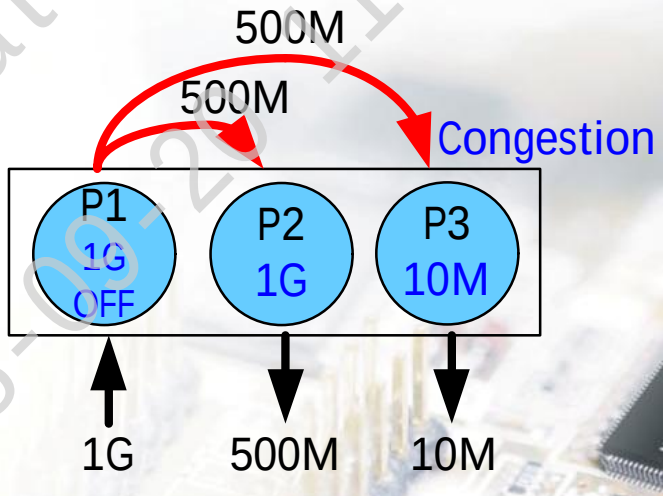
- Packet Buffer
  - 16Mbits
  - Unit: page (1 page = 360 bytes)
  - Totally 5825 pages
- Ingress Flow Control
- Packet Drop
  - System Drop
  - Egress Drop
- Flooding Traffic HOL Prevention
- Jumbo Frame Support

Flow Control Overview (2/2)

- Ingress Flow Control
  - Example: Many-to-1 Congestion
    - Only P1 sends Pause Frame



- Egress Queue Drop
  - Example: High-speed port to 2 low-speed ports
    - Only P1 → P3 traffic dropped



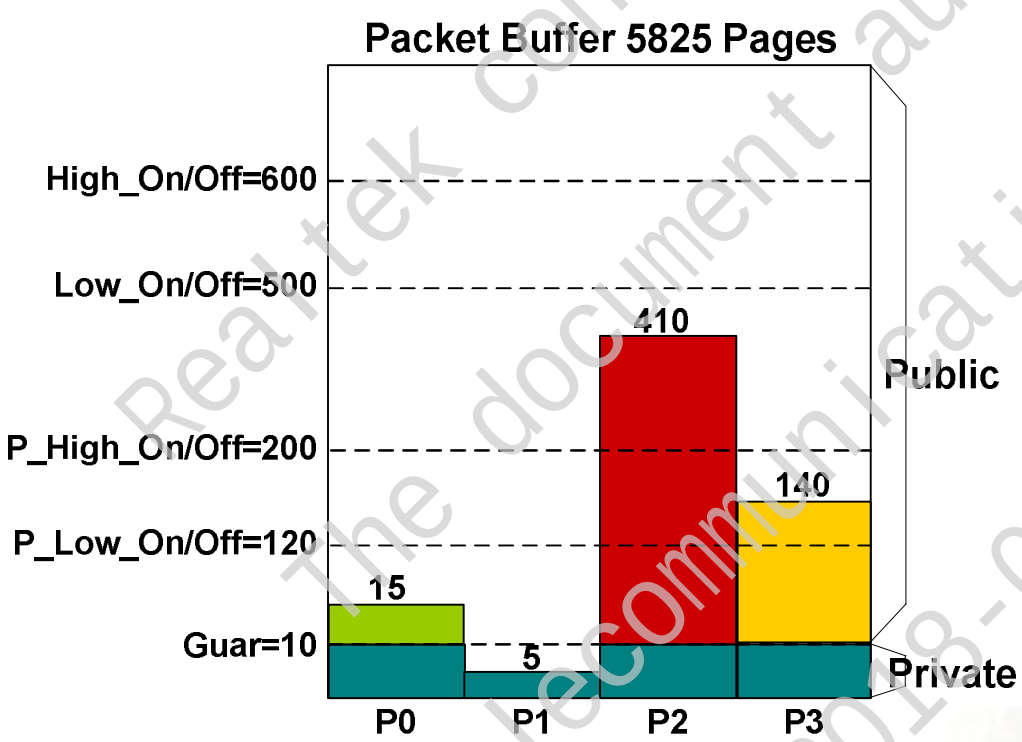
Ingress Flow Control

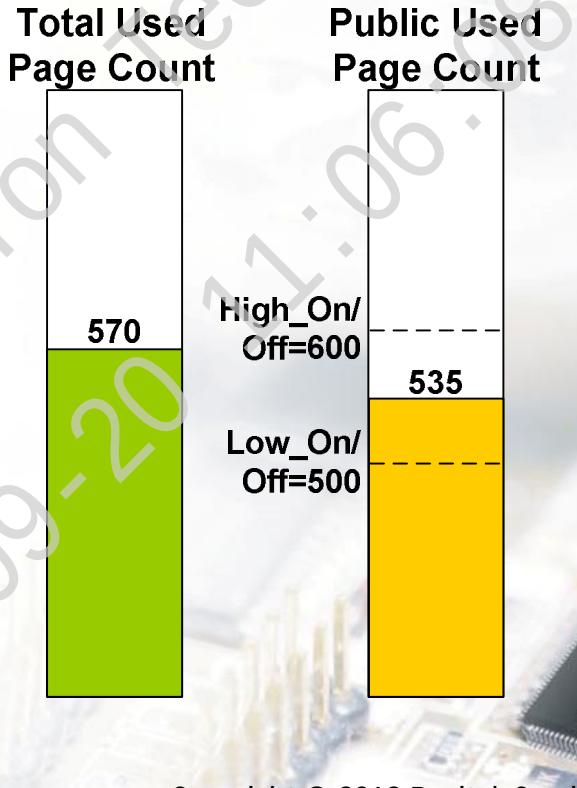
Port:

- Per-port Guarantee(Guar)
- Per Port Used Page Count (PU)

System:

- Total Used Page Count
- Public Used Page Count (PbU = PU – Guar)



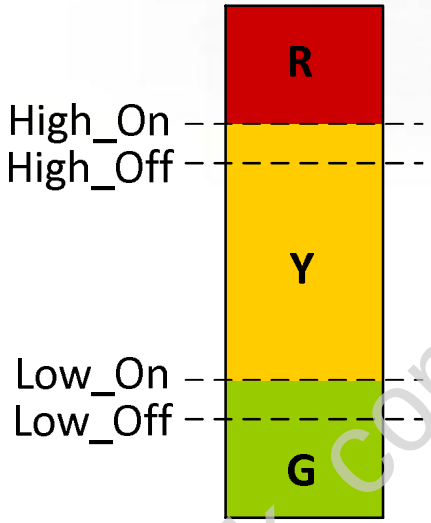


Flow Control Threshold

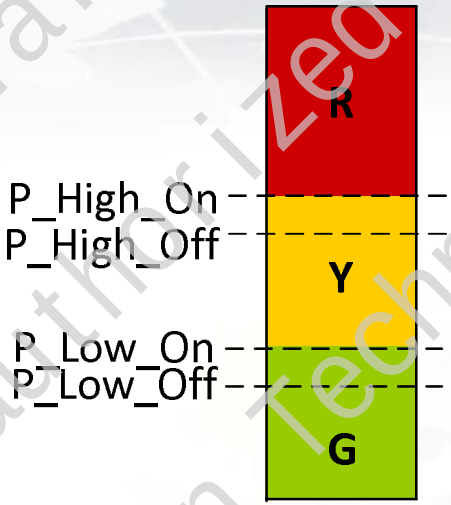
- Global Threshold
  - Flow Control ON
    - High_On / High_Off / Low_On / Low_Off
  - Flow Control OFF
    - High_On / High_Off / Low_On / Low_Off
  - System Drop
    - Drop_All
- Port-based Threshold
  - 4-set Threshold Groups
    - Flow Control ON Threshold
      - P_High_On / P_High_Off / P_Low_On / P_Low_Off
    - Flow Control OFF Threshold of each set
      - P_High_On / P_High_Off / P_Low_On / P_Low_Off
    - Guarantee Threshold of each set
      - Guar
  - Per port index to 1 of 4-set
- Flow Control OFF Thresholds are used in Egress Queue Drop

Congest State (1/2)

Public Used Page Count



Per-Port Used Page Count

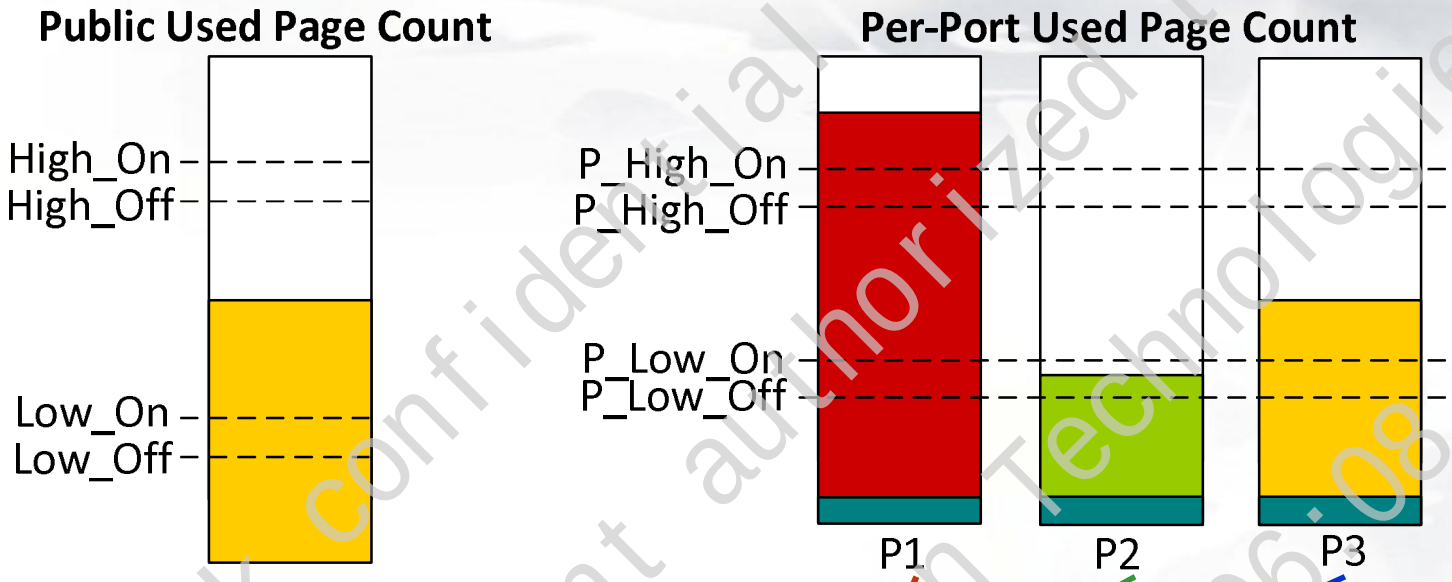


port / public	G	Y	R
G	X	X	X
Y	X	X	C
R	X	C	C

port = per port used page count

public = public used page count

Congest State (2/2)



public \ port	G	Y	R
G	X	X	X
Y	X	X	C
R	X	C	C

P1 is congested!

port = per port used page count

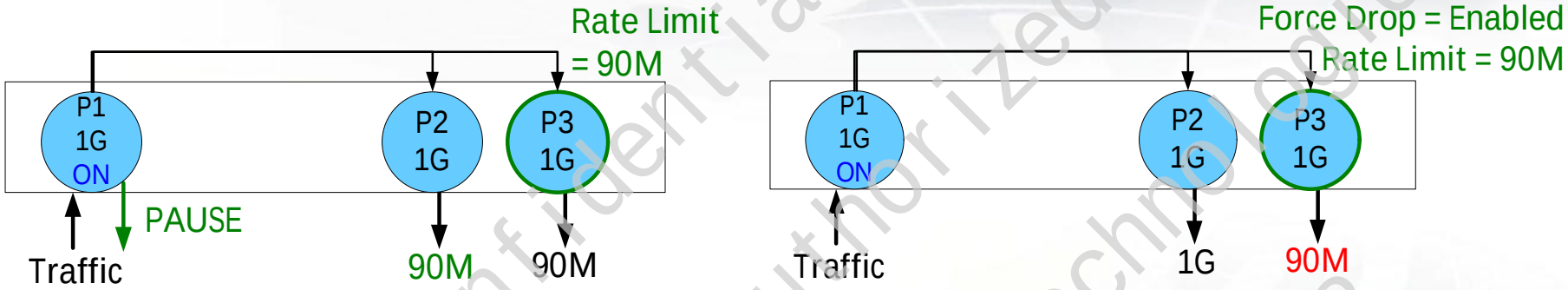
public = public used page count

Packet Drop

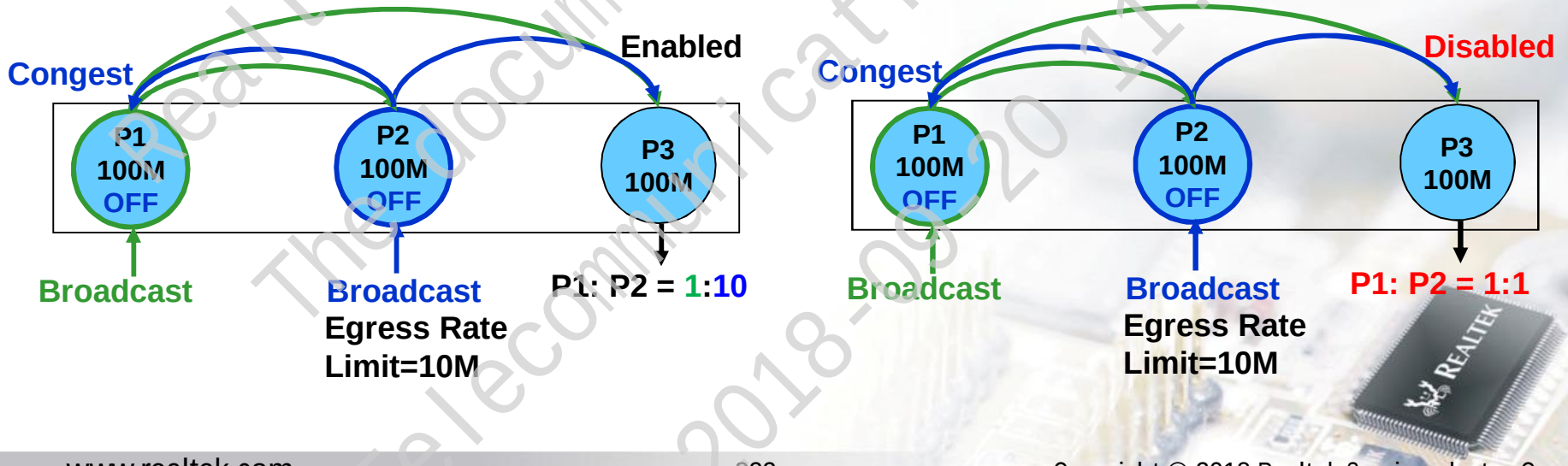
- System Drop Condition
  - Total used page count > Drop_All
- Egress Drop MUST meet all below conditions
  - Source port flow control OFF
    - per-egress-queue can bypass this condition
  - Source port in congestion state
    - Egress port can bypass this condition
  - Egress queue used page count > queue-based drop threshold
- Queue-based Threshold
  - 3-set Threshold Groups
    - Each Queue Drop_On / Drop_Off
  - Per egress port index to 1 of 3-set

Egress Drop Application

- Egress Force Drop

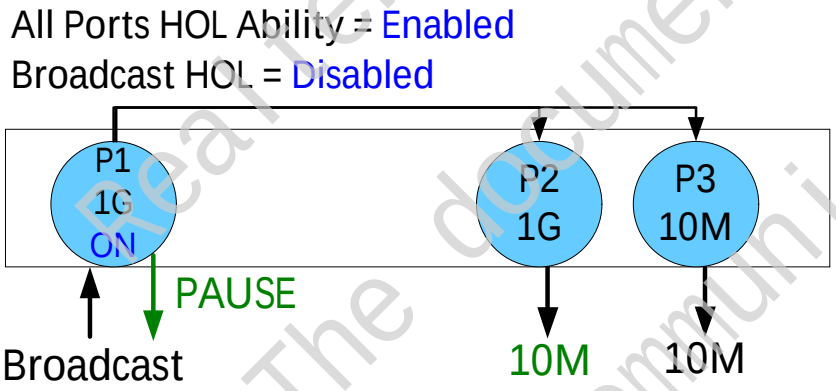


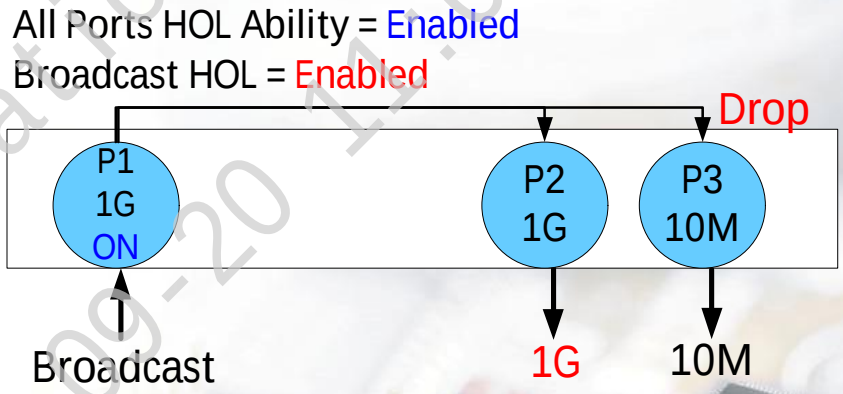
- Reference Source Port Congestion



Flooding Traffic HOL Prevention

- Global Configuration for Four Packet Types
  - Broadcast
  - L2 Multicast
  - IP Multicast
  - Unknown Unicast
- Ingress Port Ability Configuration
- Example





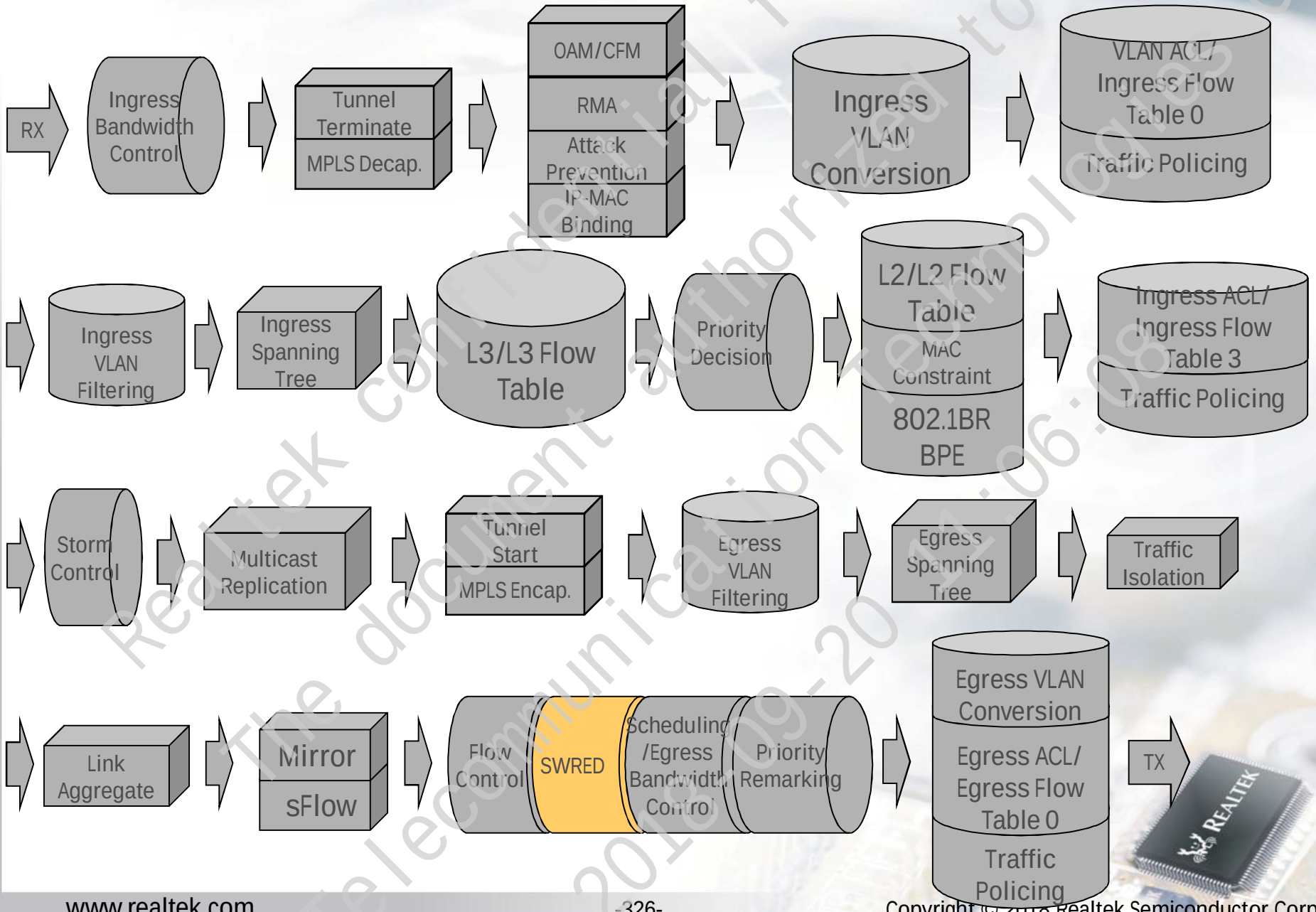
Jumbo Frame Support

- Global Configuration
  - Enabled/Disabled jumbo frame mechanism
  - Jumbo Frame Length for detecting jumbo frame
- Global Individual Threshold
  - Flow Control ON
    - High_On / High_Off / Low_On / Low_Off
  - Flow Control OFF
    - High_On / High_Off / Low_On / Low_Off
- Port-based Threshold
  - Use Group 4
- Egress Queue-based Drop Threshold
  - Use Group 3

Realtek Semiconductor Corp.

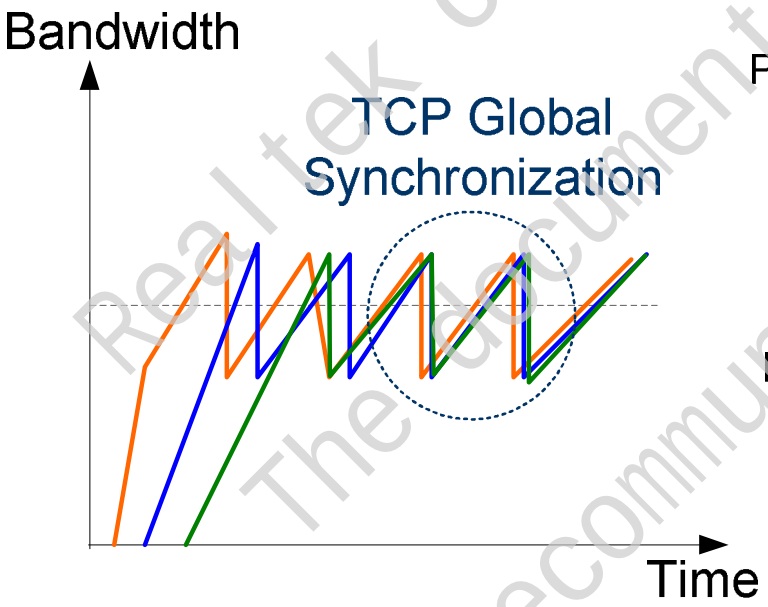
SWRED

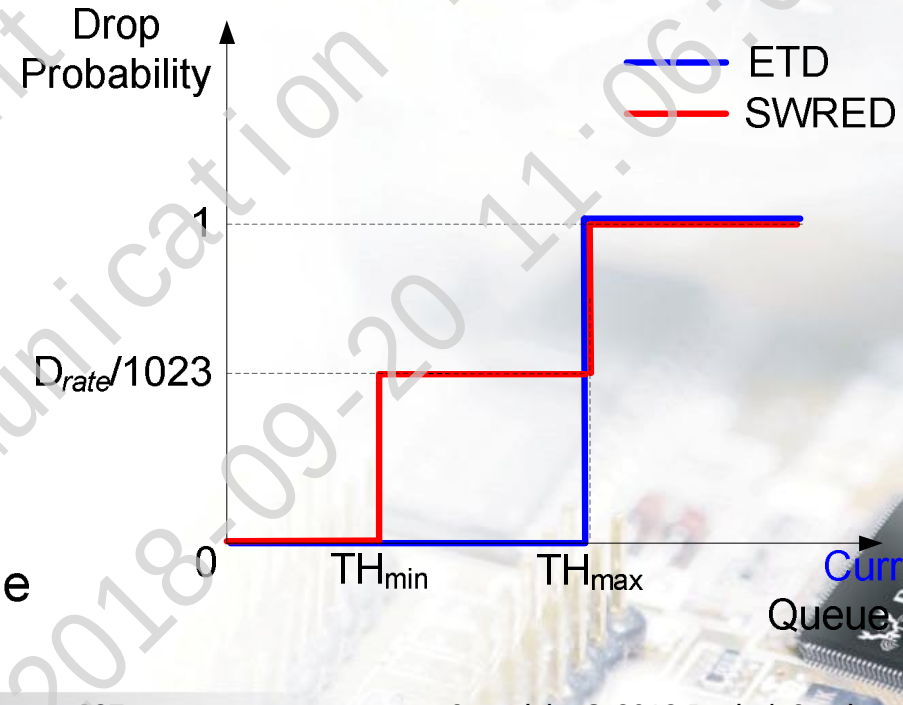
Packet Processing Pipeline



Congestion Avoidance Overview

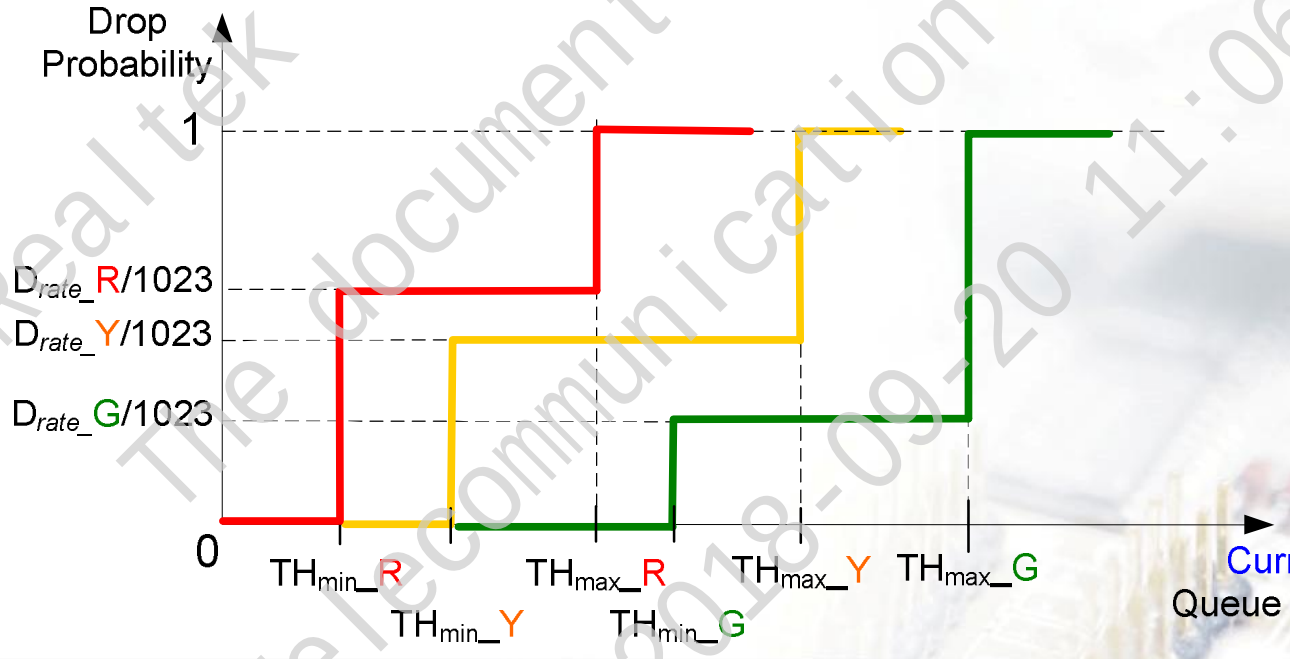
- 2 Congestion Avoidance Algorithms
  - Egress Tail Drop (ETD)
  - Simple Weighted Random Early Detection (SWRED)
- Per egress port specify congestion avoidance algorithm
- SWRED may be used to resolve TCP global synchronization



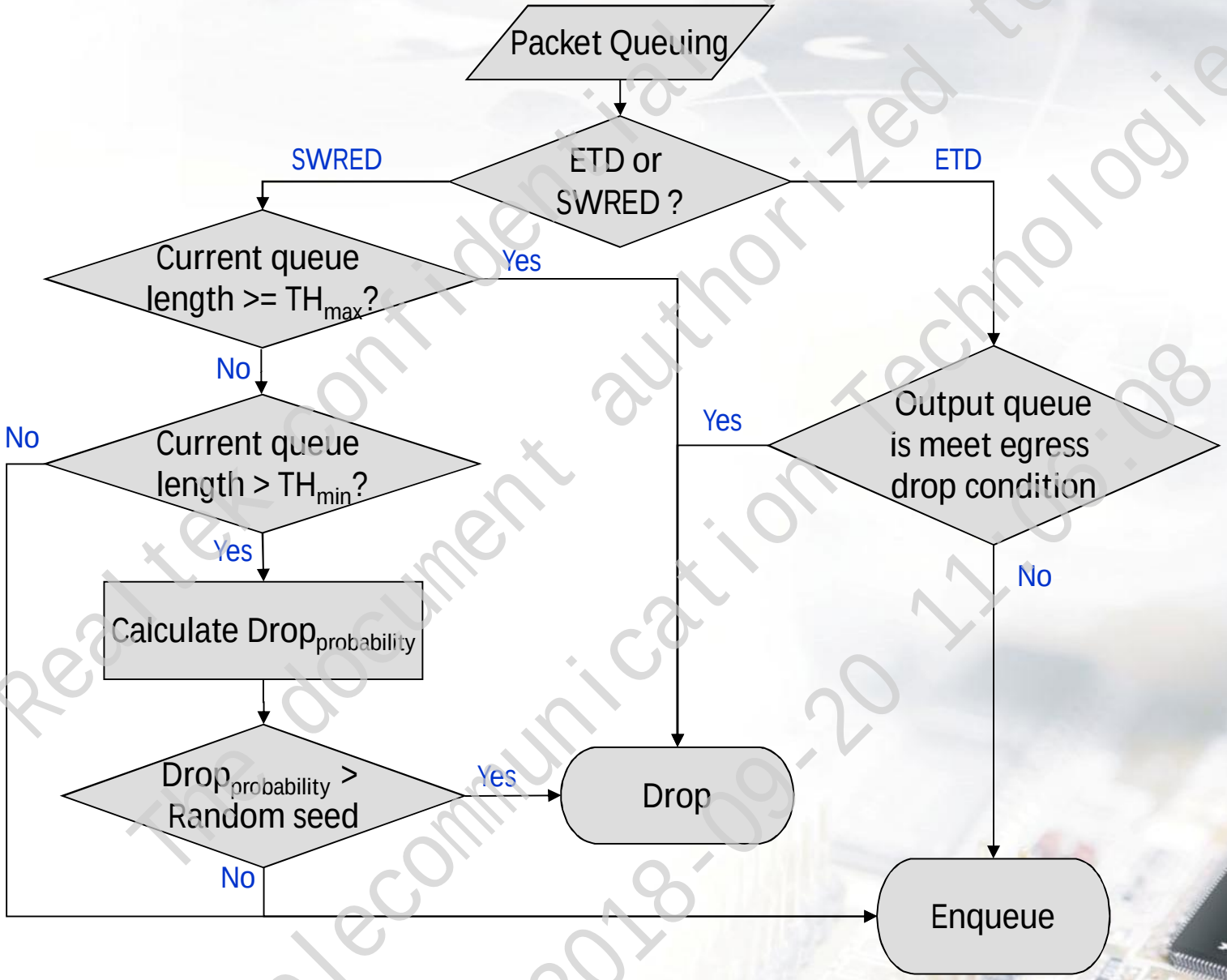


SWRED

- Queue-based Drop Mechanism
- Per Drop Precedence (Color) Per Queue has below configuration
  - Maximum Drop Threshold(TH_{max})
  - Minimum Drop Threshold(TH_{min})
  - Drop Rate(D_{rate}): 0 ~ 255
- Drop Probability ($D_{probability}$) = D_{rate}/1023



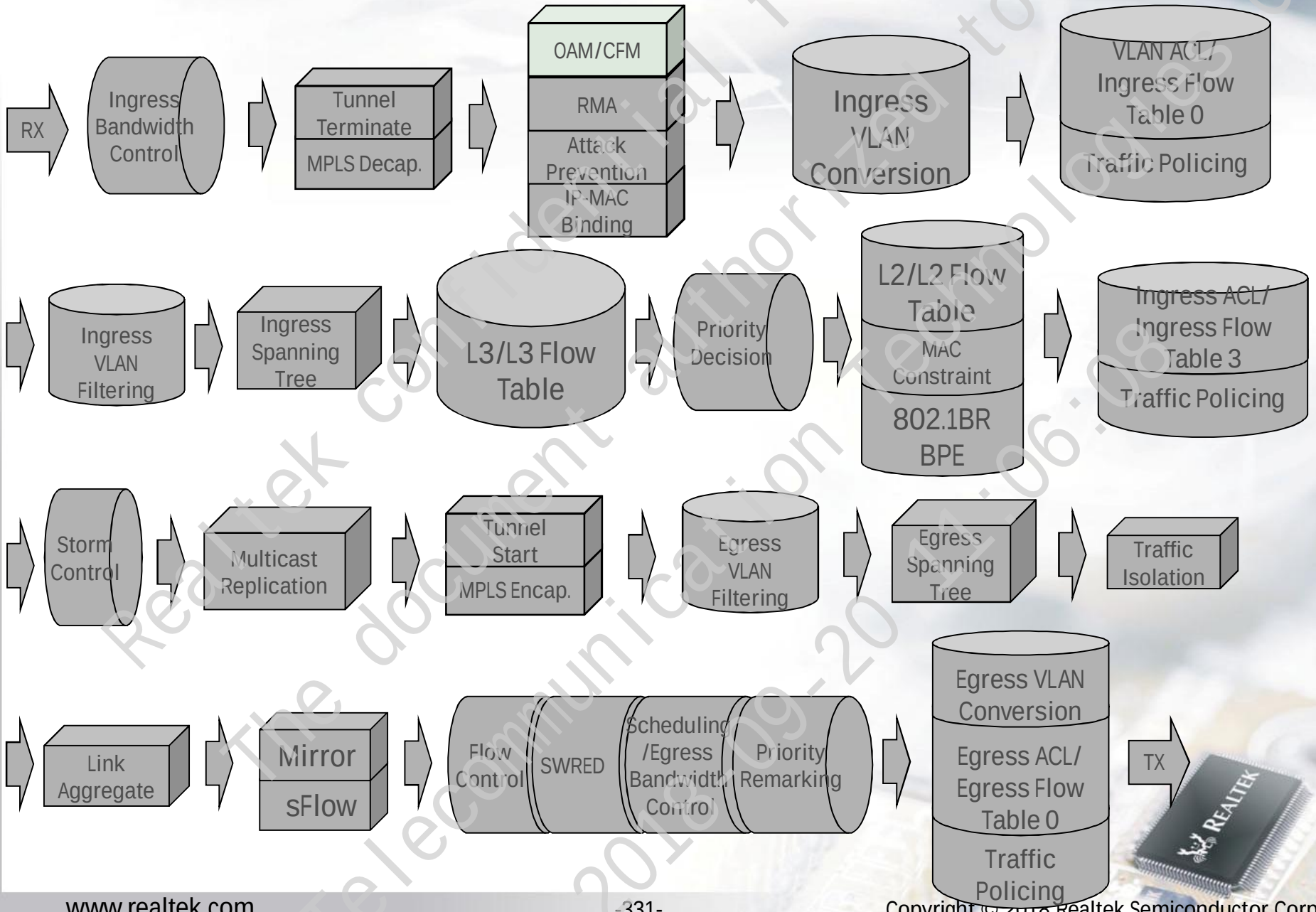
Congestion Avoidance Flow



Realtek Semiconductor Corp.

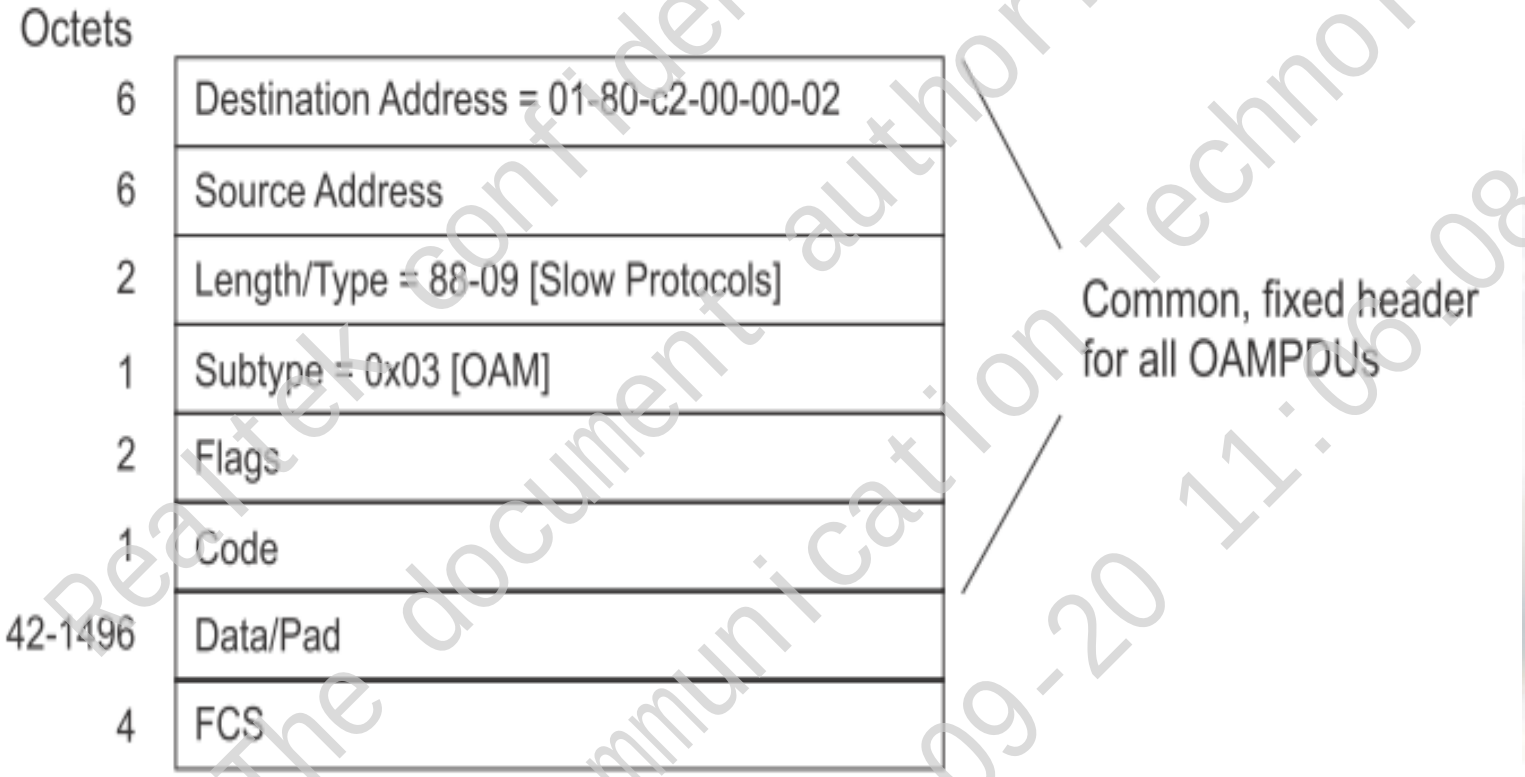
OAM

Packet Processing Pipeline

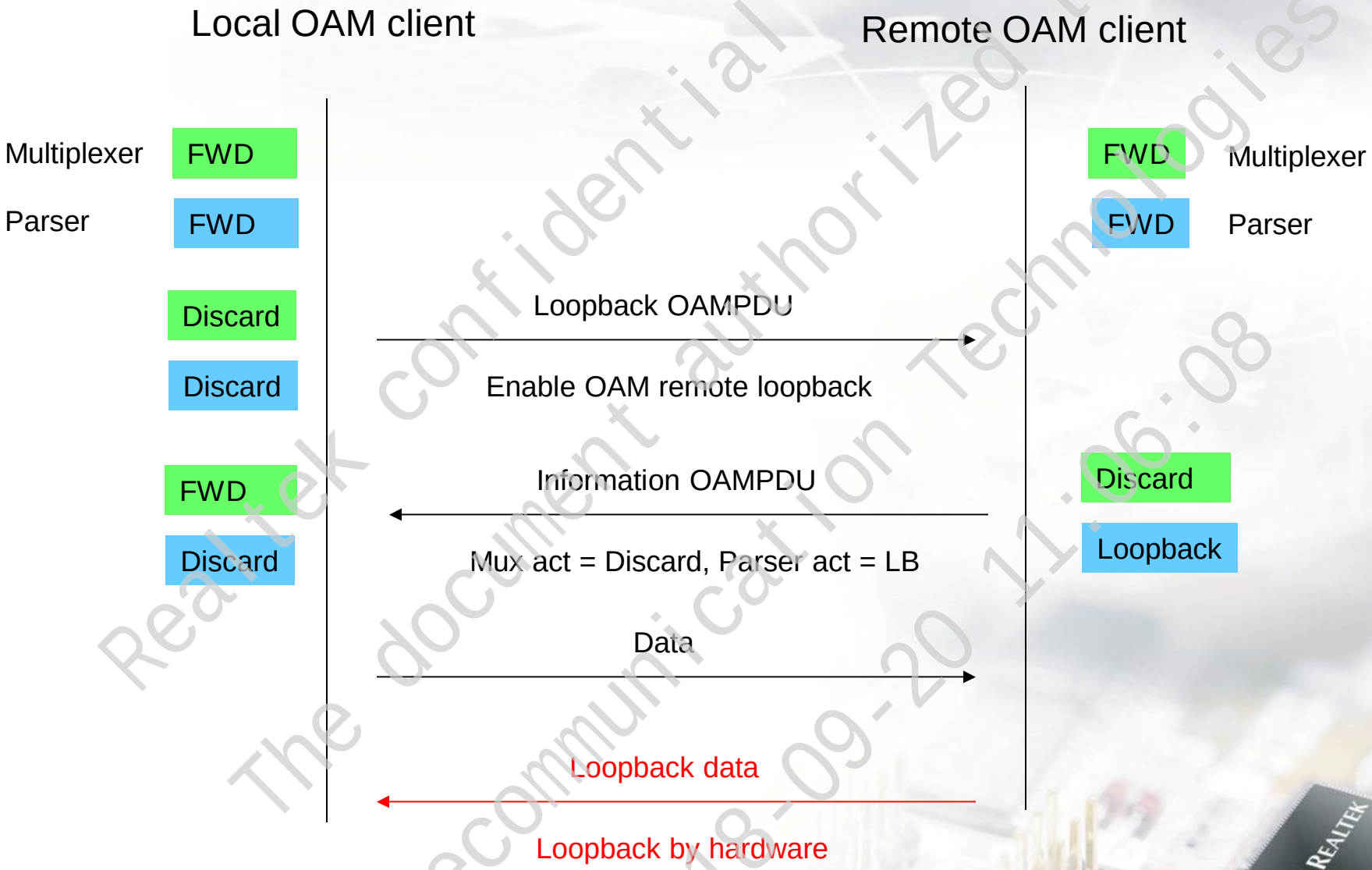


OAM Overview

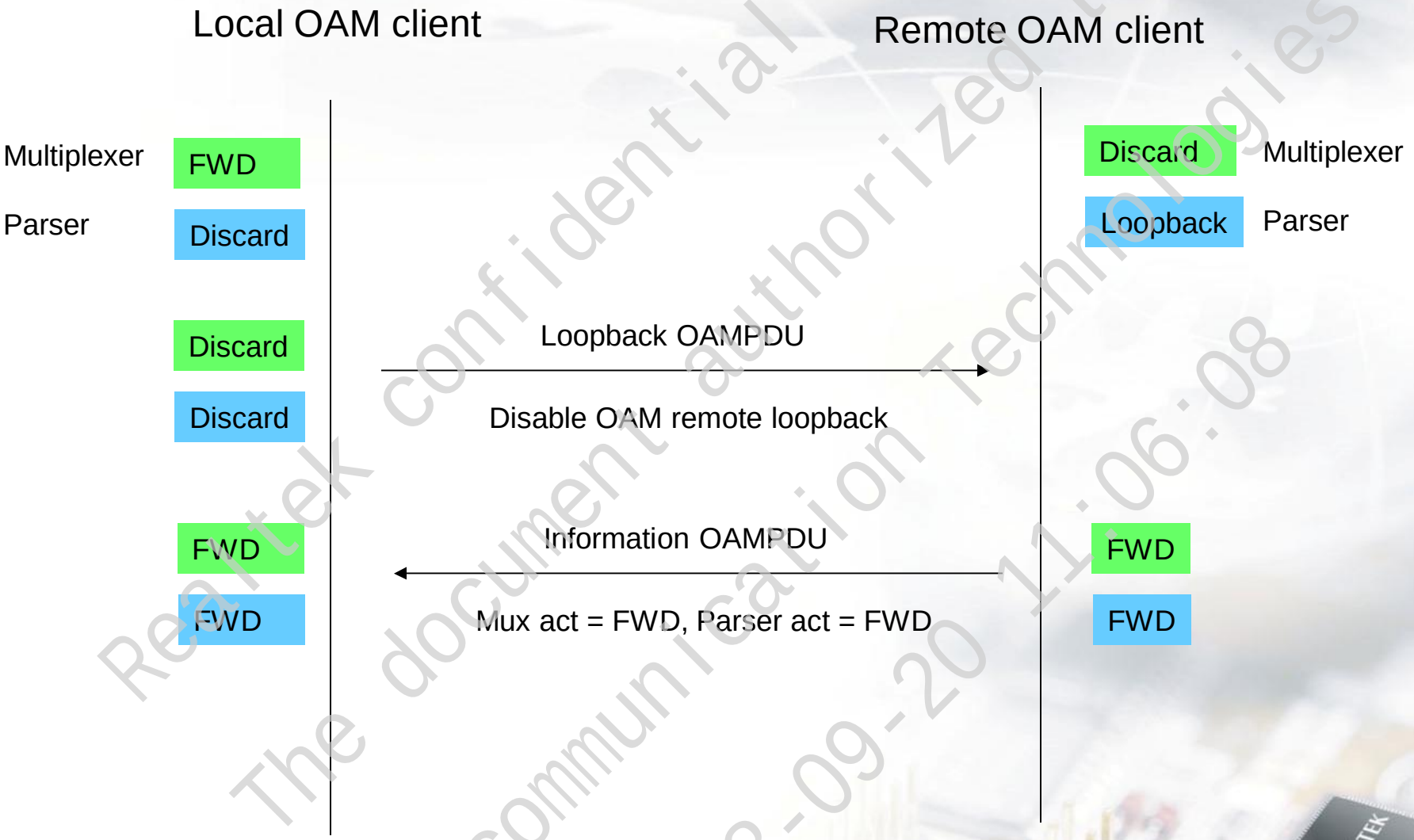
- OAM provides mechanism useful for point-to-point monitoring link operation such as remote loopback control. It can quickly determine the location of failing links or fault conditions



OAM – Initial loopback



OAM – Exiting loopback



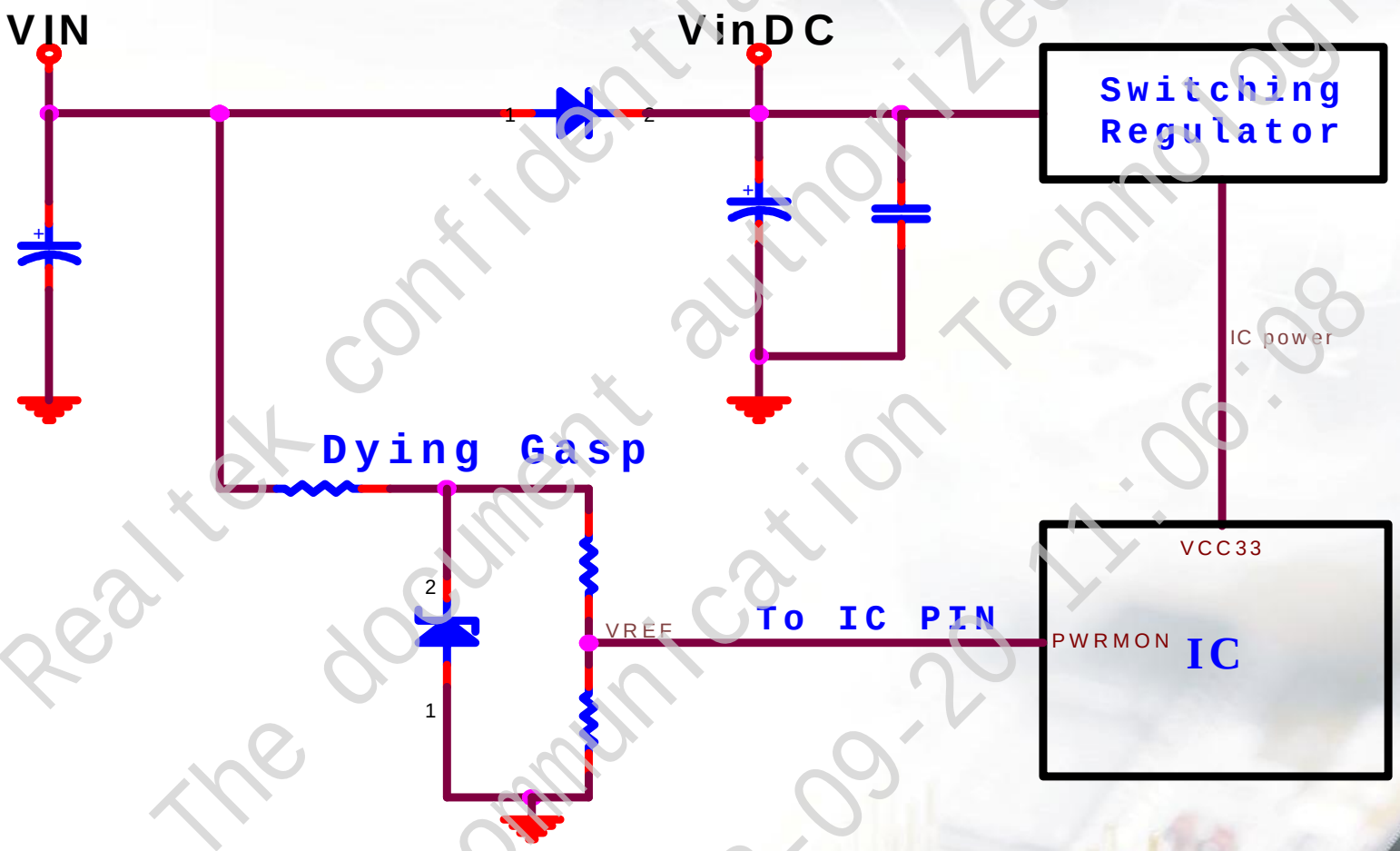
OAM Configuration

- Global configuration
  - OAM PDU action
    - Discard
    - Forward
    - Trap
  - Swap MAC addresses for loopback packet
  - OAMPDU SA learn control
- Per port configuration
  - Parser (Rx) action
    - Discard
    - Forward
    - Loopback
  - Multiplexer (Tx) action
    - Discard
    - Forward

OAM Dying Gasp

- When power failure, send specific packet to link partner.
- Voltage < 1.2V is dying gasp state, 1.55V ~ 3.3V is normal state.
- Global configuration
  - Period of voltage low
    - 40ns granularity
    - From 40ns ~ 2.6214ms
  - Send packet count (0 ~ 7)
- Per port configuration
  - Enable/Disable dying gasp
- Dying gasp packet is automatic send by hardware. The full packet content is filled by software in advance. Which is per port configurable.

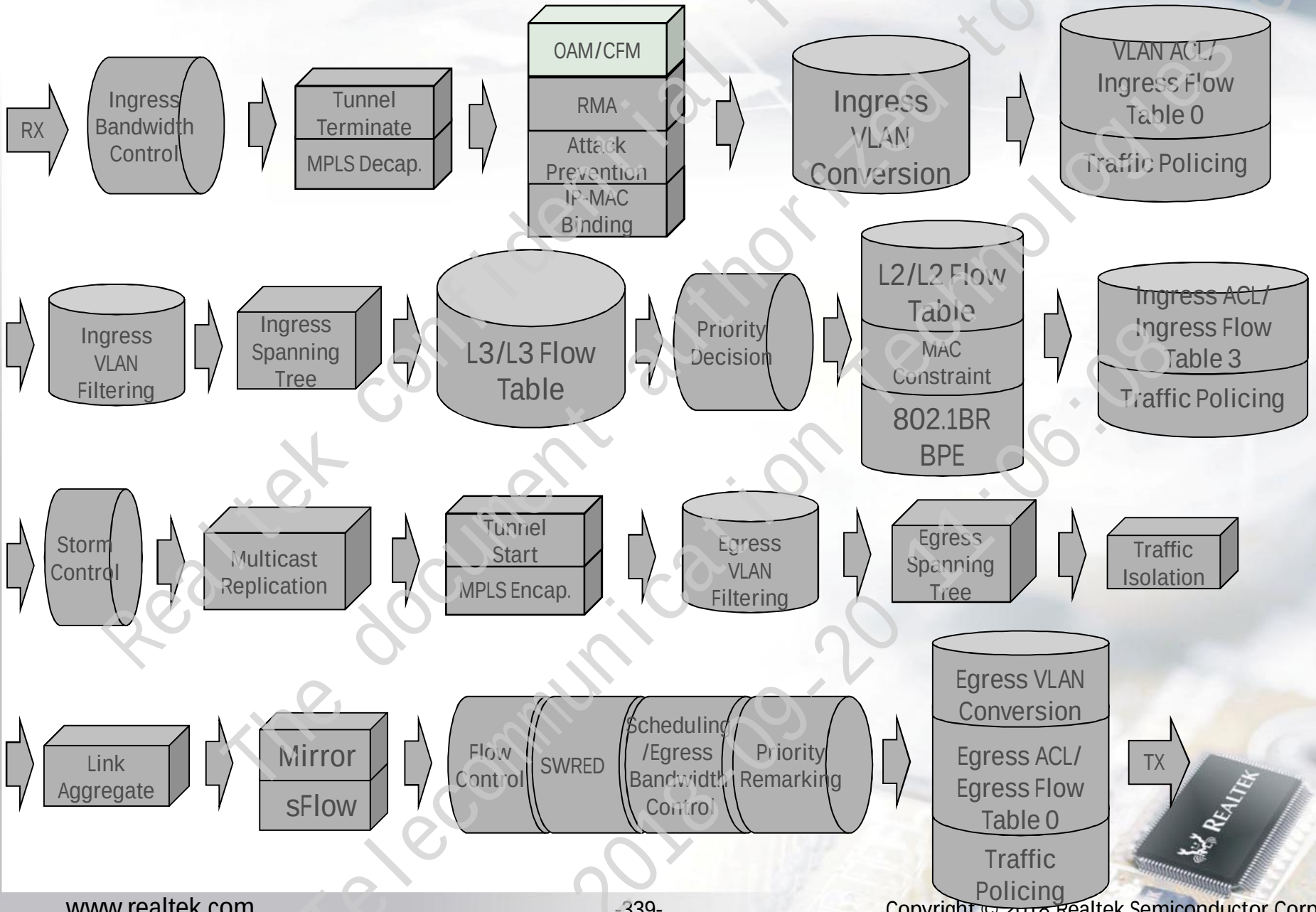
OAM Dying Gasp



Realtek Semiconductor Corp.

802.1ag CFM

Packet Processing Pipeline

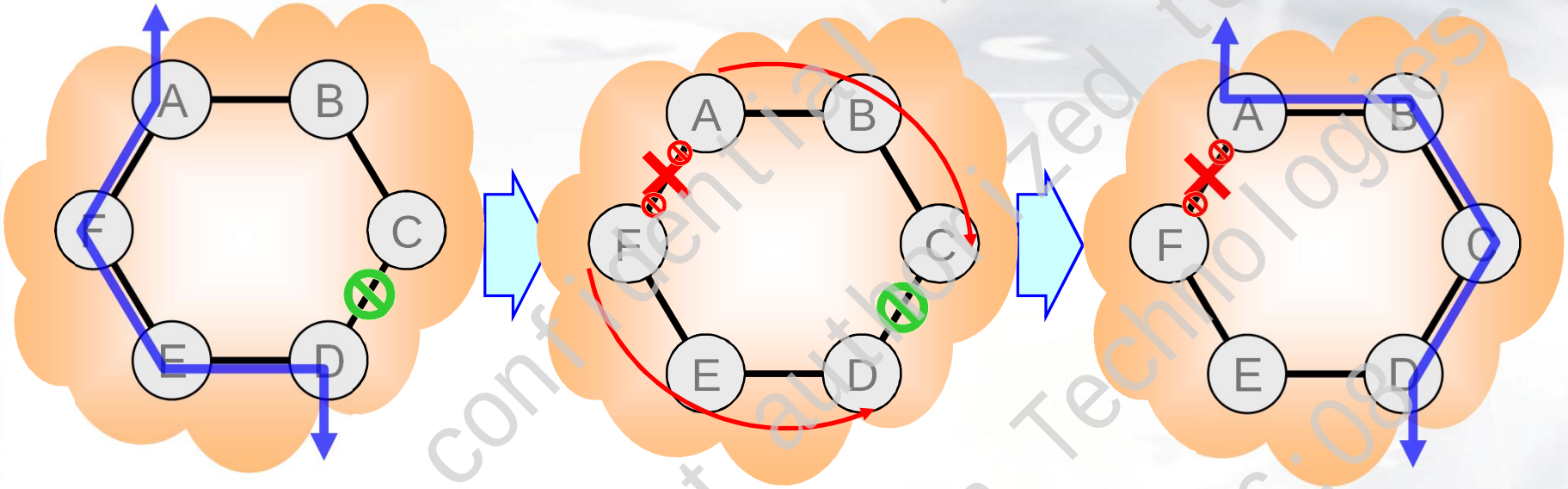


CFM

- CFM provides end-to-end connectivity fault management
- Global configuration
  - Unknown CFM packet action (exclude continuity check, loopback and link trace)
    - Forward
    - Drop
    - Trap
- Per MD (Maintenance Domain) level configuration

Action / Message	Forward	Drop	Trap	LFD (Link Failure Detect)
Loopback	✔	✔	✔	✕
Link Trace	✔	✔	✔	✕
Continuity Check	✔	✔	✔	✔

CFM (G.8032)

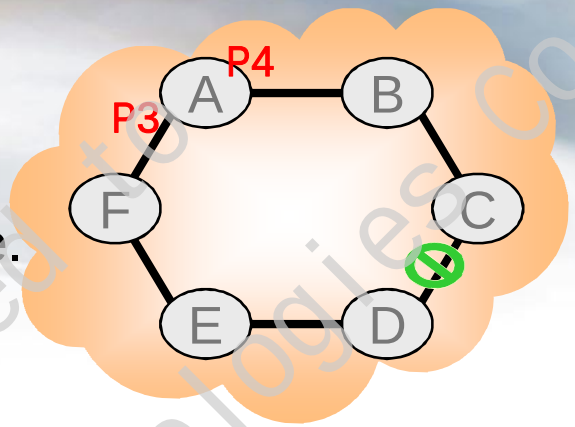


- Protection switching on Ethernet layer
- Preventing any loops by block mechanism
- VLAN-based protection switching
- Protection and recovery switch within 50 ms
- An Ethernet ring may support multiple traffic channels that may be grouped into different sets of VLANs
- Each ERP instance is independent of other ERP instances.
- Multiple ERP instances can be configured on the same ring.

CCM

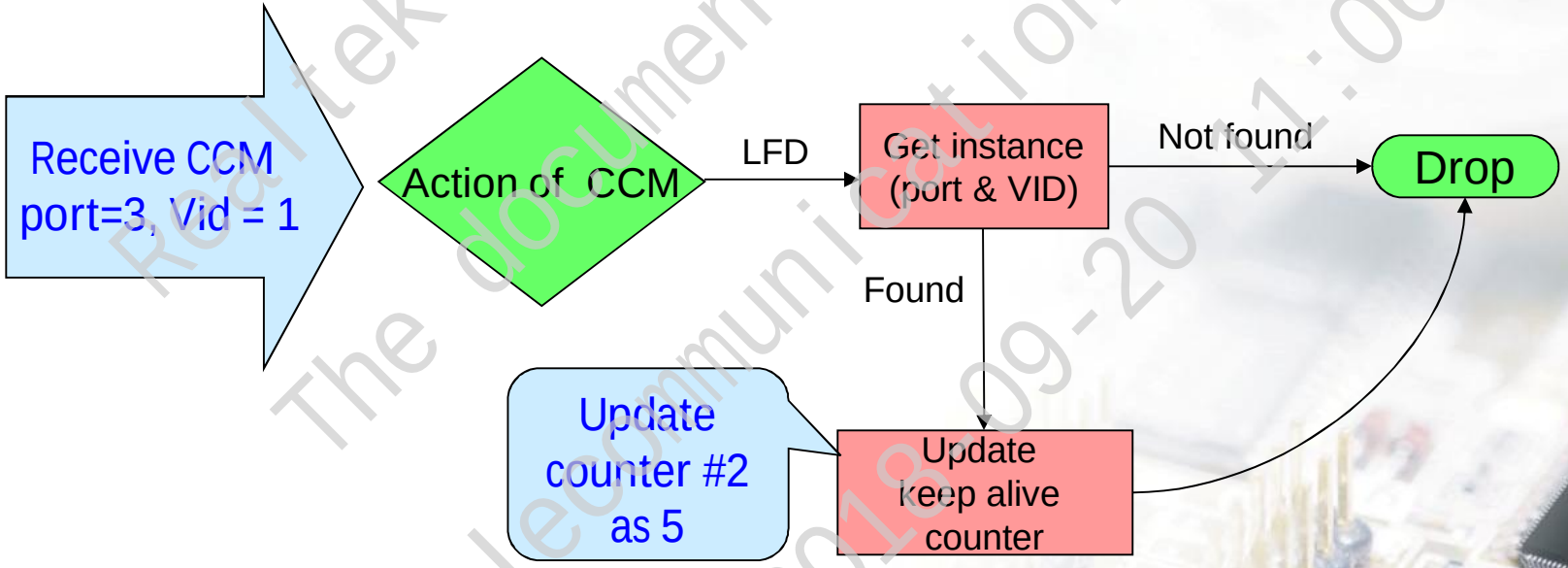
- Use CCM LFD action for link failure detection of G.8032
- 8 instance
  - 2 member ports
  - VID
  - Keep alive timer (0 ~ 1023 msec)
- CCM Rx
  - the device updates the keep alive counter
- CCM Tx
  - sent by hardware (configurable by software)

CCM RX (1/2)



- The keep alive counter is maintained by hardware.
- The other fields are configured by software.

VID	keep alive timer	Port #1	Keep alive counter #1	Port #2	Keep alive counter #2
1	**5**	4	**2**	**3**	3 → 5

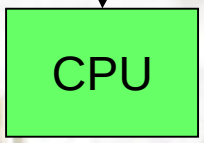


CCM RX (1/2)

- When keep alive countdown to 0, then ASIC signals an interrupt.

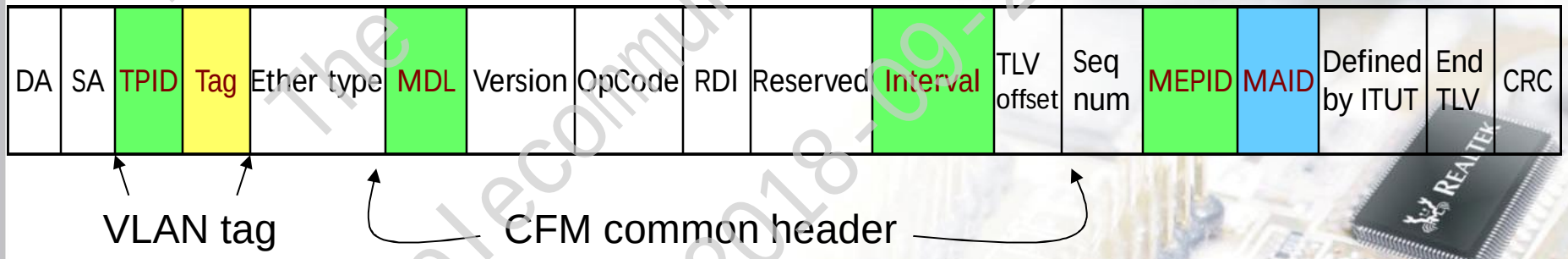
VID	keep alive timer	Port #1	Keep alive counter #1	Port #2	Keep alive counter #2
1	5	4	8	3	0

Signal an interrupt



CCM TX

- Global configuration for CCM packet content
  - Tag information
    - PCP
    - CFI
    - TPID
  - MEP ID
  - CCM Interval
  - MD level
- Per instance configuration for CCM packet content
  - Tag status
  - VID
  - MAID
- Per instance configuration for CCM transmit
  - Transmit enable state
  - Transmit interval (0 ~ 1023 msec)
  - 2 member ports



Realtek Semiconductor Corp.

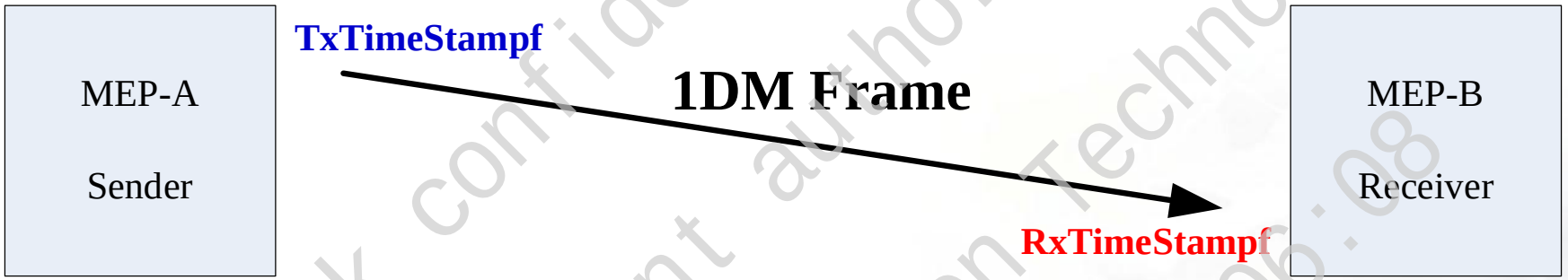
ETH-DM

ITU-T Y.1731 Support

- Used for on-demand OAM to measure frame delay and frame delay variation.
- System time clock resolution: 8ns
  - Clock jitter compensation mechanism supported
- One-Way / Two-Way ETH-DM supported
- Rx time stamp hardware recording
  - 64 global RX time-stamp buffer entries
  - Interrupt signal of time stamp recording
- TX time stamp hardware writing
  - CPU Tag controlled to trigger TX time stamp writing

One Way Delay Measurement Mechanism

- One-Way Delay Measurement (1DM) frame

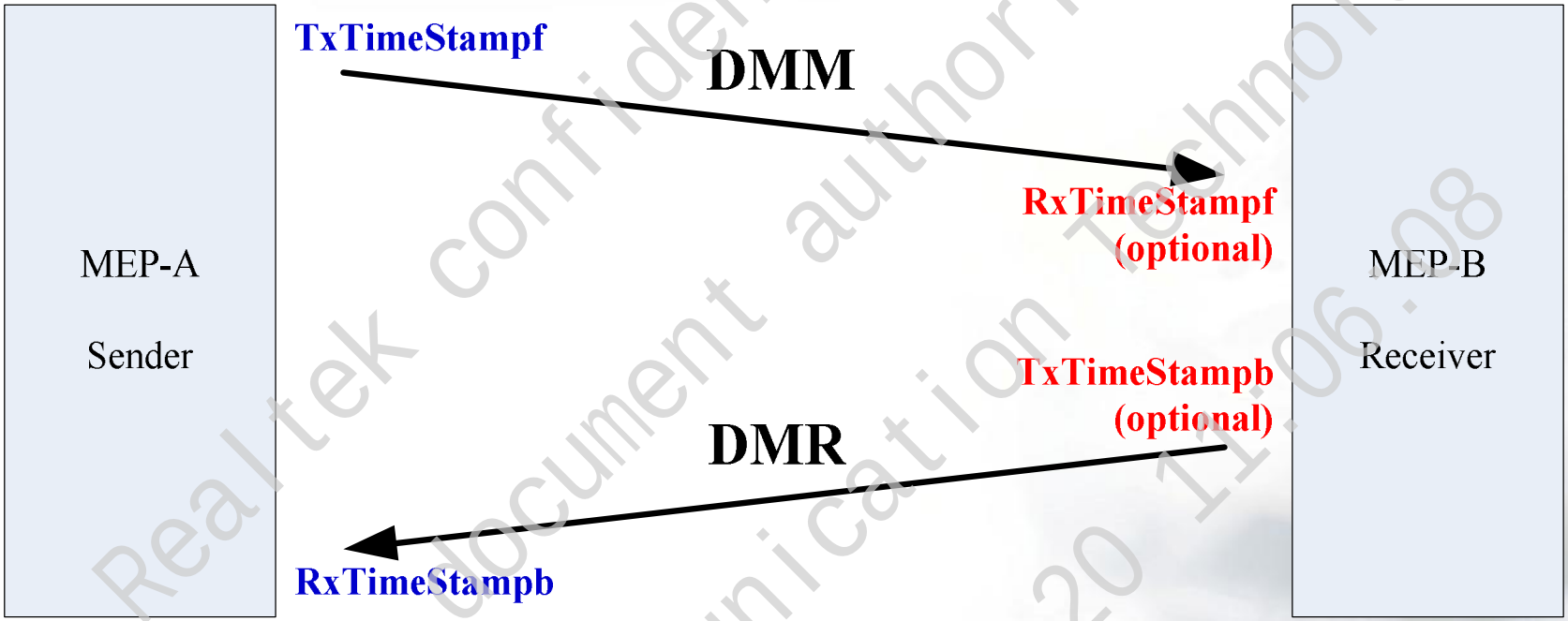


One-way Frame Delay = RxTimeStampf – TxTimeStampf
(if the clocks between MEP-A and MEP-B are sychronized)

Frame Delay Variation (n) = | Frame Delay (n) – Frame Delay (n-1) |

Two Way Delay Measurement

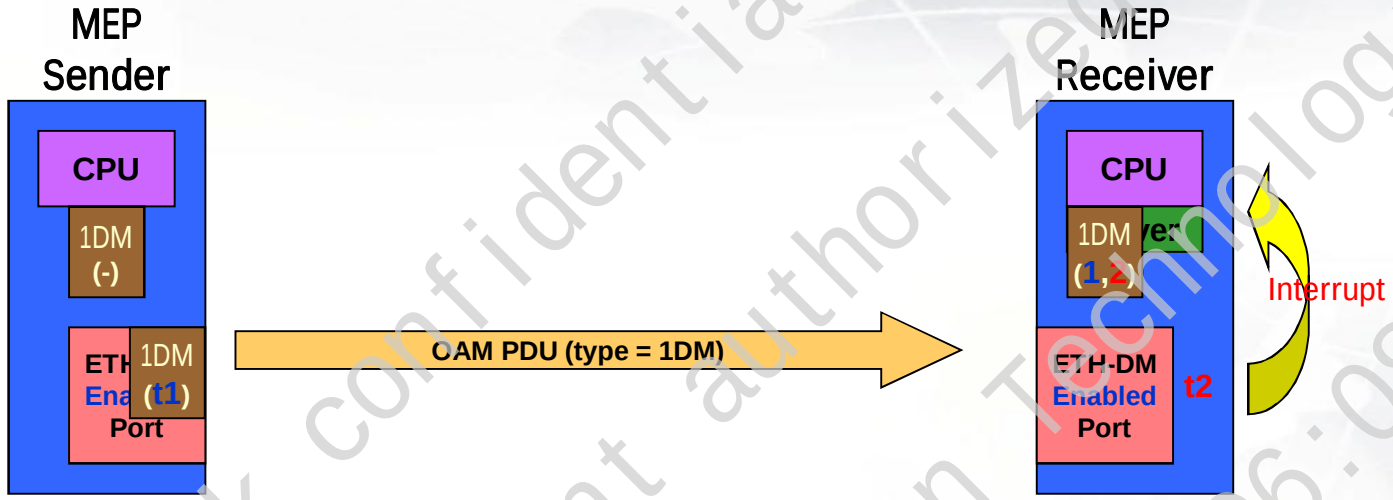
- Two-Way Delay Measurement Message (DMM) frame
- Two-Way Delay Measurement Reply (DMR) frame



Two-way Frame Delay =
(RxTimeStampb – TxTimeStampf) – (TxTimeStampb – RxTimeStampf)

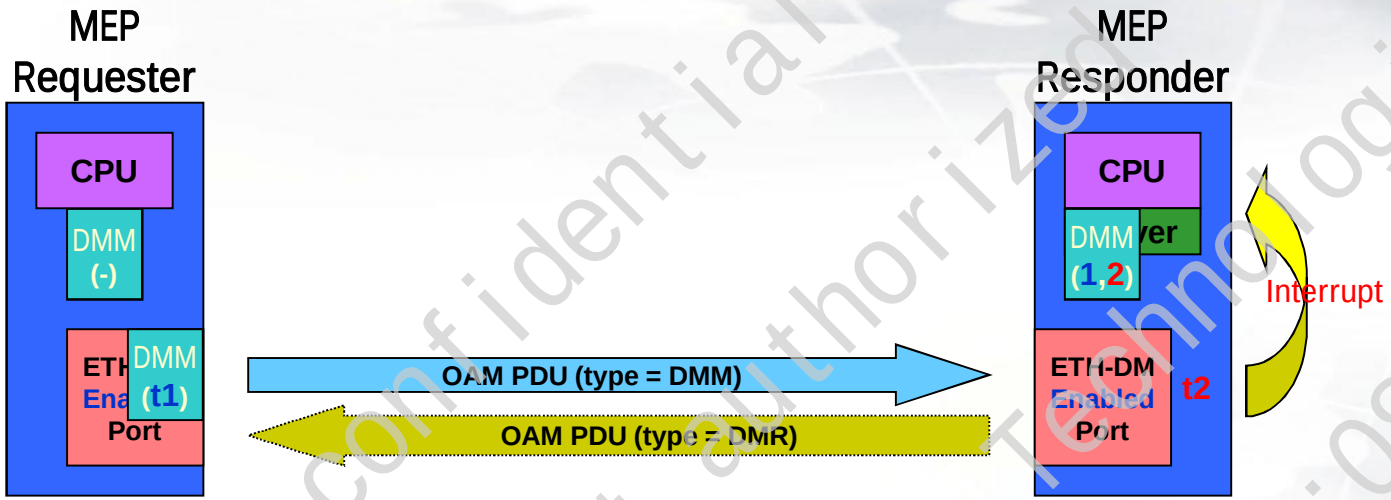
Frame Delay Variation (n) = | Frame Delay (n) – Frame Delay (n-1) |

1DM Hardware Support



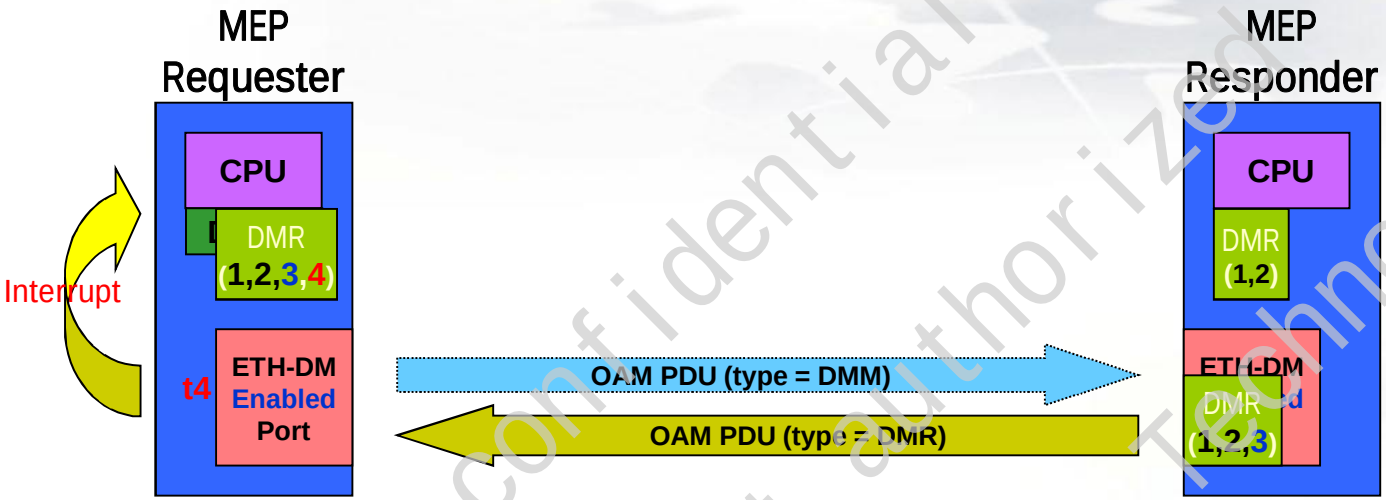
1. CPU sends an one-way ETH-DM (1DM) packet with empty *TxTimeStampf* by CPU tag.
2. When the packet is transmitted on the ETH-DM enabled port of Sender, the H/W will write the Tx time stamp (t1) into the packet content (*TxTimeStampf field*).
3. When the packet is received on the ETH-DM enabled port of Receiver, the H/W will record the Rx time stamp (t2) into the database and interrupt CPU.
4. Driver will read the Rx time stamp (t2) and save it in ISR.
5. When Driver receives this 1DM packet, it will fill the Rx time stamp (t2) into the packet content (*RxTimeStampf* field), and deliver this packet to the higher layer.
6. Finally, the CPU of Receiver will receive the completed packet that has the 2 time stamps.

DMM Hardware Support



1. CPU sends an two-way ETH-DM (DMM) packet with empty *TxTimeStampf* by CPU tag.
2. When the packet is transmitted on the ETH-DM enabled port of Requester, the H/W will write the Tx time stamp (t1) into the packet content (*TxTimeStampf field*).
3. When the packet is received on the ETH-DM enabled port of Responder, the H/W will record the Rx time stamp (t2) into the database and interrupt CPU.
4. Driver will read the Rx time stamp (t2) and save it in ISR.
5. When Driver receives this DMM Request packet, it will fill the Rx time stamp (t2) into the packet content (*RxTimeStampf* field), and deliver this packet to the higher layer.
6. Finally, the CPU of Responder will receive the completed packet that has the 2 time stamps.

DMR Hardware Support

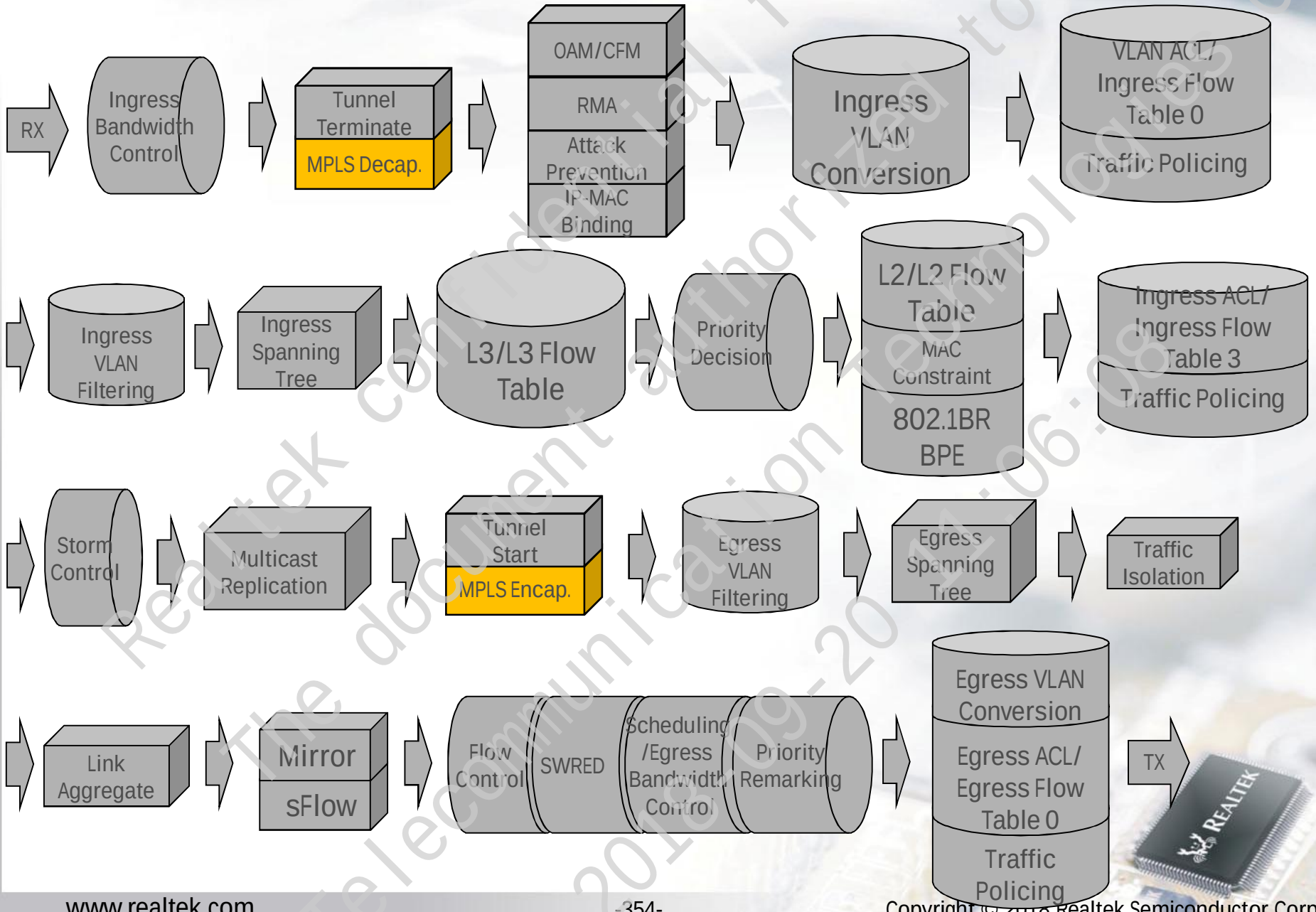


1. CPU sends an two-way ETH-DM (DMR) packet with empty *TxTimeStampb* by CPU tag.
2. When the packet is transmitted on the ETH-DM enabled port of Responder, the H/W will write the Tx time stamp (t3) into the packet content (*TxTimeStampb field*).
3. When the packet is received on the ETH-DM enabled port of Requester, the H/W will record the Rx time stamp (t4) into the database and interrupt CPU.
4. Driver will read the Rx time stamp (t4) and save it.
5. When Driver receives this DMR Reply packet, it will fill the Rx time stamp (t4) into the packet content (*RxTimeStampb* field), and deliver this packet to the higher layer.
6. Finally, the CPU of Requester will receive the completed packet that has the 4 time stamps.

Realtek Semiconductor Corp.

MPLS

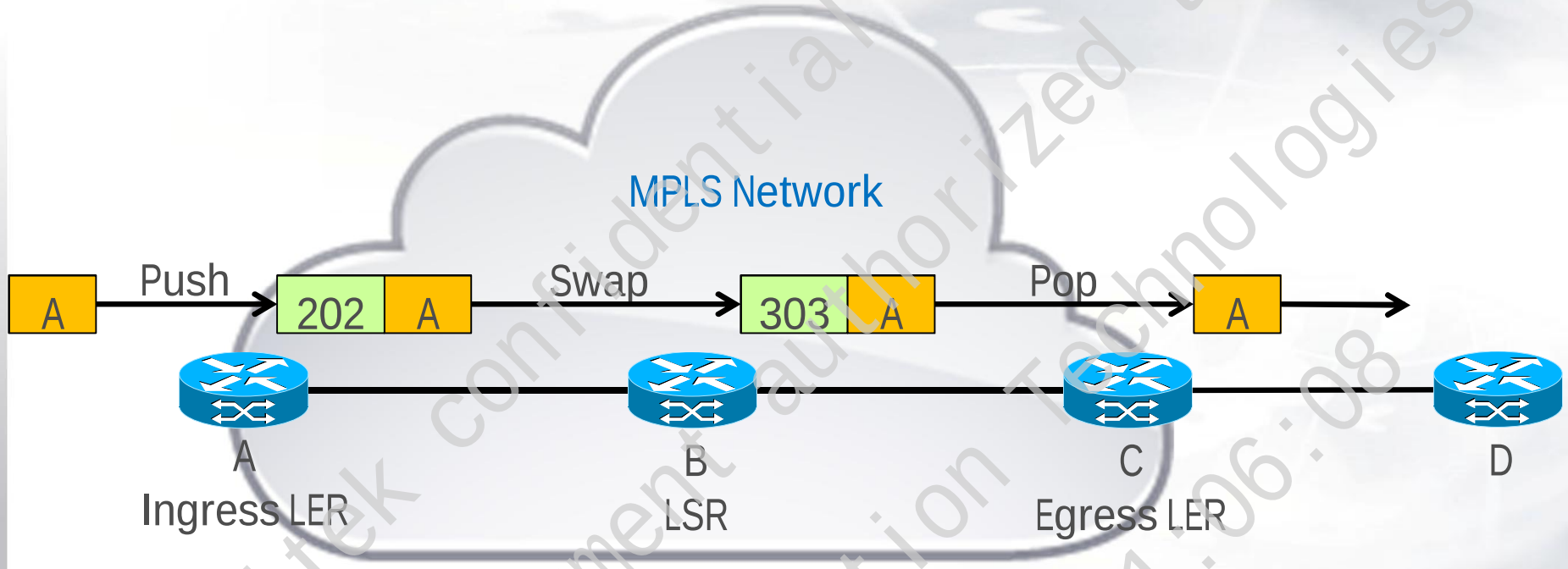
Packet Processing Pipeline



MPLS Overview

- 2K MPLS entries
- Up to 2 MPLS label
  - Pushing up to 2 labels
  - Popping up to 2 labels
  - Swapping 1 label and pushing one label
- Push, pop, swap, PHP actions
- MPLS DiffServ

MPLS Forwarding Behavior



A's LFIB

Network	In label	Out label	NH
A	-	202	B

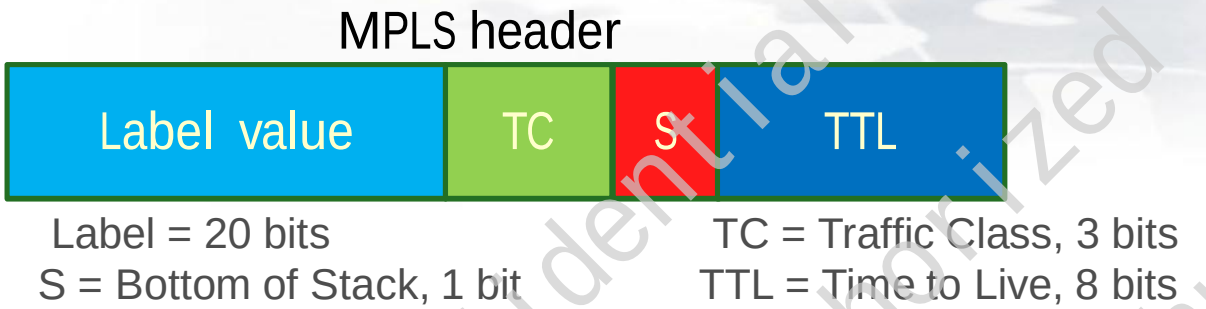
B's LFIB

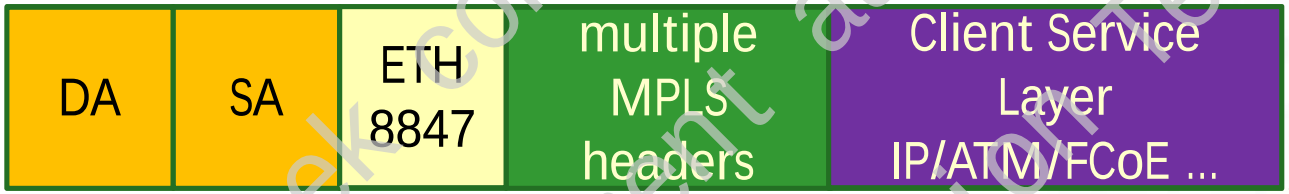
Network	In label	Out label	NH
A	202	303	C

C's LFIB

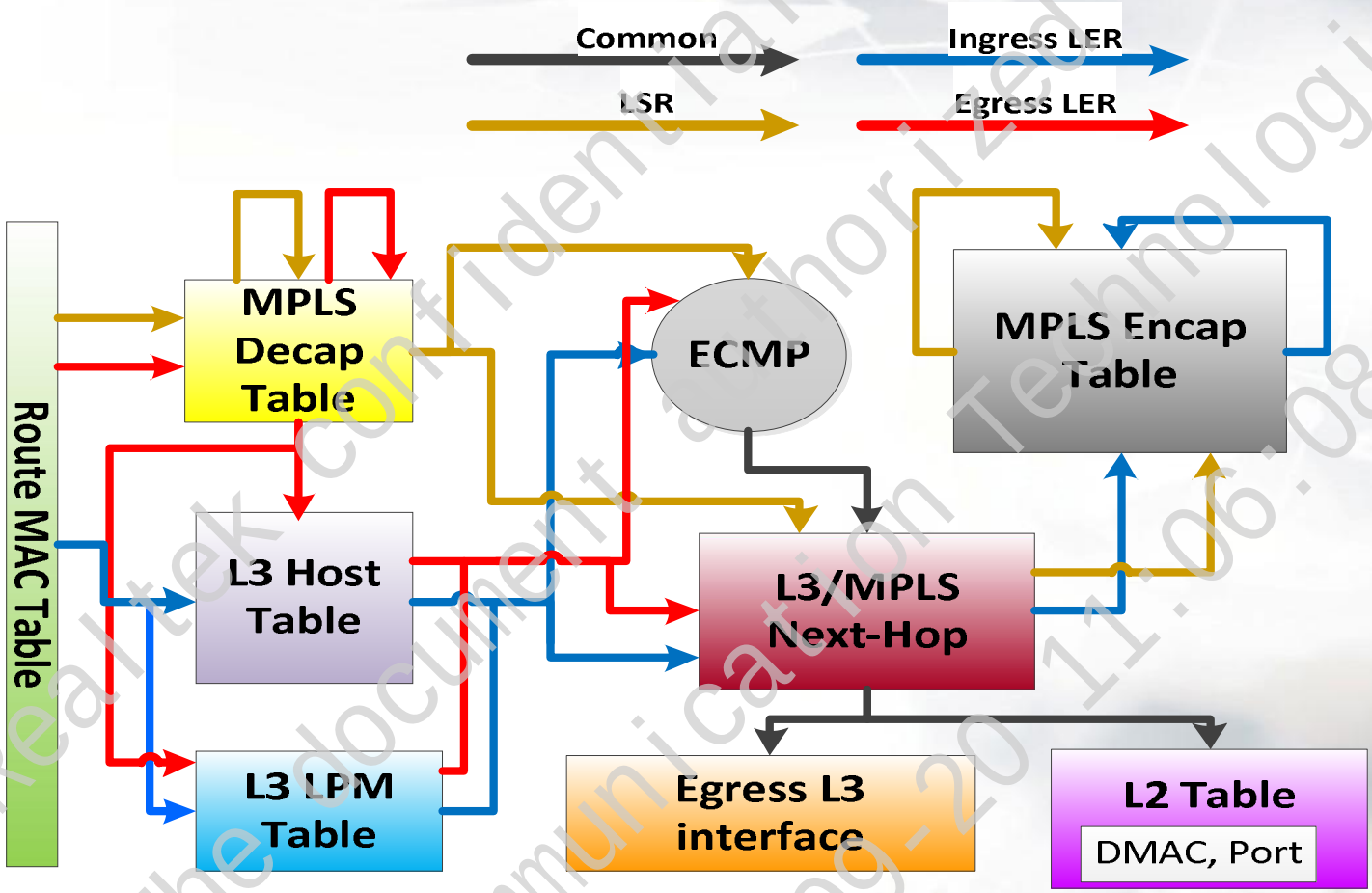
Network	In label	Out label	NH
A	303	-	D

MPLS Packet Format





MPLS Block Diagram



MPLS - Encapsulation

- Encapsulation process is triggered when
  - L3 routing to MPLS router
  - MPLS swap action
- 2K encapsulation entries
  - Indexed by routing NH assignment
- Per entry configuration
  - New label (20 bits)
  - Push or Swap
  - TTL operation: Inherit/Assign
  - TC operation: Inherit/Keep/Assign/Internal priority mapping
  - Next label index for pushing outer label
- 2 Label push capable

MPLS - Decapsulation

- Label lookup for MPLS tag of packet
  - Ethertype: 0x8847
- 2K decapsulation entries for hash lookup
- Per entry configuration
  - Label (key)
  - Pop, Swap or PHP
    - Pop: Ingress interface index
    - Swap and PHP: Next hop or ECMP
  - Inherit TTL to next label / IP header
  - Inherit TC to next label / IP header
  - Priority selection group
  - Internal priority assignment
- 2 label pop capable
  - When first label is popped, second label lookup is executed

MPLS – DiffServ

Tunnel Mode	IP -> Label (Ingress LER)	Label -> Label (LSR)	Label -> IP (Egress LER)
Uniform	Copy IP Prec/DiffServ into MPLS TC (may be changed by the Service Provider)	MPLS TC may be changed by Service Provider	MPLS TC copied to IP Prec/DiffServ (egress queuing based on IP Prec/DiffServ)
Pipe	MPLS TC set by the Service Provider QoS policy		Original IP Prec/DiffServ preserved (egress queuing based on MPLS TC)
Short-Pipe			Original IP Prec/DiffServ preserved (egress queuing based on IP Prec/DiffServ)

- Per encap entry configuration
  - TC from MSB 3 bits of DSCP
  - Assign specific TC
  - Internal priority maps to TC
- Per decap entry configuration
  - Copy TC to MSB 3 bits of DSCP
  - Assign specific internal priority (L-LSP)
  - TC maps to internal priority (E-LSP)
  - Priority selection group chooses internal priority from MPLS or IP

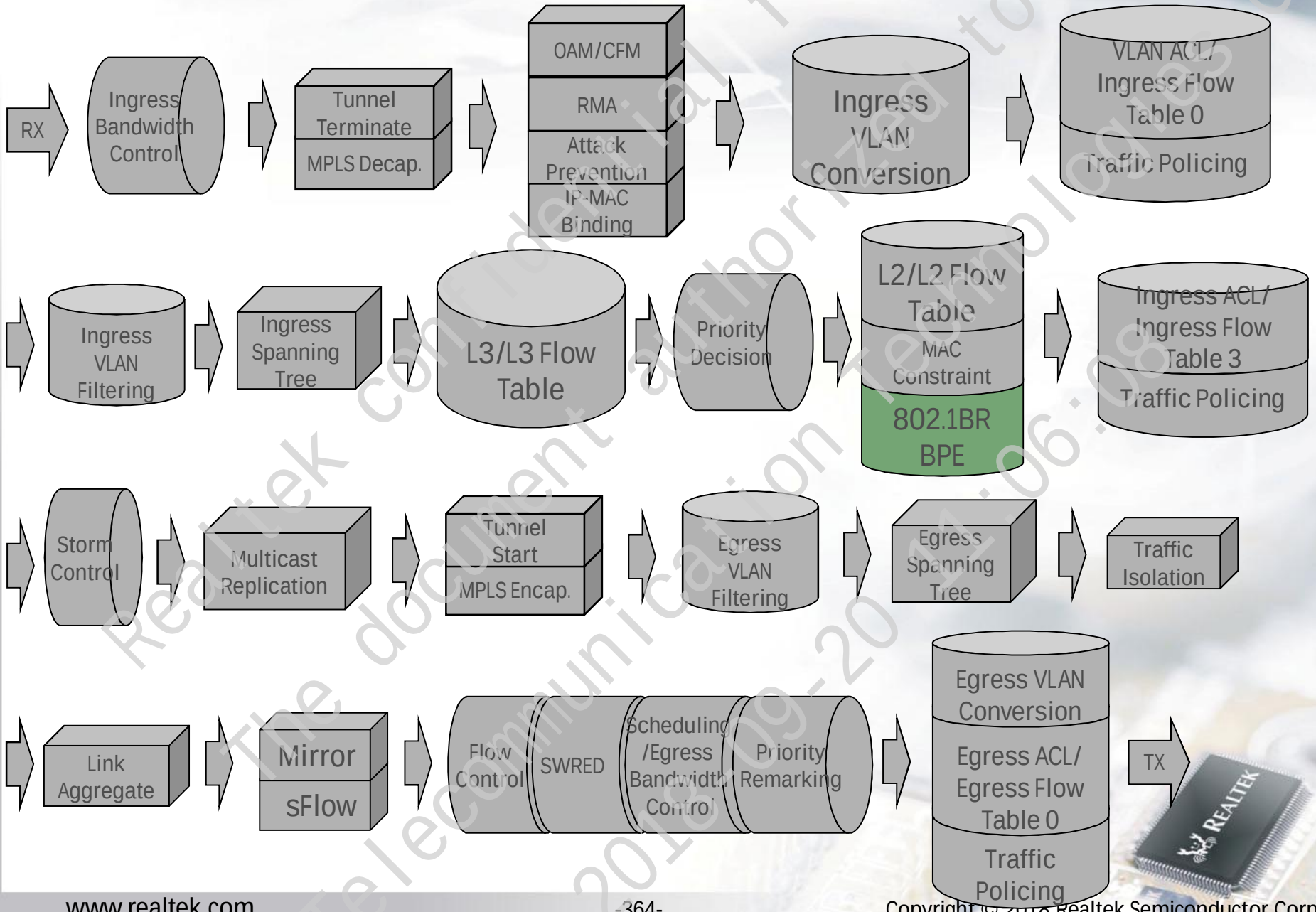
MPLS – Exception handle

- Configurations for below exceptions
  - TTL = 0 : drop/trap
  - Unknown label : drop/trap
  - Label number exceed : drop/trap
  - Reserved label (0 ~ 15) : trap/forward

Realtek Semiconductor Corp.

802.1BR
Bridge Port Extension

Packet Processing Pipeline



Outline

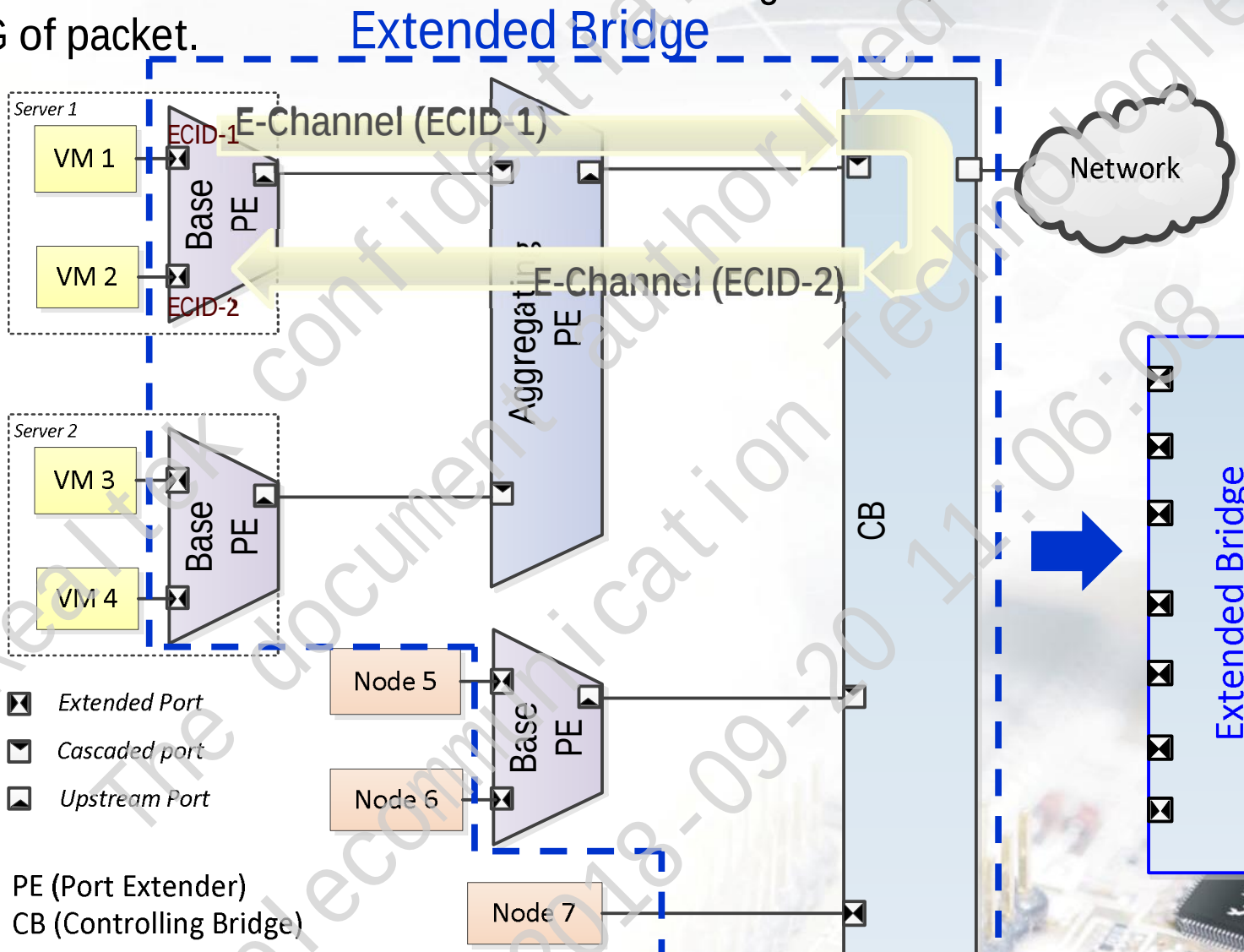
- Overview
- Topology
- Unicast over E-Channel
- Cascaded Port Extender
- NSG
- Multicast over E-Channel – Upstream
- Multicast over E-Channel – Downstream
- Table Relation

Overview

- Per port configuration
  - Enable processing E-TAG packet
  - L2 destination lookup
    - CB (control bridge) mode {FID, MAC_DA} , so as normal mode
    - PE (port extender) mode {NSG, ECID}
  - Ingress filter
    - Reserved ECID (0x0 and 0x3FFFFF)
    - Multicast ECID (GRP != 0)
  - Egress control
    - Keep priority/ remark DEI
    - E-Tag status
    - Source extended port filter
  - Configuration to replace extension bits (USE_DEFAULT)
  - Configuration to ignore extension bits (BASE_PE)
- Related tables
  - 8k ECID portmask list table (configurable table for ECID portmask or MPLS nexthop)
  - 32k PE forwarding table (shared with L2 table)
  - 2k ECID to PVID table (extra 64 entries for hash collision)

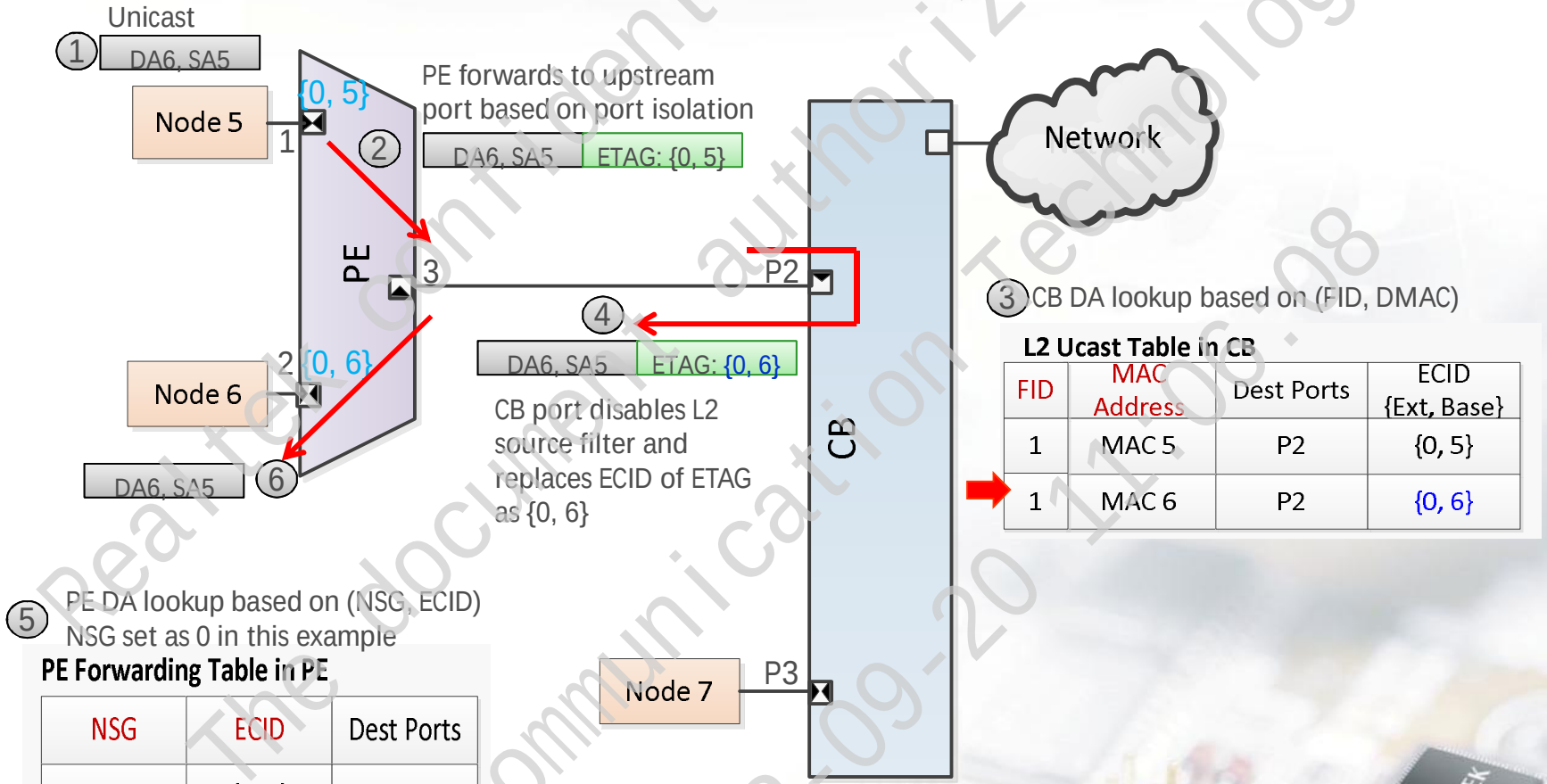
Topology

- Traffic on the E-Channel is forwarded according to ECID, which is carried in the ETAG of packet.



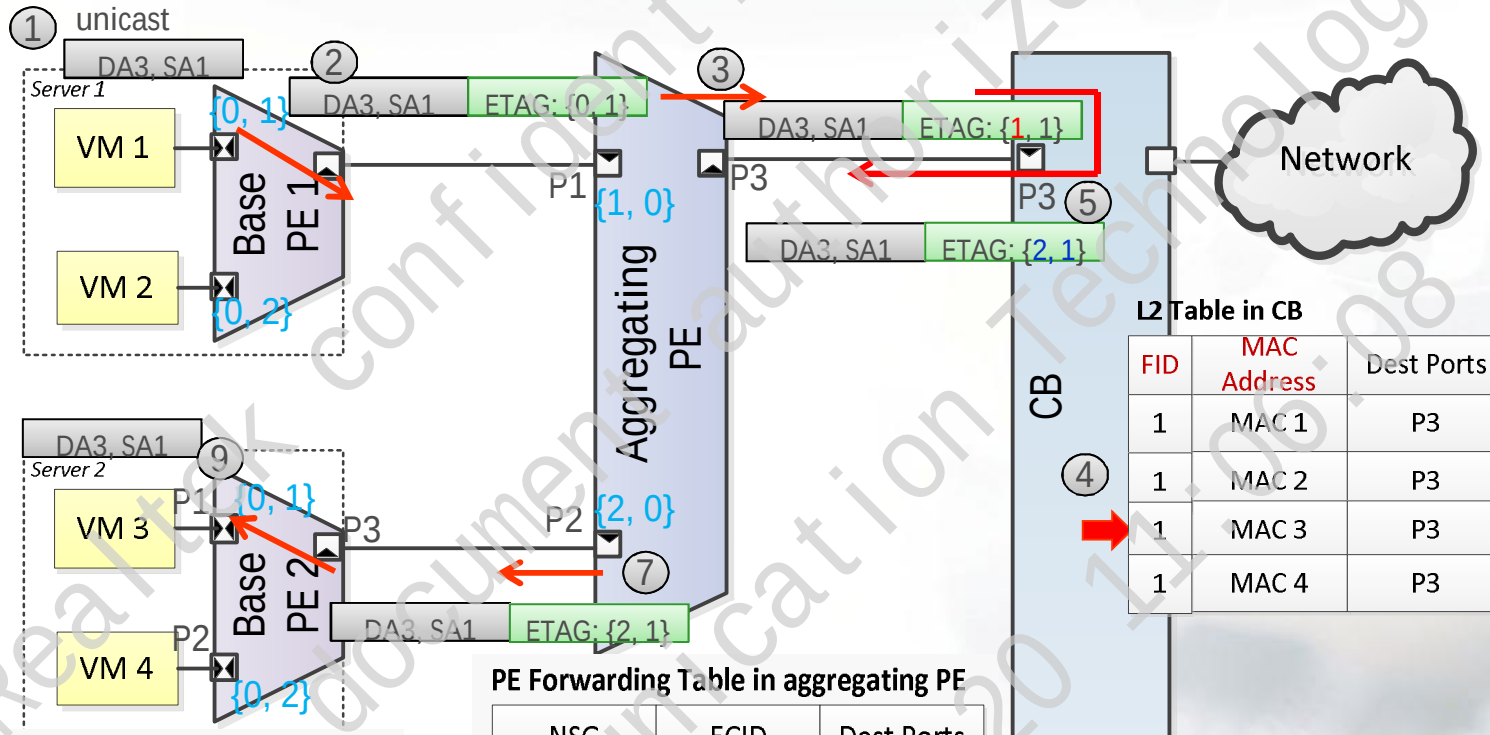
Unicast over E-Channel

- Example: unicast from node 5 to node 6



Cascaded Port Extender

- Aggregating PE replaces extension bits when USE_DEFAULT is enabled.
- Base PE is only capable to process base bits (ignore extension bits of ECID).



L2 Table in CB

FID	MAC Address	Dest Ports	ECID {EXT, BASE}
1	MAC 1	P3	{1, 1}
1	MAC 2	P3	{1, 2}
1	MAC 3	P3	{2, 1}
1	MAC 4	P3	{2, 2}

PE Forwarding Table in aggregating PE

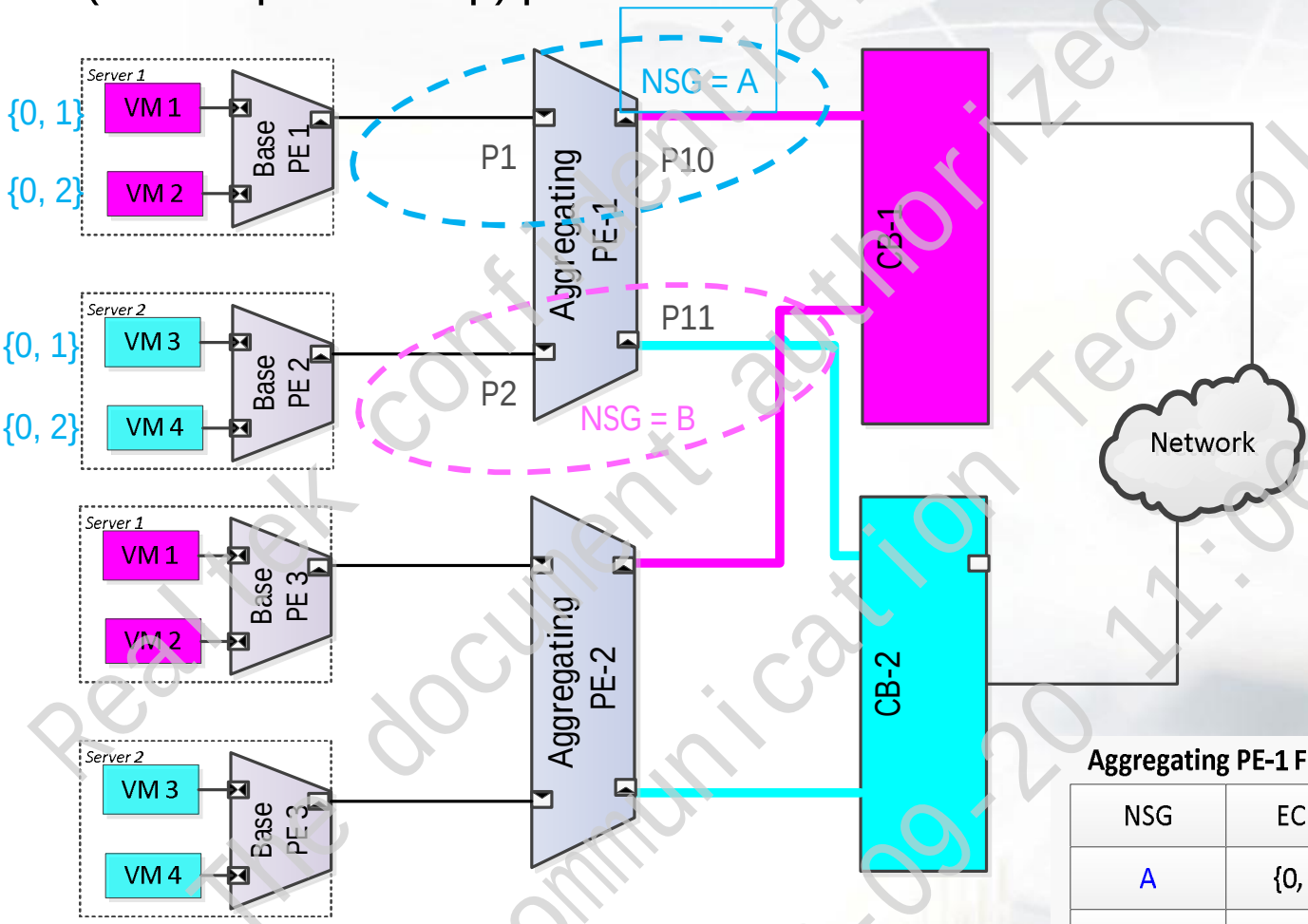
NSG	ECID	Dest Ports
0	{1, 1}	P1, P3
0	{1, 2}	P1, P3
6 → 0	{2, 1}	P2, P3
0	{2, 2}	P2, P3

PE Forwarding Table in Base PE 2

8	NSG	ECID	Dest Ports
→	0	{0, 1}	P1, P3
	0	{0, 2}	P2, P3

NSG

- NSG (Name Space Group) provides virtual domain in the same device.

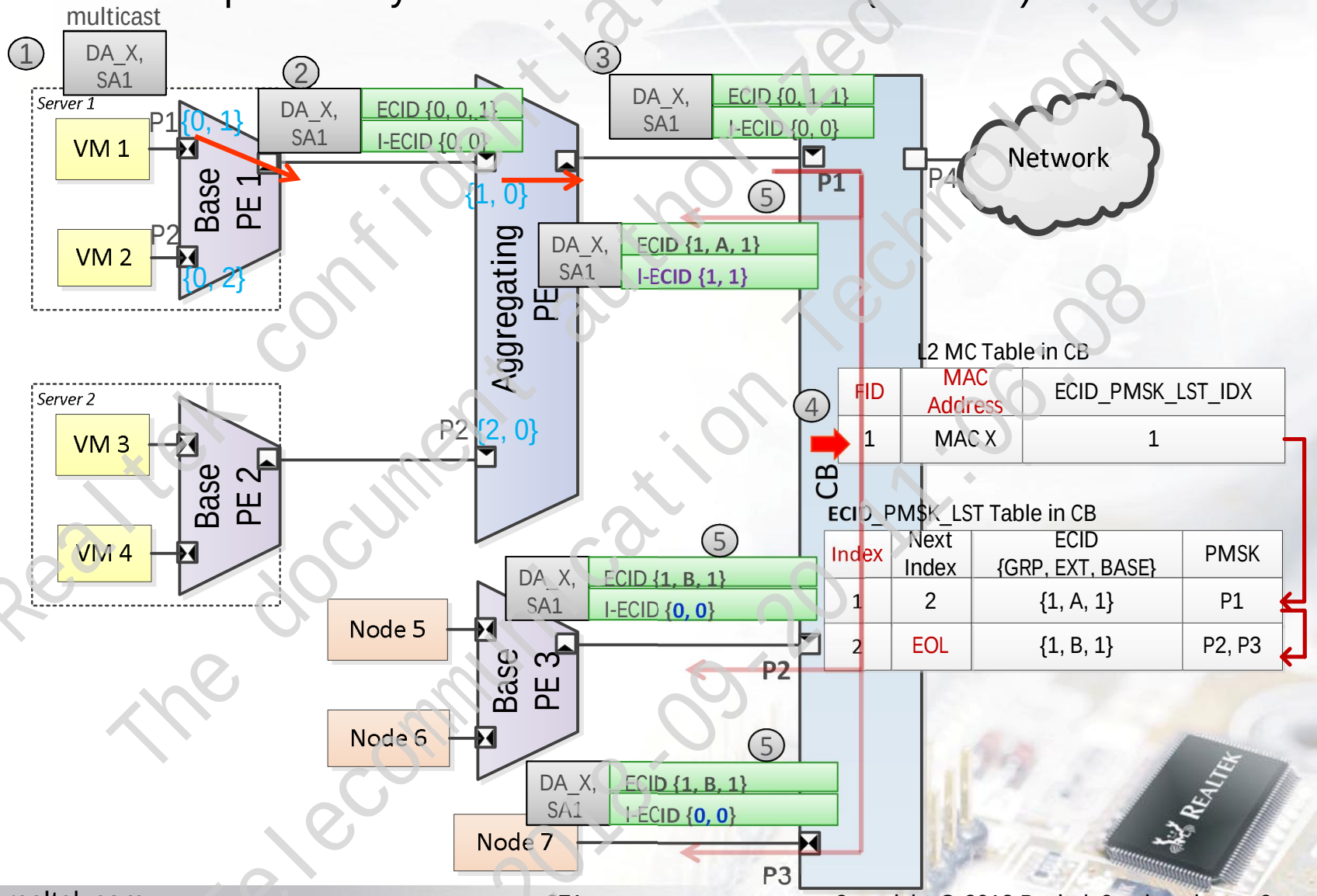


Aggregating PE-1 Forwarding Table

NSG	ECID	Dest Ports
A	{0, 1}	P1, P10
A	{0, 2}	P1, P10
B	{0, 1}	P2, P11
B	{0, 2}	P2, P11

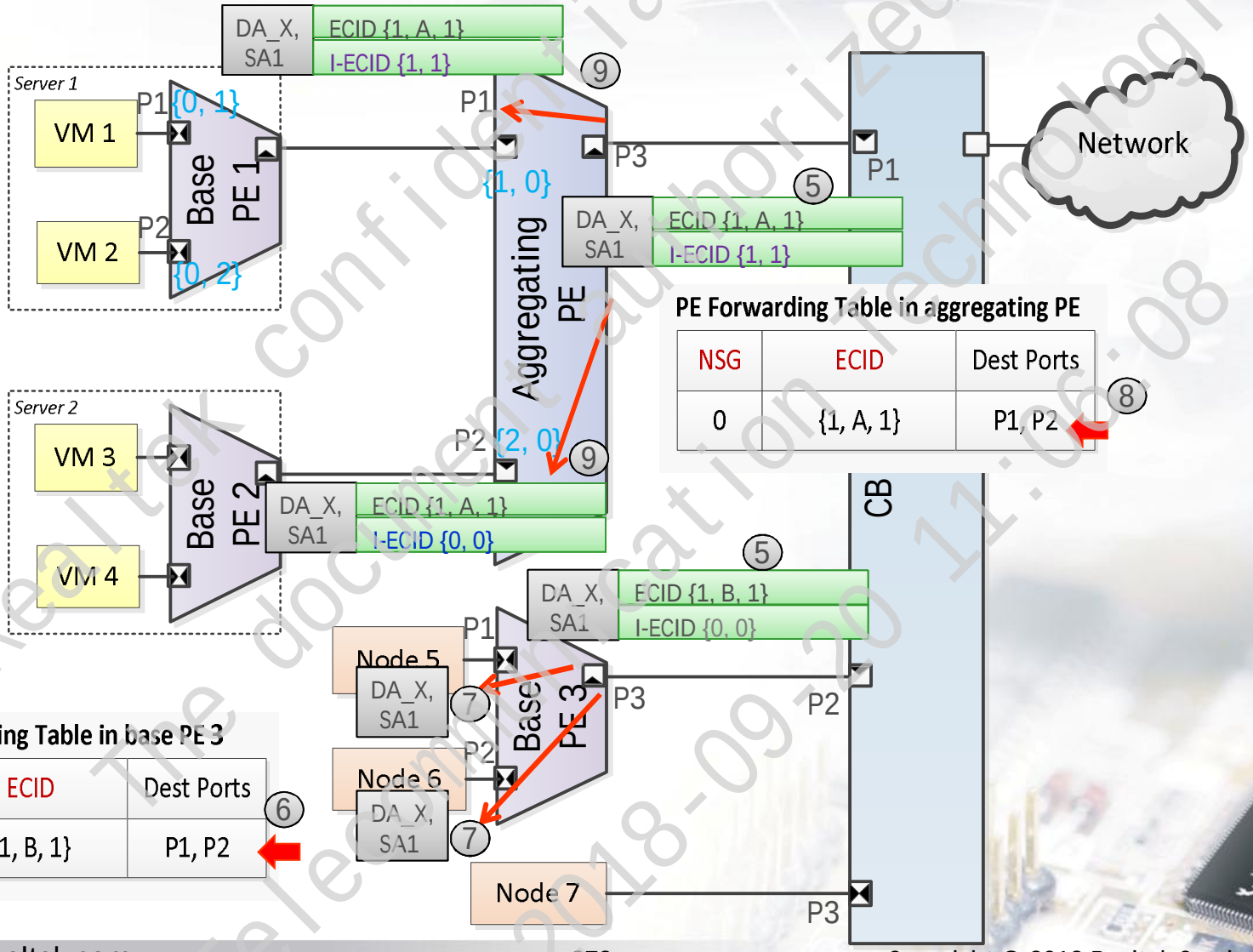
Multicast over E-Channel - Upstream

- L2 multicast is replicated by CB with multicast channel (GRP != 0).



Multicast over E-Channel -Downstream

- I-ECID is used for source extended port filter. Otherwise, set as zero.



Source Extended Port Filter

- Multicast should not be forwarded to the source port (verified by I-ECID).

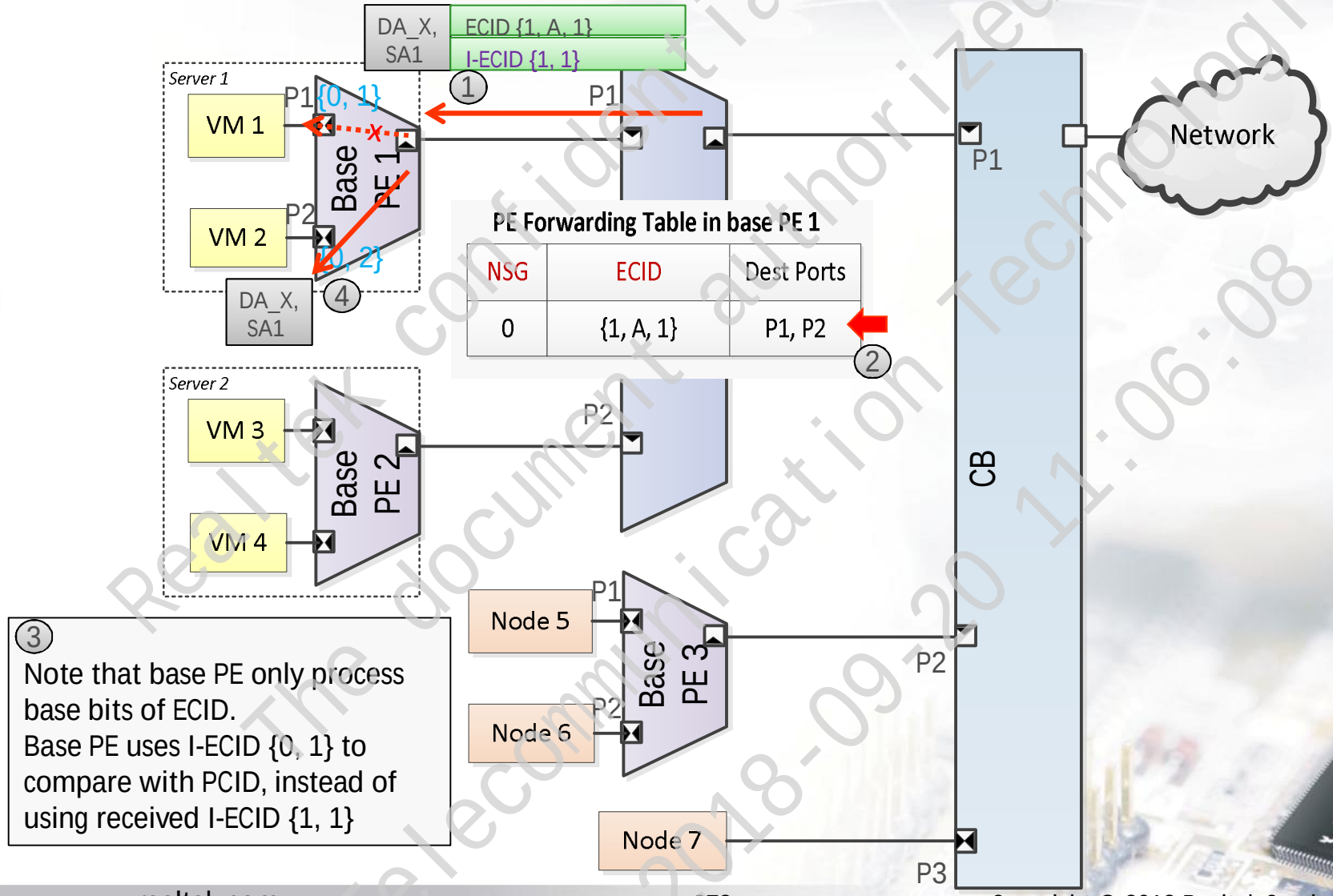
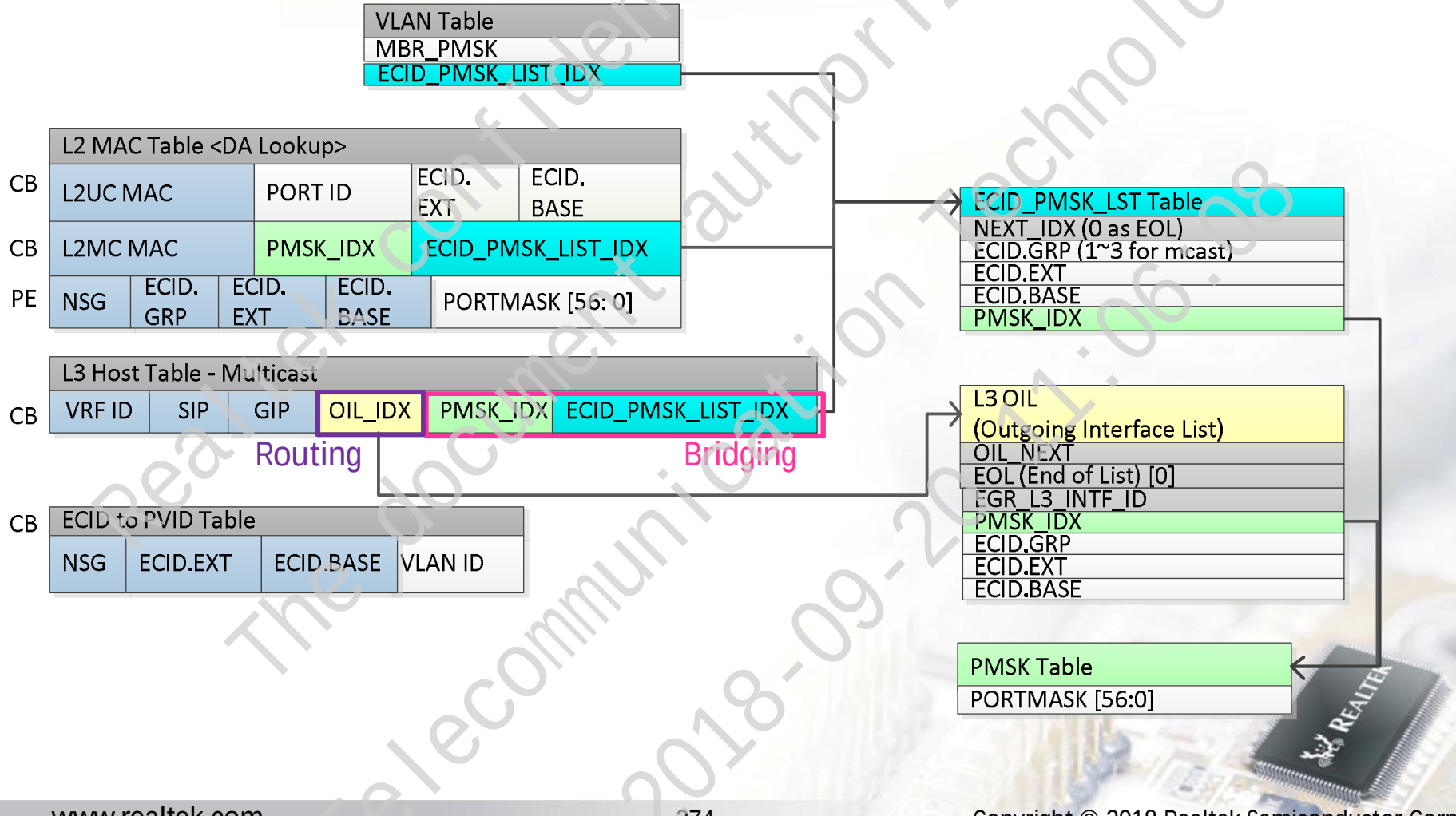


Table Relation

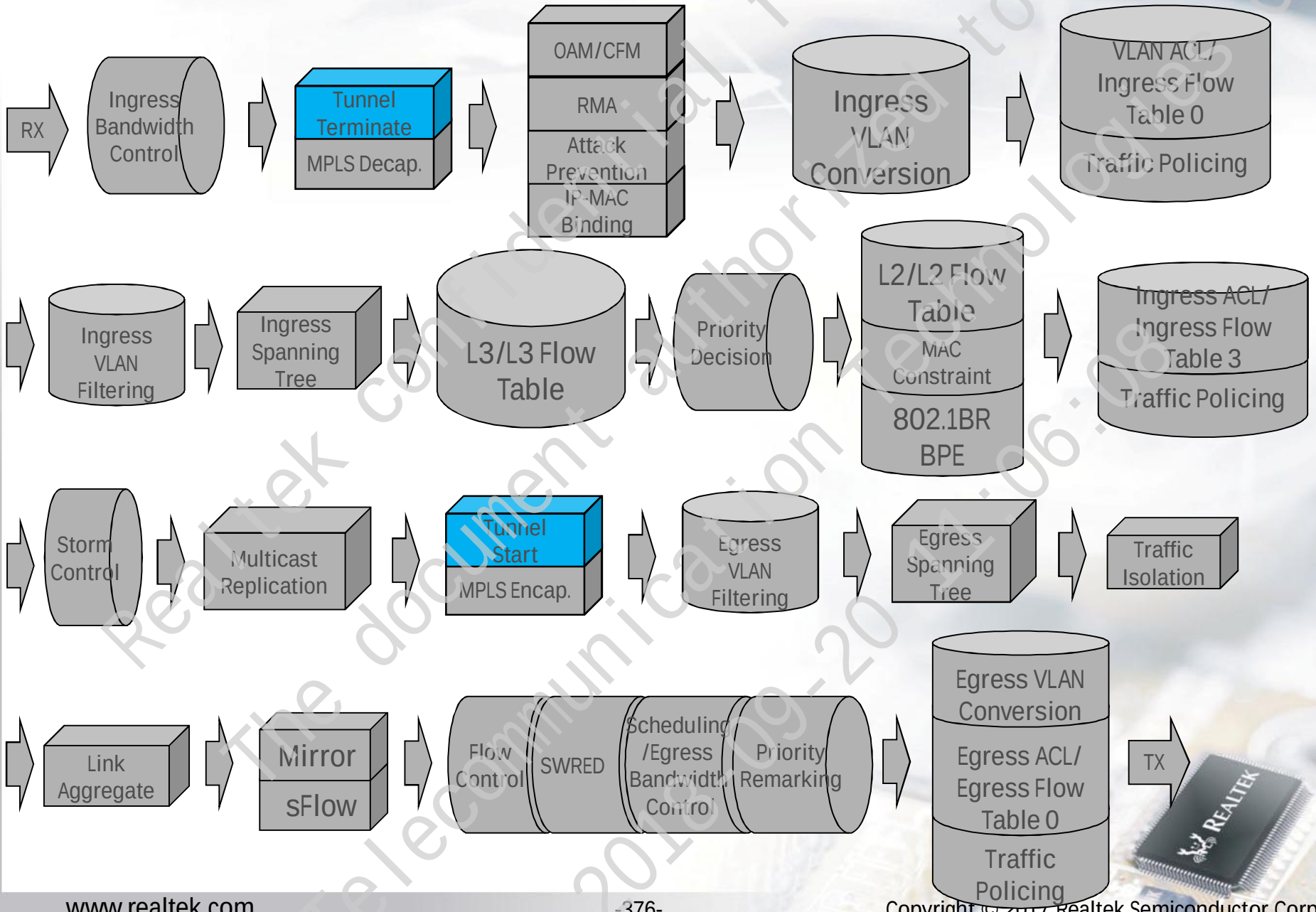
- BPE extend the description of port as {ECID + Port}
- Support both normal bridging and E-channel forwarding



Realtek Semiconductor Corp.

VXLAN

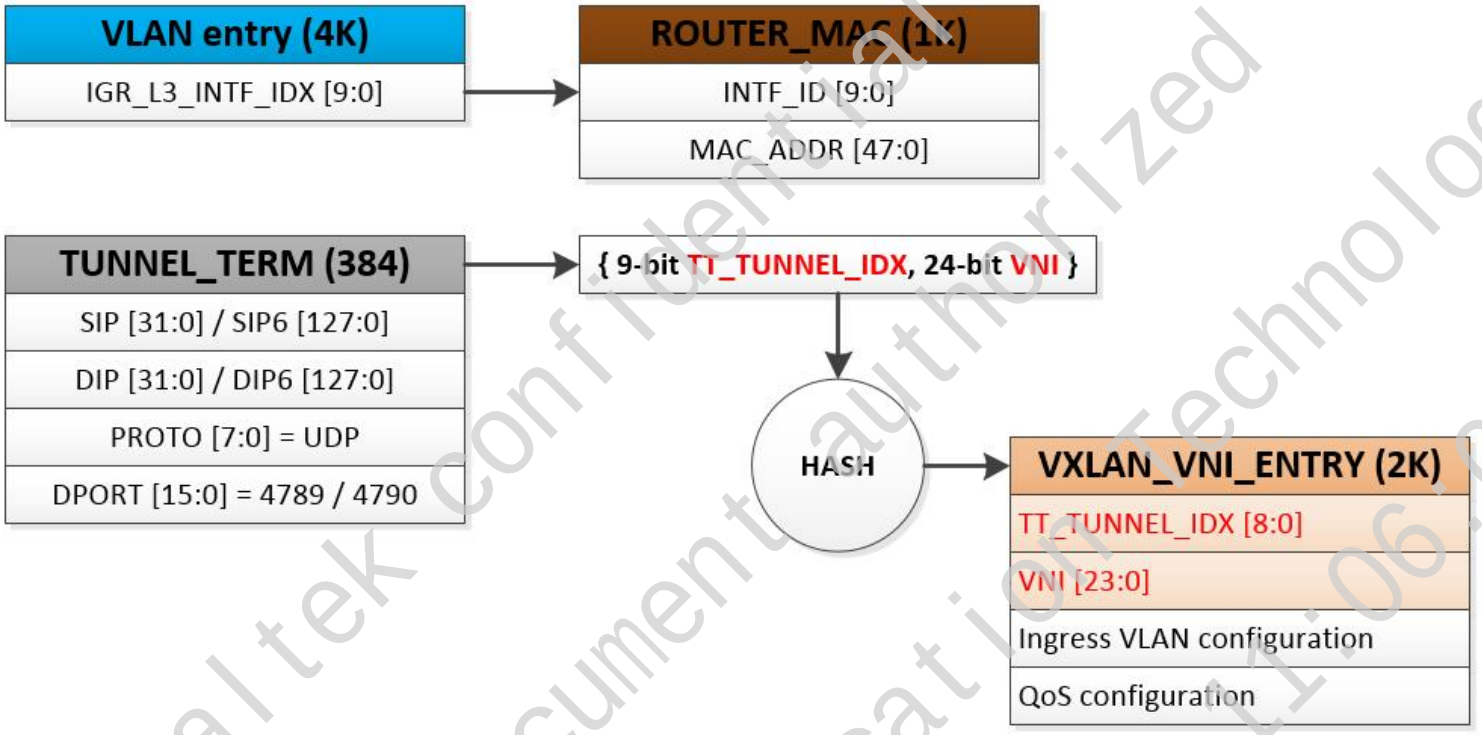
Packet Processing Pipeline



Overview

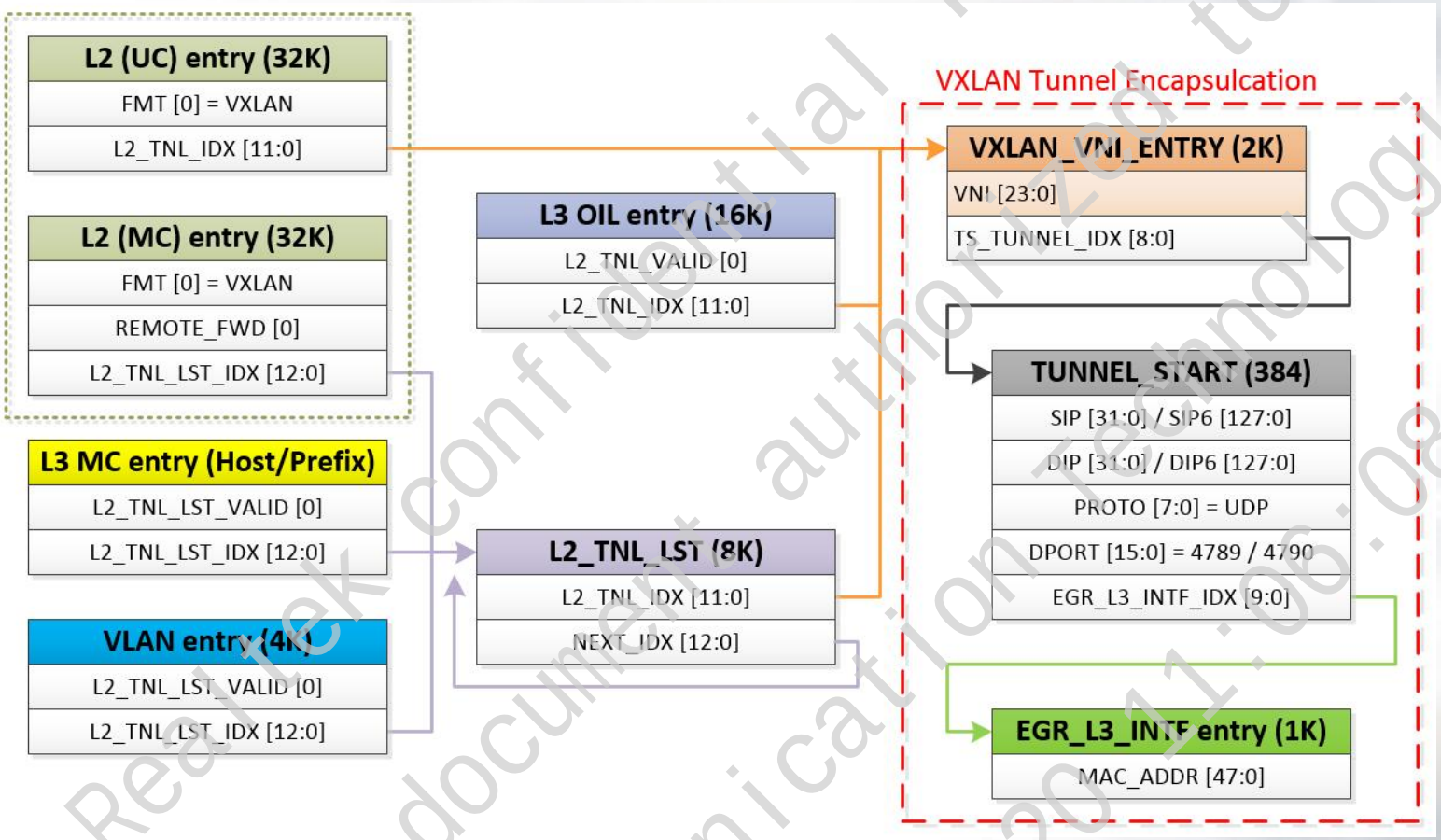
- Virtual eXtensible Local Area Network (VXLAN)
  - A framework for overlaying virtualized layer 2 networks over layer 3 networks
  - Documented in RFC 7348
  - Conceptually an L2 tunnel (interface) technology
  - Support the packet format of VLAN-GPE (RFC draft-ietf-nvo3-vxlan-gpe-06)
- 384 tunnel entries
  - 384 IPv4 or 128 IPv6 tunnels (IPv6 tunnel consumes 3 physical entries)
  - Shared with L3 IP tunnels
- 2048 VNI (Virtual Network Identifier) entries
  - Each VNI entry is bound with a specified VXLAN tunnel (End-to-End paired VTEPs)
  - 24-bit VNI as lookup key of VNI table
  - When termination, assign outer/inner VID and forwarding VLAN source
- Remote L2 MAC address auto-learning
  - Inner source MAC address is auto-learned with the bound VNI entry

VXLAN – Decapsulation (Table Relationship)



- Use outer header's VLAN to derive the ingress L3 interface and to lookup ROUTER_MAC table with { Ingress L3 interface, Outer DMAC }.
- { termination entry index, VNI } to lookup VNI table.
- VNI lookup miss action: Drop, Trap to CPU
- Per-VXLAN tunnel, Per-VNI to assign ingress VLAN and QoS.

VXLAN – Encapsulation (Table Relationship)

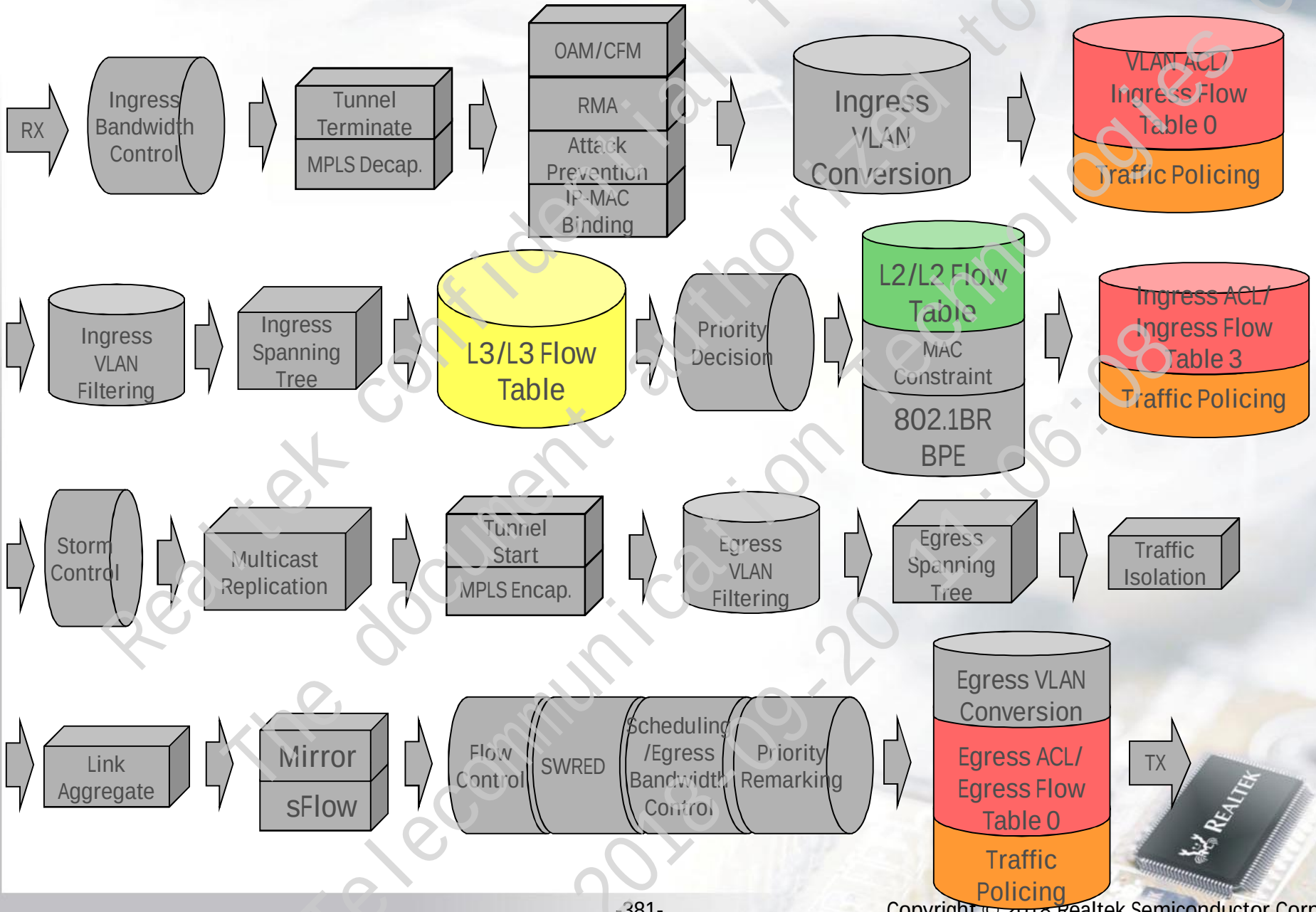


- L2 unicast entry can bind with a VNI entry as the L2 interface instead of “PORT”.
- L2/L3 multicast and VLAN (for flooding) entry can index to L2_TNL_LST table to replicate packets for bridging the packet to many VTEPs with different VNI.
- For L3 multicast routing, L3 OIL (outgoing interface list) entry is used to form VXLAN tunnel.

Realtek Semiconductor Corp.

OpenFlow

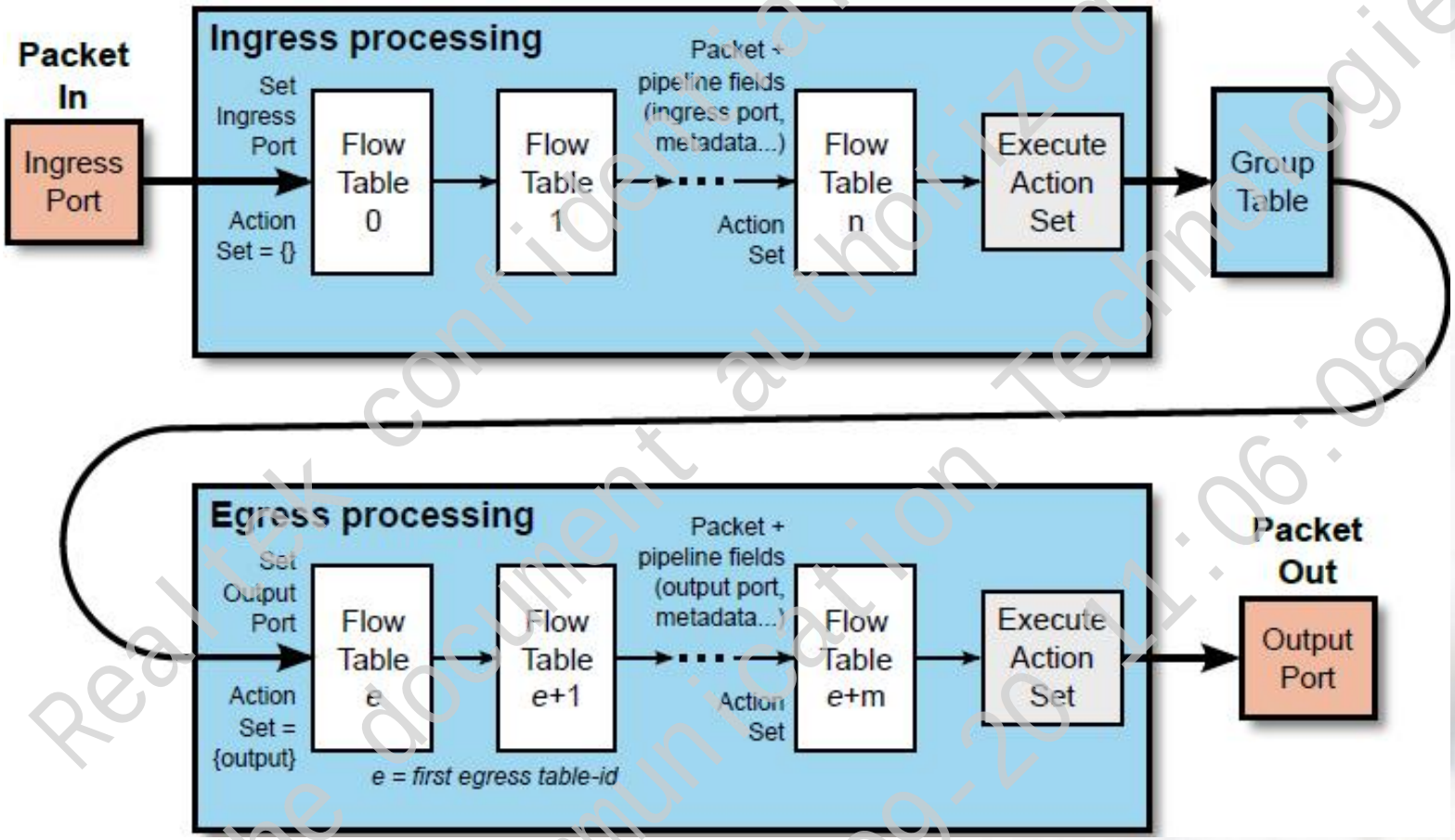
Packet Processing Pipeline



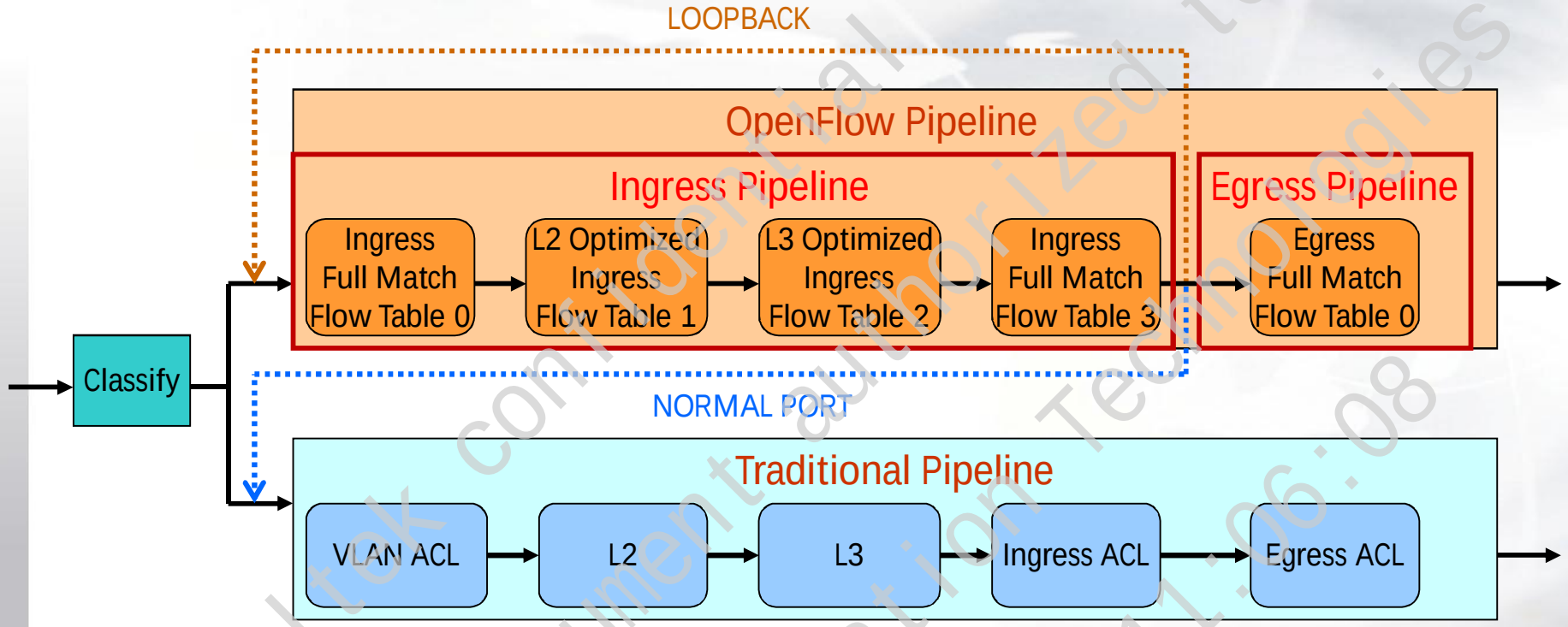
Feature Summary

- OpenFlow v1.5.1 Compliant
- OpenFlow Hybrid Switch
- Native Five Flow Tables
  - ✓ Ingress Pipeline
    - ✓ Ingress Flow Table 0
    - ✓ Ingress Flow Table 1 (L2 Optimized, 16K)
    - ✓ Ingress Flow Table 2 (L3 Optimized, 12K)
    - ✓ Ingress Flow Table 3
  - ✓ Egress Pipeline
    - ✓ Egress Flow Table 0
- Ingress Flow Table 0,3 and Egress Flow Table 0 share 4K entries
- Extensible Ingress Flow Table Stage through Loopback
- Table-Miss Actions
- Group Table Support
  - ✓ 2K group entry and 8K action buckets
- Meter Table Support
  - ✓ 512 dual band meter
  - ✓ Drop and DSCP Remark band types

Standard OpenFlow Packet Processing Pipeline



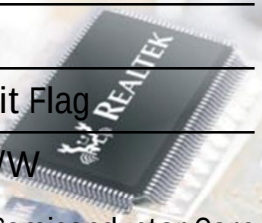
RTL9310 OpenFlow Hybrid Switch Pipeline



- Classification
  - ✓ Per port
  - ✓ Per VLAN
  - ✓ Per VLAN and Port

Flow Table Summary

		Ingress Full Match Flow Table 0	L2 Optimized Ingress Flow Table 1	L3 Optimized Ingress Flow Table 2	Ingress Full Match Flow Table 3	Egress Full Match Flow Table 0
Table Size		4K*	16K	6K TCAM-based 6K Hash-based	4K*	4K*
Match Fields		L2~L4 Width 240-bit	(0,SA/DA,MD) (VID,SA/DA,MD) (SA,DA,MD) (SIP,DIP,L4SPORT, L4DPORT,IPPRO/ VID)	(SIP,MD) (DIP,MD) (SIP,DIP,MD)	L2~L4 Width 240-bit	L2~L4 Width 240-bit
Priority		Block-based	N/A	N/A	Block-based	Block-based
Counters		36bit packet-based 42bit byte-based	N/A	N/A	36bit packet-based 42bit byte-based	36bit packet-based 42bit byte-based
Instruction	Meter	512*	N/A	N/A	512*	512*
	Apply Action	N/A	N/A	N/A	N/A	N/A
	Clear Action	V	V	V	V	X
	Write Action	V	V	V	V	V
	Write Metadata	V(12-bit)	V(12-bit)	V(12-bit)	V(12-bit)	X
	Goto Table	V	V	V	V	X
Timeouts		Hit Flag	Hit Flag	Hit Flag	Hit Flag	Hit Flag
Cookie		S/W	S/W	S/W	S/W	S/W



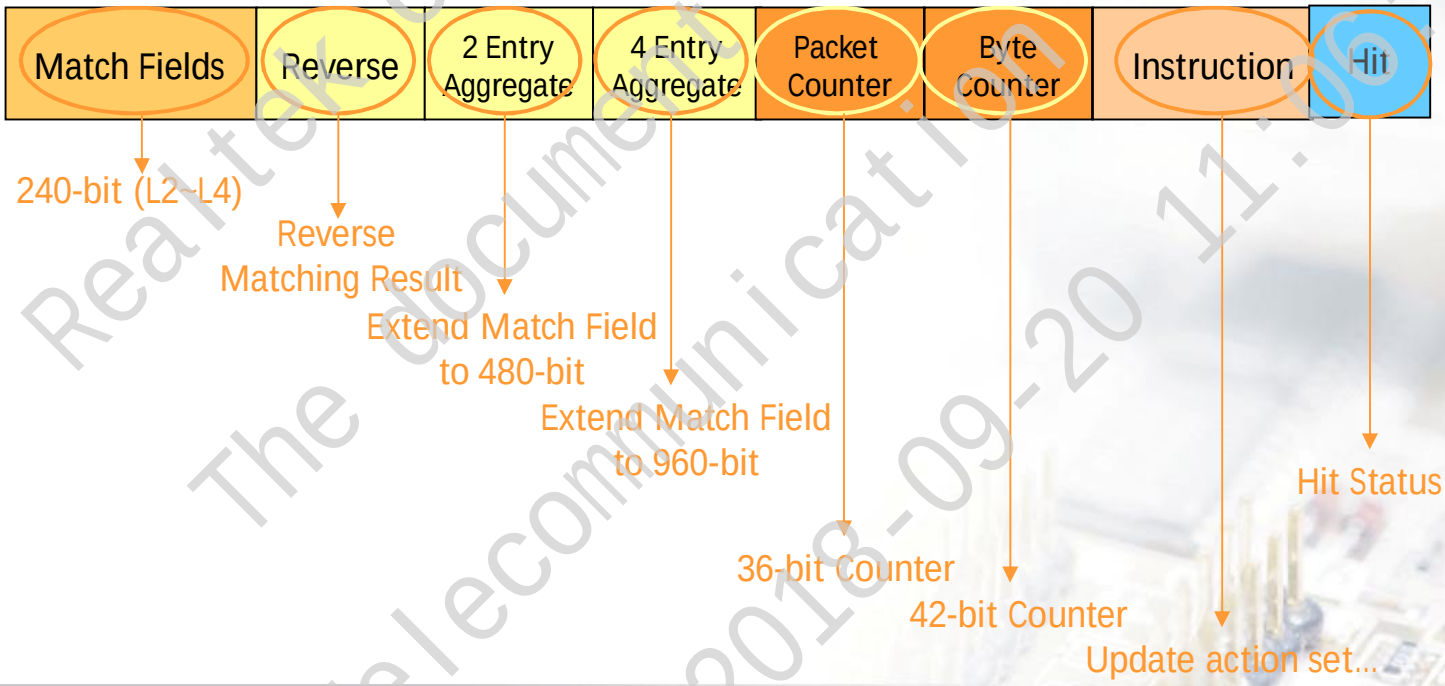
Action Set Summary

		Ingress Flow Table 0	Ingress Flow Table 1	Ingress Flow Table 2	Ingress Flow Table 3	Egress Flow Table 0
Copy TTL Inward		V			V	
Pop VLAN		V			V	V
Pop MPLS		V			V	
Push MPLS		V			V	
Push VLAN		V	V	V	V	V
Copy TTL Outward		V			V	
Dec MPLS TTL		V			V	
Dec IP TTL		V		V	V	V
Set Field	Max. Set Field	5	2	3	5	2
	SMAC/DMAC	V		V	V	
	VLAN ID/Prior.	V	V	V	V	V
	IP DSCP	V	V	V	V	V
	IP TTL	V			V	
	IPv4 SIP/DIP	V			V	
	SPORT/DPORT	V			V	
	MPLS Label/TC/TTL	V			V	
Set Queue		V	V	V	V	V
Group		V		V	V	V
Output		V	V	V	V	V

Full Match Flow Table

Full Match Flow Table Overview - 1/2

- 4K TCAM entries shared by Full Match Flow Tables and ACL Tables
  - ✓ 4K entries divided into 32 TCAM blocks
- Per TCAM Block
  - ✓ Can be Ingress Flow Table 0 or Ingress Flow Table 3 or Egress Flow Table 0 or VACL or IACL or EACL
  - ✓ Map to two templates
  - ✓ Block-based Priority for multiple matching arbitration
- Full Match Flow Table Entry Structure

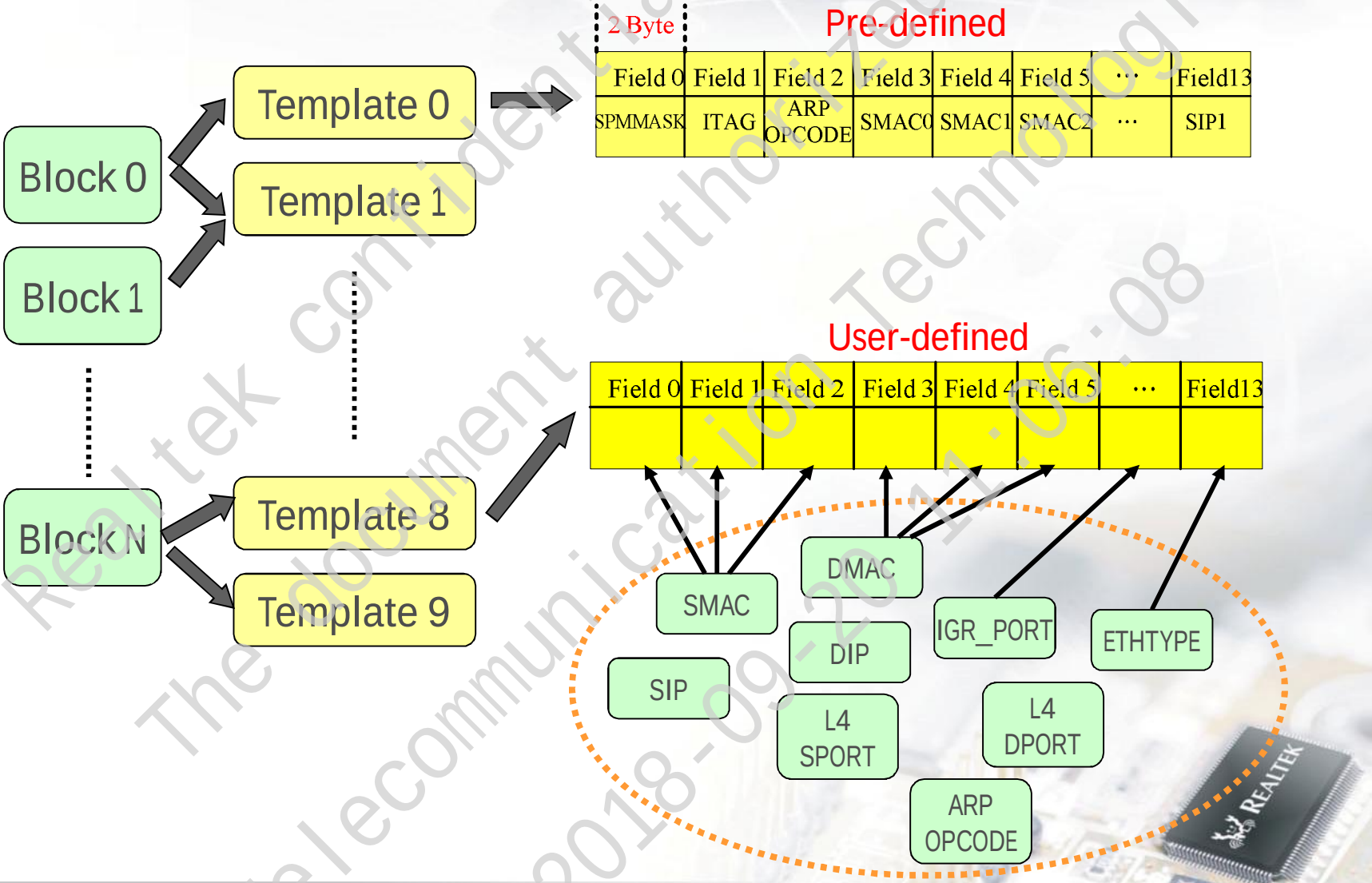


Full Match Flow Table Overview - 2/2

- Per Entry Supports
  - ✓ Reverse Operation
    - ✓ Drop 192.168.1.X except 192.168.1.1
    - ✓ Drop all the VLAN except VLAN 1~10
  - ✓ 2 Entry Match Field Aggregation Operation
    - ✓ Extend Match Field to 480-bit width
  - ✓ 4 Entry Match Field Aggregation Operation
    - ✓ Extend Match Field to 960-bit width
  - ✓ Counter Mode + 36-bit Packet Counter + 42-bit Byte Counter
    - ✓ STAT-TRIGGER instruction defined in OpenFlow v1.5.0
    - ✓ Counter mode can be one of below
      - ✓ Packet Counter + Byte Counter
      - ✓ Packet Counter + Packet Counter Trigger Threshold
      - ✓ Byte Counter + Byte Counter Trigger Threshold
    - ✓ Interrupt is triggered when counter over the specified threshold
  - ✓ Hit Indication Flag
    - ✓ Assist software to implement IDLE timeout

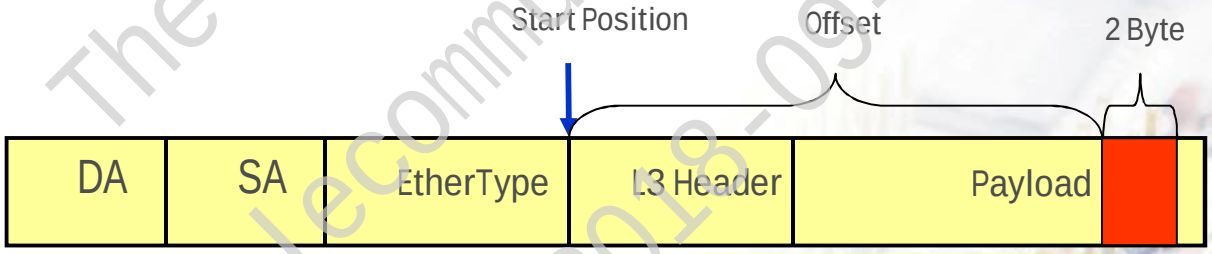
Full Match Flow Table – Template (Match Field)

- PIF (Protocol Independent Forwarding) perspective in OpenFlow v2.0.0
  - 5 Pre-defined and 5 User-defined Templates



Full Match Flow Table – User Defined Match Field

- PIF (Protocol Independent Forwarding) perspective in OpenFlow v2.0.0
- To locate the packet characteristic not supported in current match field
- 14 User Defined Match Field
  - Each field is 16 bits width
  - Used by user-defined template
- 7 Starting Position Options
  - Raw packet (Start after SFD, begin with DA)
  - LLC packet (Start with length “0000-05FF”)
  - L3 packet (Start after EtherType field，VLAN tag is bypassed)
  - ARP/RARP packet (Start from ARP/RARP header)
  - IP packet (Start from IPv4/IPv6 header)
  - IP payload (Start from IP payload，also means start of layer 4 packet)
  - L4 payload (Start after TCP/UDP header，ICMP/IGMP is not supported)



L2 Optimized Flow Table

L2 Optimized Flow Table

- 16K Hash-based flow entries (resided in two memory blocks)
  - ✓ Physically combo with 32K entries L2 table
  - ✓ Each flow entry occupies two physical L2 entries

- Match Field can be one of below
  - ✓ (VID, SA) and (0, SA)
  - ✓ (VID, SA) and (VID, DA)
  - ✓ (VID, SA) and (0, DA)
  - ✓ (0, SA) and (VID, DA)
  - ✓ (0, SA) and (0, DA)
  - ✓ (VID, DA) and (0, DA)
  - ✓ (DA, SA) and (VID, SA)
  - ✓ (DA, SA) and (0, SA)
  - ✓ (DA, SA) and (VID, DA)
  - ✓ (DA, SA) and (0, DA)
  - ✓ (IP_PROTO,SIP,DIP,L4_SPORT,L4_DPORT) and (VID,SIP,DIP,L4_SPORT,L4_DPORT)

- Match field Metadata(6-bit) comparison is per-entry configurable
- Per Entry supports a Hit indication flag for implementing IDLE timeout
- Each memory block can specify its own hash algorithm among two algorithms
- Applications could be
  - ✓ L2 forwarding
  - ✓ MAC-based QoS, MAC-based VLAN

L3 Optimized Flow Table

L3 Optimized Flow Table - TCAM-based

- TCAM-based flow entries
  - ✓ Physically combo with 12K L3 Prefix table
  - ✓ Entry Capacity
    - ✓ 6K IPv4 SIP/DIP entries(1 entry occupies 2 physical L3 prefix entries)
    - ✓ 6K IPv4 SIP+DIP entries(1 entry occupies 2 physical L3 prefix entries)
    - ✓ 4K IPv6 SIP/DIP entries(1 entry occupies 3 physical L3 prefix entries)
    - ✓ 2K IPv6 SIP+DIP entries(1 entry occupies 6 physical L3 prefix entries)

- Match Field has below choice
  - ✓ (Metadata, SIP) and (Metadata, DIP)
  - ✓ (Metadata, SIP, DIP) and (Metadata, SIP)
  - ✓ (Metadata, SIP, DIP) and (Metadata, DIP)

- Match Field can be mask off separately per-entry based
- Per Entry supports a Hit indication flag for implementing IDLE timeout
- Support entry clearance and movement by H/W
  - Offload CPU computing resource

L3 Optimized Flow Table - Hash-based

- Hash-based flow entries (resided in two memory blocks)
  - ✓ Physically combo with 12K L3 Host table
  - ✓ Entry Capacity
    - ✓ 6K IPv4 SIP/DIP entries(1 entry occupies 2 physical L3 prefix entries)
    - ✓ 6K IPv4 SIP+DIP entries(1 entry occupies 2 physical L3 prefix entries)
    - ✓ 4K IPv6 SIP/DIP entries(1 entry occupies 3 physical L3 prefix entries)
    - ✓ 2K IPv6 SIP+DIP entries(1 entry occupies 6 physical L3 prefix entries)

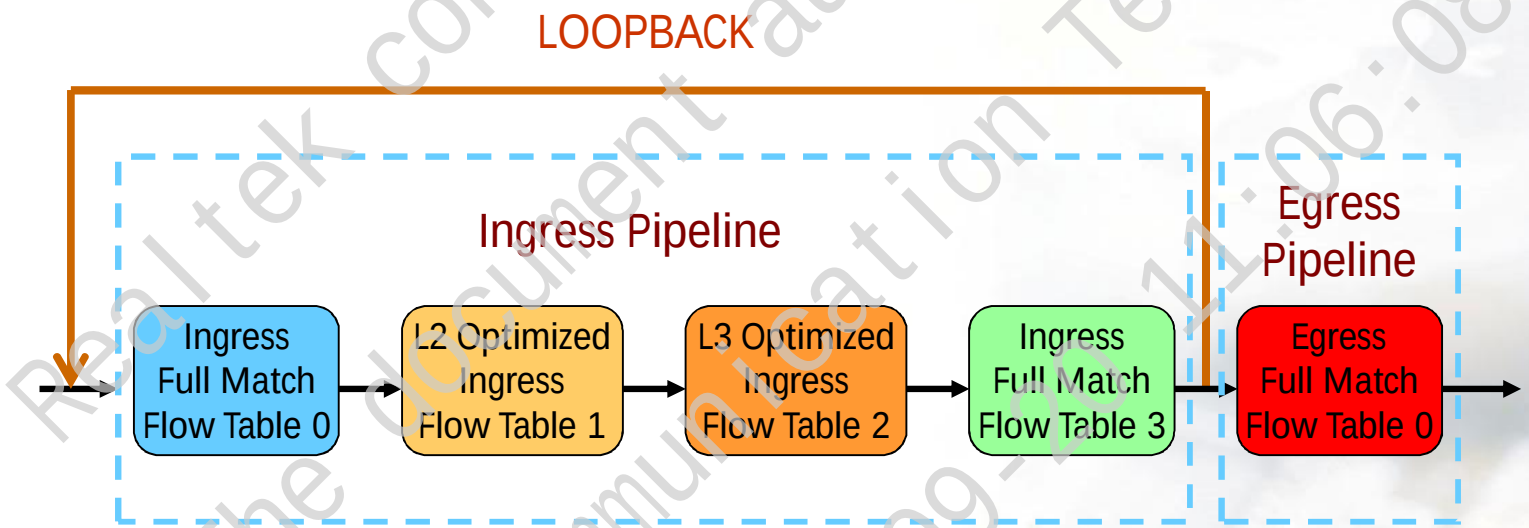
- Match Field has below choice
  - ✓ (SIP) and DIP
  - ✓ (SIP, DIP) and SIP
  - ✓ (SIP, DIP) and DIP

- Match Field Metadata(12-bit) comparison is per-entry configurable
- Per Entry supports a Hit indication flag for implementing IDLE timeout
- Each memory block can specify its own hash algorithm among two algorithms
- Applications
  - ✓ L3 Routing
  - ✓ IP-based QoS, IP-Subnet-based VLAN

Extensible Flow Table Stage

Extensible Ingress Flow Table Stage

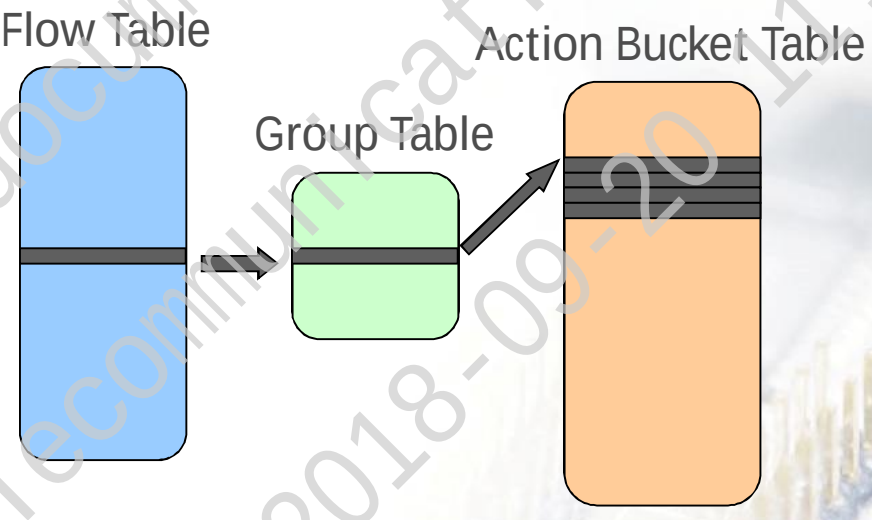
- Ingress Flow Table stage can be extended through LOOPBACK operation
- Up to 64 loopback times ➔ 256 ingress flow tables
- Maximum loopback bandwidth is 10G
- In addition to extend flow table stage



Group and Meter Table

Group Table

- 2048 group entries
- 8192 action buckets
- Group types
  - ✓ All → Multicast Forwarding/Routing
  - ✓ Select → ECMP
  - ✓ Indirect → Unicast Routing
  - ✓ Fast FailOver → change forwarding w/o conduct controller
- A group entry can contain up to 128 action buckets
- Group action only supported in Full Match and L3 Optimized Flow Tables



Meter Table

- 512 Dual-Band meters shared with ACL
- Rate can be PPS or BPS(unit: 16kbps)
- Band Type
  - ✓ Drop
  - ✓ DSCP Remark (RFC 2475 DiffServ)
- Hierarchical Metering/QoS

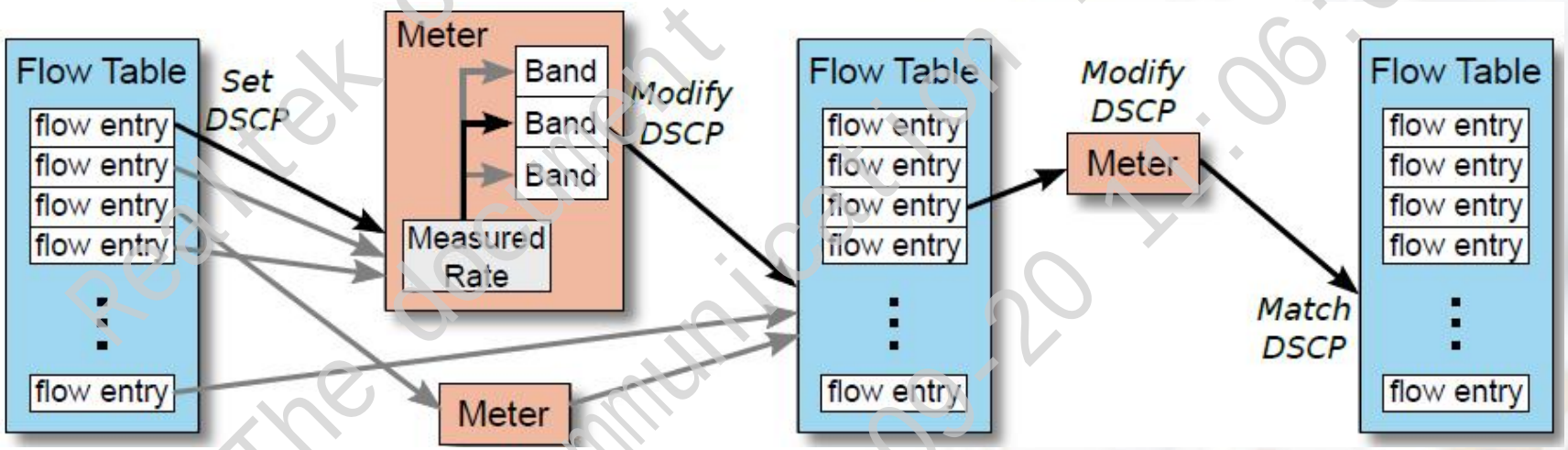


Table Configuration

- Per Flow Table supports a Table Miss action
  - ✓ Drop (default)
  - ✓ Trap
  - ✓ Forward to next table
  - ✓ Execute current action set

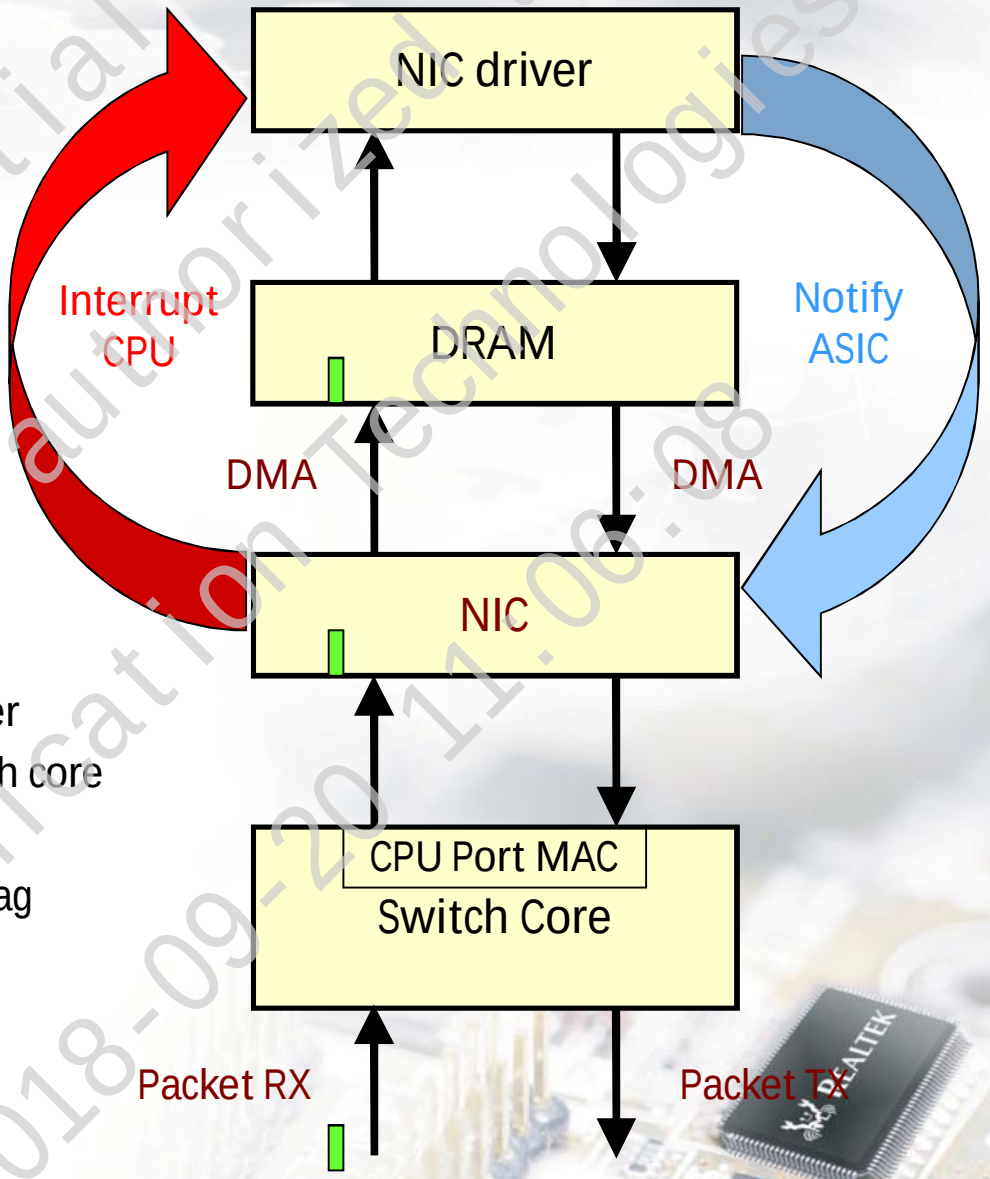
- Table-based Counters
  - ✓ Packet Lookup Counter
  - ✓ Packet Matches Counter

REALTEK
Realtek Semiconductor Corp.

NIC

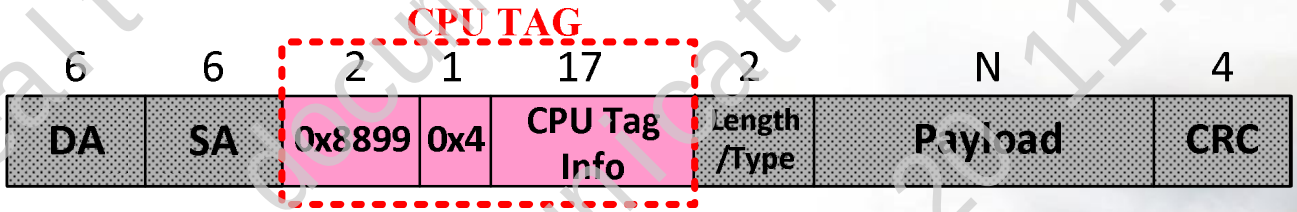
NIC (Network Interface Controller)

- NIC only for embedded CPU
- NIC RX (Switch Core->CPU)
  - Switch core forward packet to CPU port
  - NIC DMA the packet to DRAM
  - NIC interrupts CPU
  - CPU processes packet within ISR
  - 32 Rx queues mapped to CPU port MAC
  - Deliver packet info with CPU RX tag
- NIC TX(CPU->Switch Core)
  - CPU writes packet to DRAM
  - CPU notifies NIC through writing a register
  - NIC DMA the packet from DRAM to switch core
  - 2 Tx queues (High/Low strict-priority)
  - Deliver packet specific info with CPU TX tag



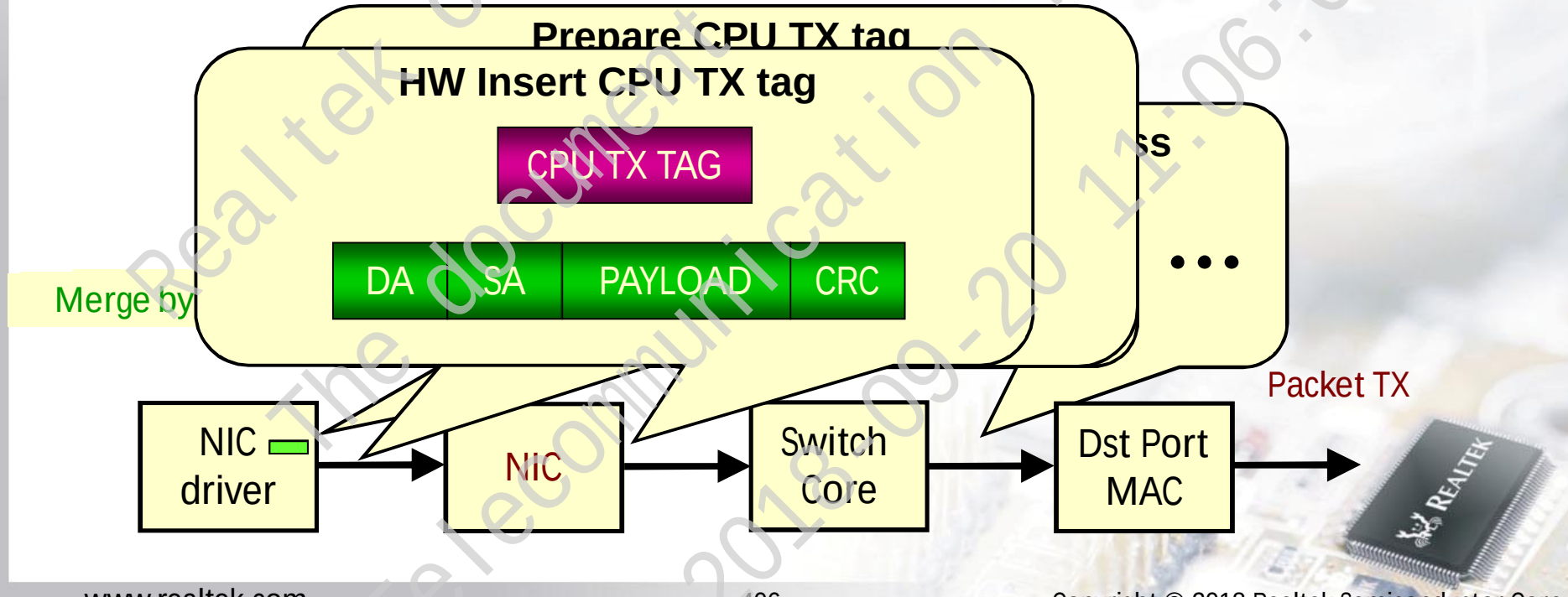
Proprietary CPU Tag

- Only CPU port (Port 56) can recognize this tag
- CPU Tag (Total 20 Bytes)
  - 2 byte PID + 1 byte protocol id (0x4) + 17 byte content
  - Default CPU tag PID = 0x8899 (configurable)
- Two formats
  - CPU TX Tag
  - CPU RX Tag



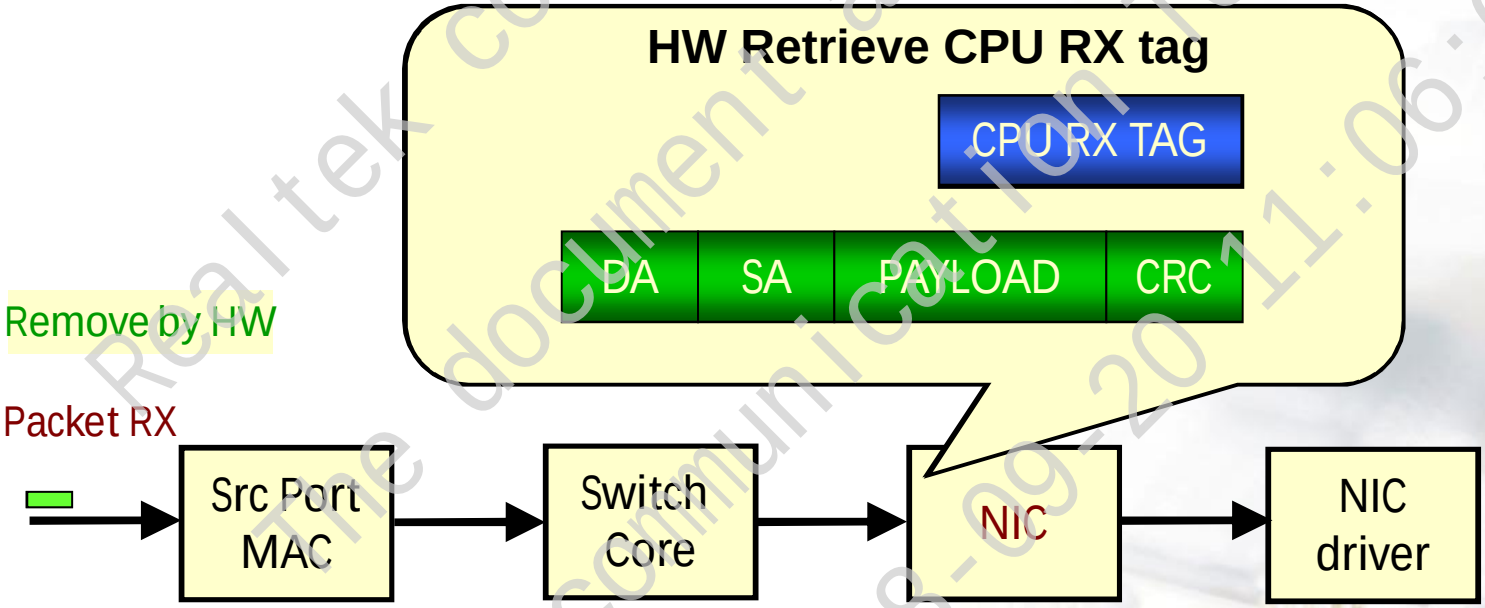
CPU Transmission Tag

- CPU TX tag
  - Control switch core behavior
  - TX packet carried without CPU tag
    - Go through normal table lookup process for deciding the destination port
  - TX packet carried with CPU tag can
    - Embedded in packet (External CPU should adopt this way)
    - Separate Packet and CPU tag in two different buffers and NIC will merge them



CPU Reception Tag

- CPU RX tag
  - Provides the extra info from switch core
  - Attached by switch core
  - Options to attach CPU RX Tag for forwarded and trapped packet respectively
  - CPU RX tag can be processed by
    - Internal CPU: Hardware extract the CPU tag to NIC descriptor
    - External CPU: Software has to retrieve Rx tag info and recover original packet



CPU TX Tag (1/3)

Field Name	Bits	Description
FWD_TYPE	4	0: follow switch forwarding 1: use SW_UNIT and DPM as the destination port mask(Physical) 2: use SW_UNIT and DPM as the destination port mask (Logical) 3: use DPM as trunk ID ..... 11: notify all CPU, including local CPU (all CPUs in stacking system would receive)
SW_UNIT	4	If FWD_TYPE is 1 or 2 or 6 or 7, this field indicates destination unit ID
DPM	56	DPM is used as portmask when FWD_TYPE is 1, 2, 4, 5 DPM is used as trunk ID when FWD_TYPE is 3
ACL_ACT	1	0 : Don't apply any action of ACL rules 1 : apply the ACL effect
AS_QID	1	0: the output queue is decided by lookup process. 1: CPU assign the output queue directly.
QID	5	Assigned QID.
CNGST_DROP	1	0: Don't drop it when congestion. 1: Can be dropped when the destination port is congestion.

CPU TX Tag (2/3)

Field Name	Bits	Description
DM_PKT	1	0b0: This is NOT an ETH-DM packet. 0b1: Notifies TX Modifier to fill the TX Timestamp field based-on OpCode.
DG_PKT	1	0: It's not a dying gasp packet 1: It's a dying gasp packet
BP_FLTR	1	0: filter by Egress OAM mux & Port Isolation & Mirror Isolation 1: bypass Egress OAM mux & Port Isolation & Mirror Isolation
BP_STP	1	0: filter by egress STP port state 1: bypass egress STP
BP_VLAN_EGR	1	0: filter by Egress VLAN 1: bypass Egress VLAN filtering
ALE_AS_TAGSTS	1	0 : keep TAG status (do not any modification) 1 : follow switch core decision (i.e., VLAN table's tag/untag set)
L3_ACT	1	0: Don't apply the L3 effect 1: Apply L3 logic
FWD_VID_SEL	1	0: follow the inner tag 1: follow the outer tag

CPU TX Tag (3/3)

Field Name	Bits	Description
FWD_VID_EN	1	0: Forwarding VID is determined by switch core 1: force FWD_VID
FWD_VID	12	Forwarding VID (The field is used when FWD_VID_EN=1)
ORI_TAGIF_EN	1	Decide whether reference ORI_ITAGIF and ORI_OTAGIF
ORI_ITAGIF	1	Indicates whether this packet contains the inner tag
ORI_OTAGIF	1	Indicates whether this packet contains the outer tag
SRC_FILTER_EN	1	If SRC_FILTER_EN is 0b1, SPN should be filtered from outgoing portmask
SP_IS_TRK	1	(The field is used when SRC_FILTER_EN=1) 0: SPN is not trunk ID 1: SPN is trunk ID
SPN	10	(The field is used when SRC_FILTER_EN=1) If SP_IS_TRK=0, SPN[9:6] indicate source unit ID SPN[5:0] indicate the source port ID If SP_IS_TRK=1, SPN[6:0] indicate the trunk ID

CPU RX Tag (1/3)

Field Name	Bits	Description
PROTO	8	Define the CPU tag. Value is 0x04
PHY_RX_PORT	6	Local Physical RX Port
OF_LU_MIS_TBL_ID	2	The table ID when OpenFlow table lookup miss
ACL_OF_HIT	2	Indicates whether ACL or OpenFlow hit
OF_TBL_ID	2	OpenFlow hit table ID
IS_TRK	1	To indicate whether the packet is received from trunk port.
TRK_ID	7	If IS_TRK=1, this field indicates the trunk ID
L2_ERR_PKT	1	L2 CRC error packet
L3_ERR_PKT	1	L3 chksum error packet
ATK_TYPE	5	Indicates that type of attack that this packet matched
QID	5	The queue id that packet queued
SPN	10	Source physical (UNIT, port ID)
ORI_ETAGIF	1	Asserted if the packet contains the ECID tag when received on an ingress port
IDX	15	Used for ACL to indicate the entry index or Egress Sflow port id.

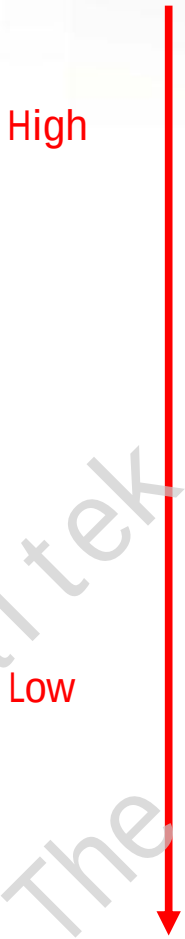
CPU RX Tag (2/3)

Field Name	Bits	Description
MIR_HIT	4	Mirroring hit status for 4-sets of configuration
SFLOW	2	S-Flow sampling packet
TT_HIT	1	Asserted if tunnel terminate entry hit
TT_IDX	9	Tunnel terminate entry index
ETAGIF	1	Asserted if the packet contains the ECID tag
OTAGIF	1	Packet contains the outer tag
ITAGIF	1	Packet contains the inner tag
FWD_VID_SEL	1	Forwarding VLAN ID is from inner or outer VID (0: inner VID 1: outer VID)
FWD_VID	12	Forwarding VLAN ID
MAC_CST	1	Packet has caused the MAC constraint
DM_RXIDX or OF_LB_TIMES	6	Y.1732 ETH-DM RX Timestamp Index (0~63) or Packet loopback times.
NEW_SA	1	Packet is new SA
PMV_FORBID	1	Packet has cause the port moving forbiden rule
L2_STTC_PMV	1	Packet has caused the static L2 port moving

CPU RX Tag (3/3)

Field Name	Bits	Description
L2_DYN_PMV	1	Packet has caused the dynamic L2 port moving
OVER_SIZE	1	Packet is truncated by NIC
PORT_DATA_IS_TRK	1	Indicates if the PORT_DATA is trunk port
PORT_DATA	10	Original port this packet is learnt on L2 Table
HASH_FULL	1	L2 table hash bucket full
INVALID_SA	1	Invalid SA(MC/BC/All Zero)
REASON	6	The reason that packet to CPU

CPU RX Tag – REASON



0. Invalid
1. CPU to CPU talk
2. OpenFlow IP/MPLS TTL exception
3. OF table lookup miss
...

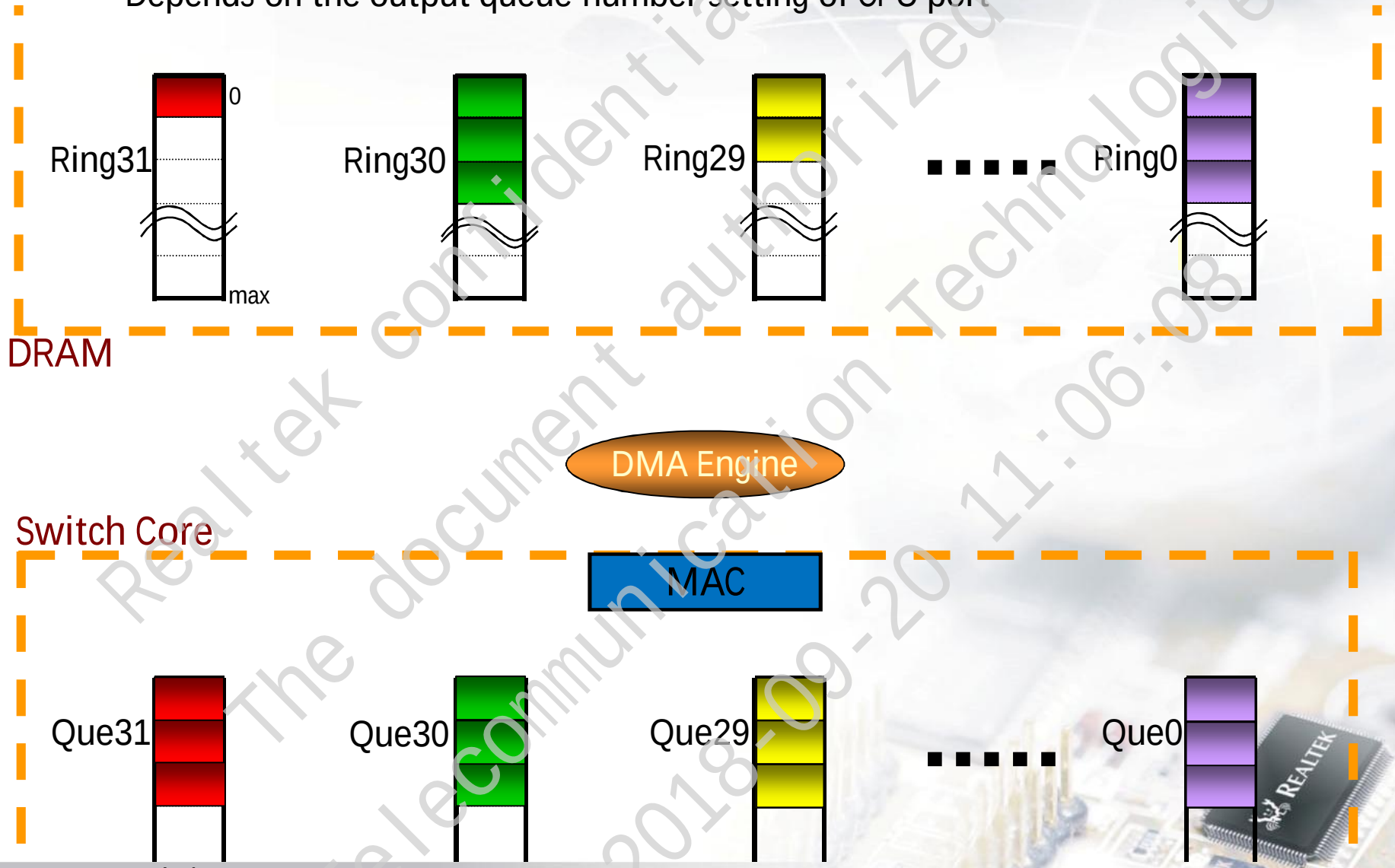
17. L2 UC/MC bridge lookup miss
18. IP4/IP6 MC bridge lookup miss
19. PTP
20. User-defined RMA

....

60. MPLS exception
61. Reserved
62. L2 Notification
63. Normal forward

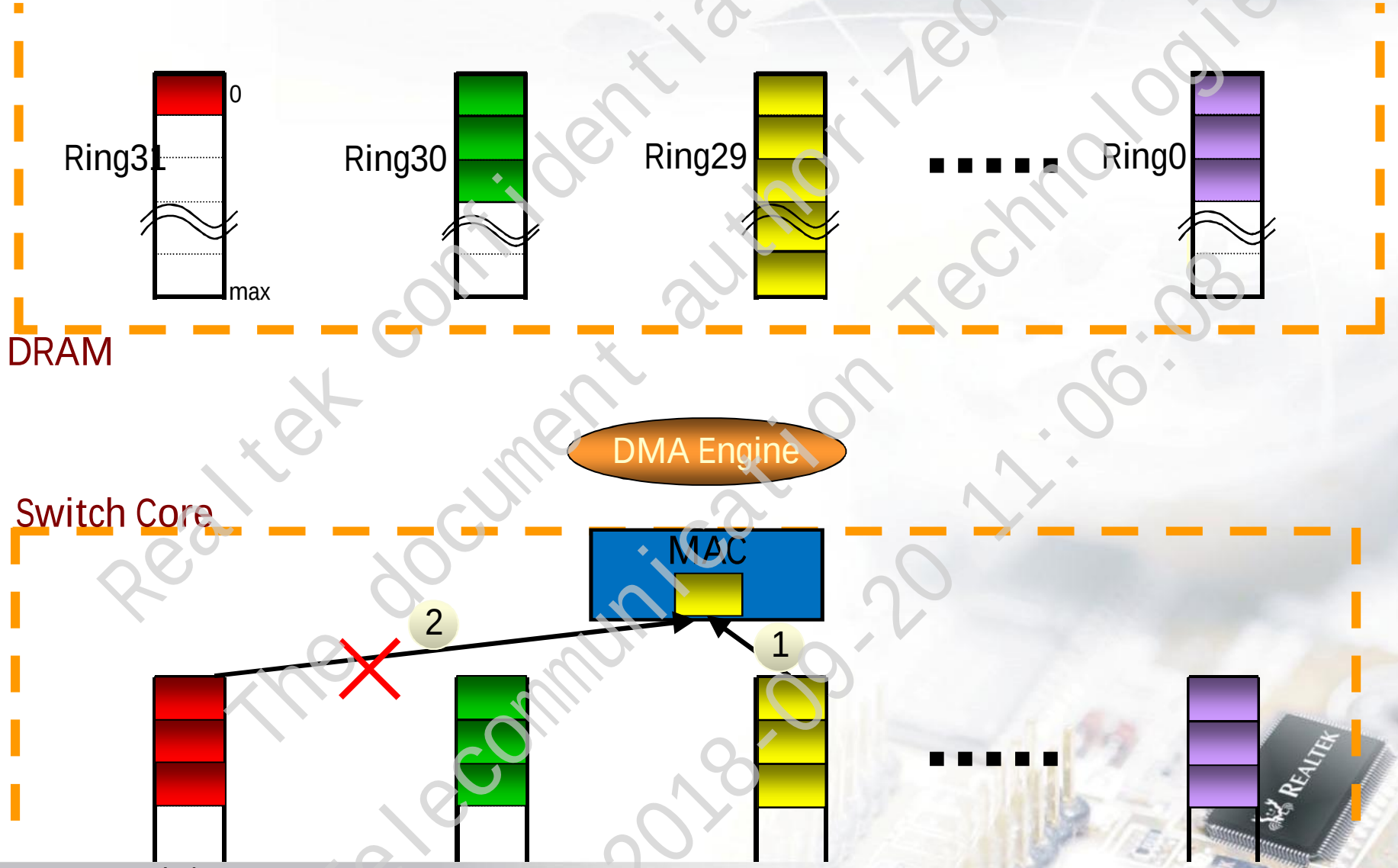
RX Descriptor Ring Buffer

- 32 descriptor rings for RX
  - Depends on the output queue number setting of CPU port



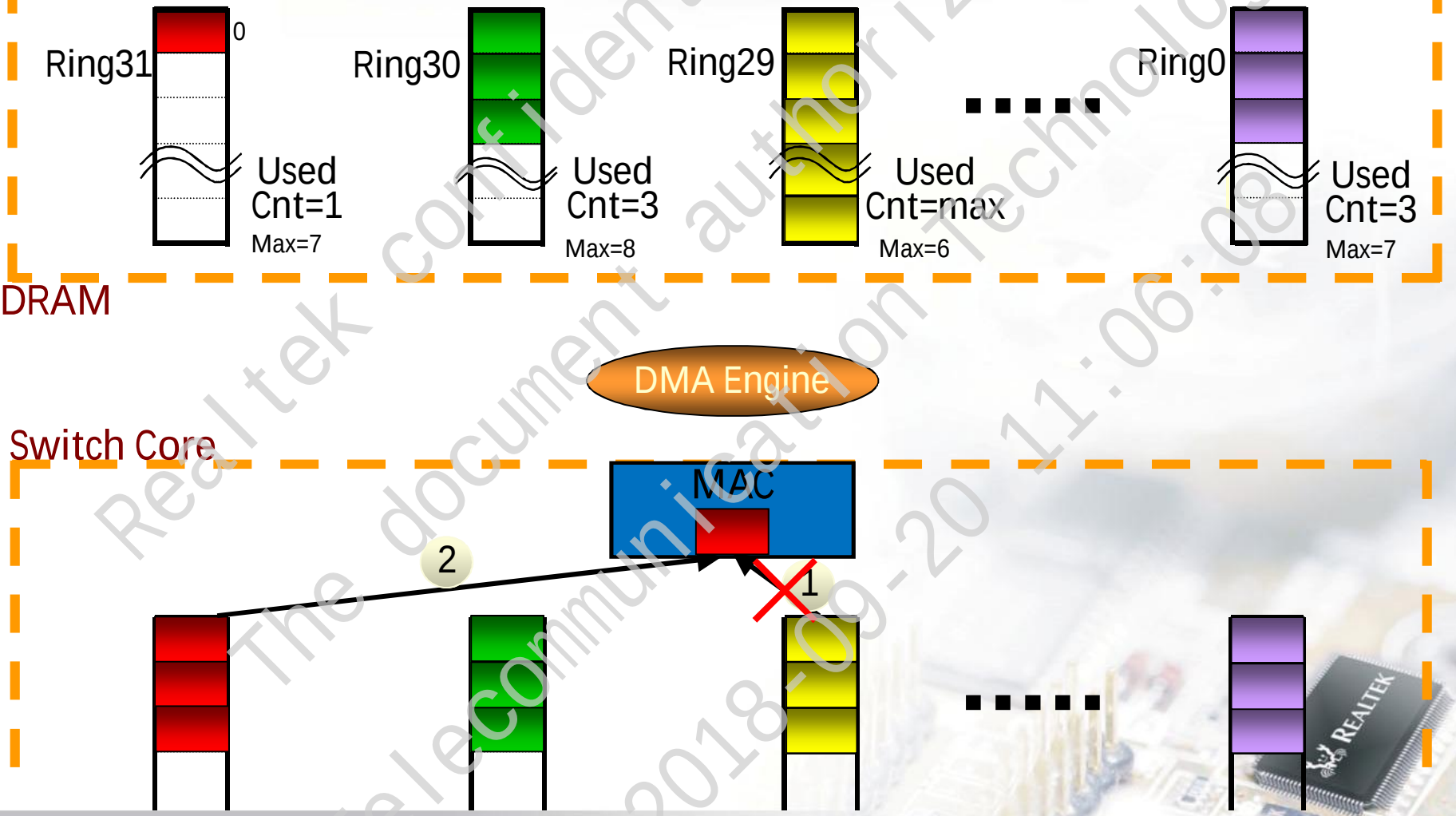
Head of Line Blocking (HOL)

- A line of packets are blocked by first packet



HOL Prevention

- ASIC checks the available descriptor count of a specific RX queue in the DRAM
  - When used descriptor count of a queue ≧ queue size ⇨ Pause that queue
  - The queue size is per-queue configurable (unit: packet)



Realtek Semiconductor Corp.

Stacking System

Overview

- 16 Unit Stacking
- 16 Stacking Port
  - Any port could be selected as stacking port
- 8 Stacking Trunk
- Multiple Stacking Topology
  - Star
  - Line
  - Ring – support Loop Prevention

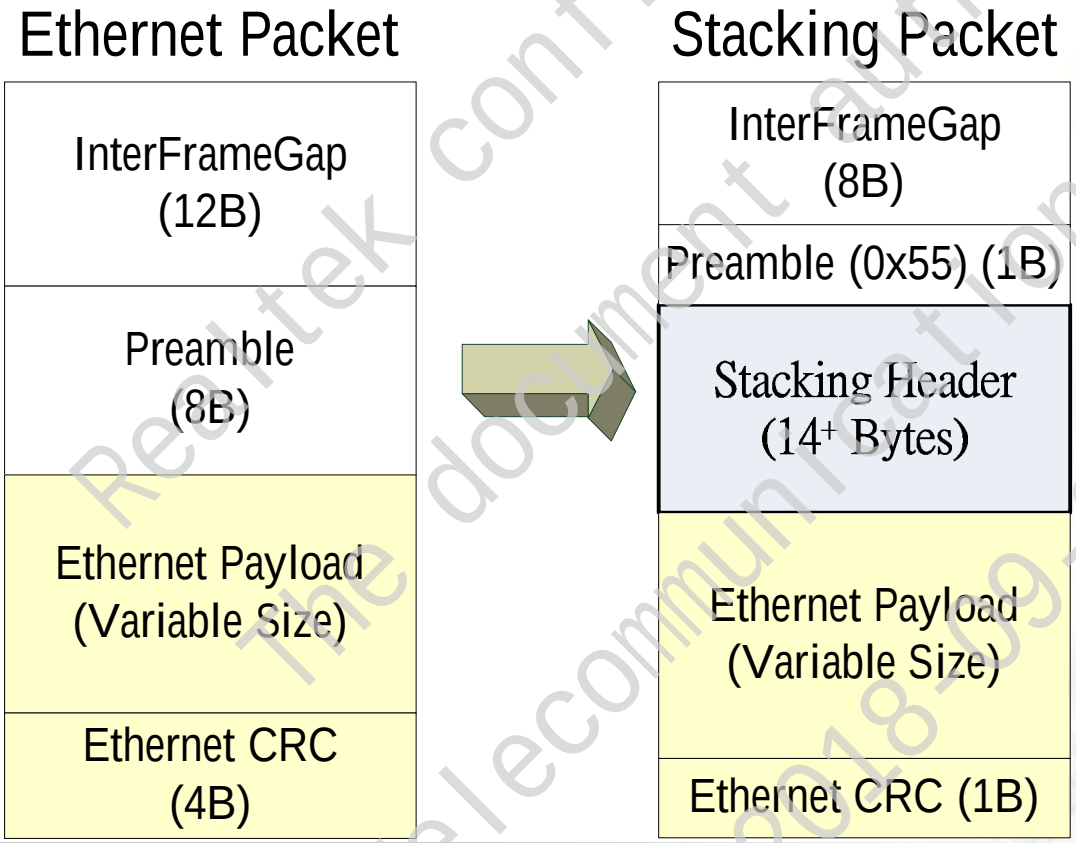
Basic Configuration

- Unit ID and Master Unit ID Configurable
- Stacking Port Selection
  - At most 16 ports can be selected as stacking ports
  - Any port can be selected as stacking port
- Unit to stacking port mapping
  - If mapped ports are trunked, load balancing is supported
  - For example
    - For a packet sent to Unit0, it would be sent to Port 52 or Port 53 by hash
    - For a packet sent to Unit1, it would be sent to Port 54 or Port 55 by hash

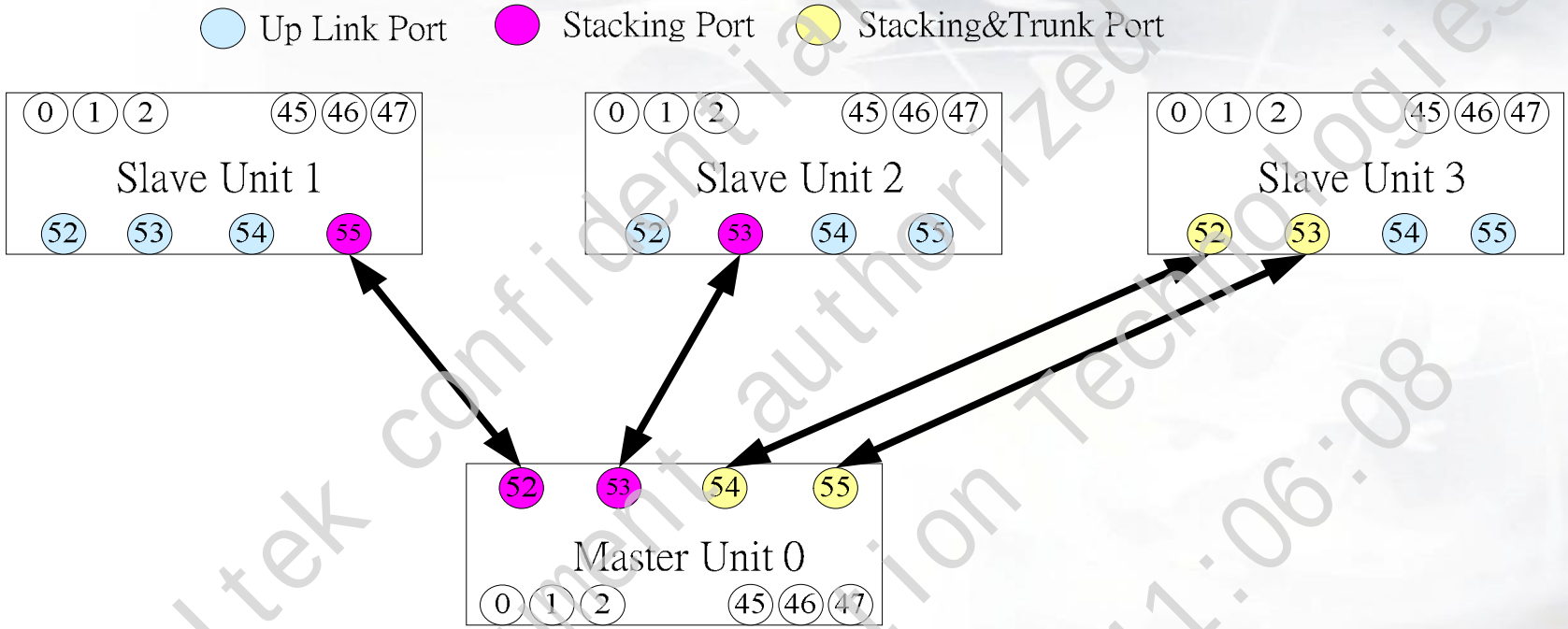
Unit ID	Stacking Port
0	52,53 (port 52,53 trunked)
1	54,55 (port 54,55 trunked)
....	
15	54,55

Stacking Frame Format

- 14 bytes header + optional header for stacking functions
  - Optional header is for unattached VLAN/ ETAG information so optional header does not increase packet total length
- IPG, preamble, and CRC length are reduced (4+7+3=14 bytes) so attached header brings 0 % bandwidth impact



Stacking Topology - Star



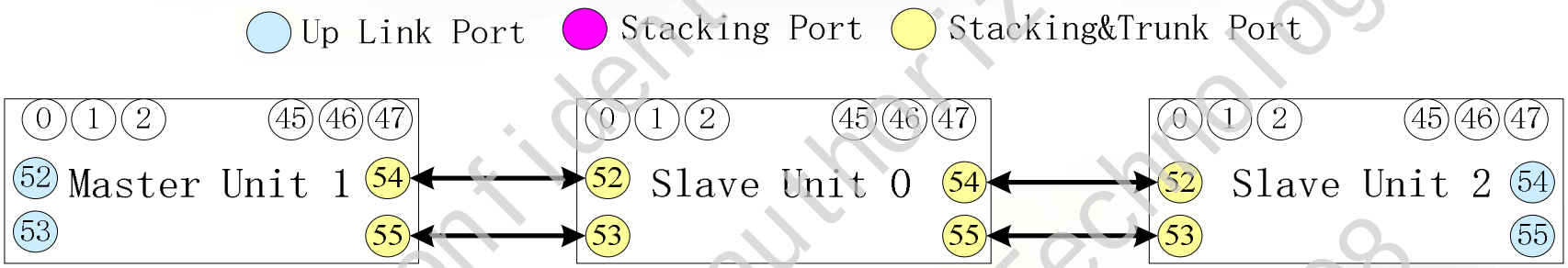
Unit 0

Unit Configuration	
Unit ID	0
Master ID	0

Port Mapping	
Unit ID	Mapped Ports
0	-
1	52
2	53
3	54,55
.....	.....
15	-

Stacking Trunk	
Trunk ID	Portmask
0	54,55
1 ~ 7	-

Stacking Topology - Line



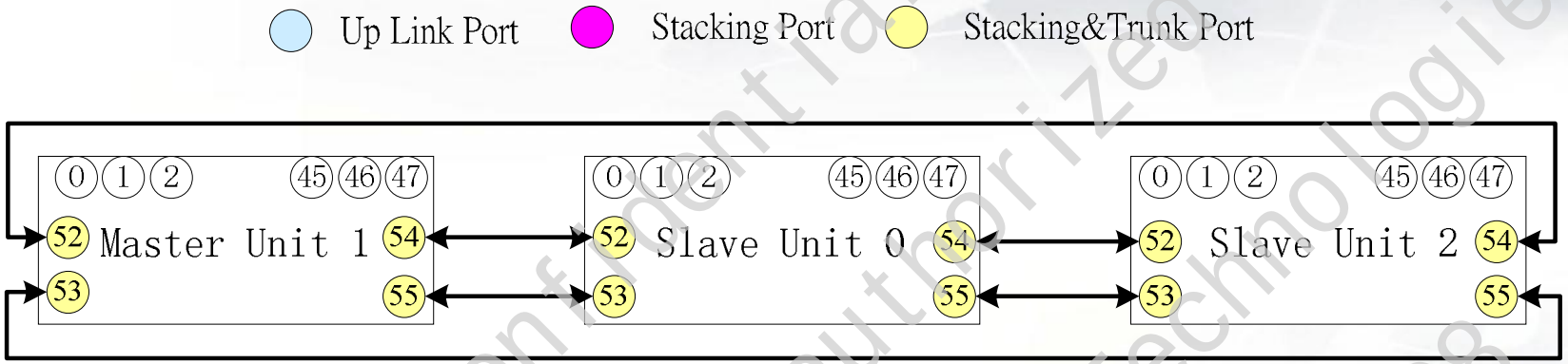
Unit 0

Unit 0 Configuration	
Unit ID	0
Master ID	1

Unit 0 Port Mapping	
Unit ID	Mapped Ports
0	-
1	52,53
2	54,55
3	-
.....	.....
15	-

Unit 0 Stacking Trunk	
Trunk ID	Portmask
0	52,53
1	54,55

Stacking Topology - Ring



Unit 1

Unit Configuration	
Unit ID	1
Master ID	1

Unit Port Mapping	
Unit ID	Ports
0	54,55
2	52,53
Other	-

Stacking Trunk	
Trunk ID	Ports
0	52,53
1	54,55

Unit 0

Unit Configuration	
Unit ID	0
Master ID	1

Unit Port Mapping	
Unit ID	Ports
1	52,53
2	54,55
Other	-

Stacking Trunk	
Trunk ID	Ports
0	52,53
1	54,55

Unit 2

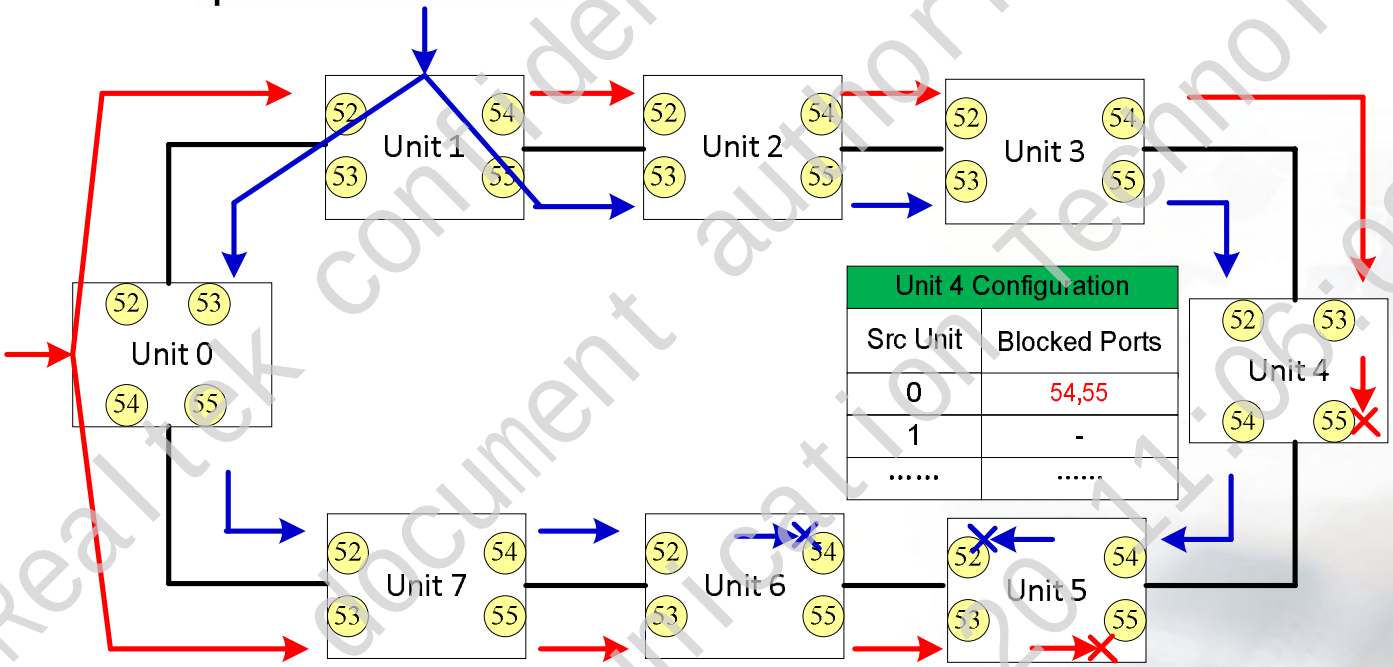
Unit Configuration	
Unit ID	2
Master ID	1

Unit Port Mapping	
Unit ID	Ports
0	52,53
1	54,55
Other	-

Stacking Trunk	
Trunk ID	Ports
0	52,53
1	54,55

Loop Prevention

1. Drop the packet which source Unit == My Unit
2. Source Unit Based Block Portmask
   - Stops unknown/ multicast/ broadcast packets looping
   - Traffic hops can be controlled and balanced



Unit 4 Configuration	
Src Unit	Blocked Ports
0	54,55
1	-
……	……

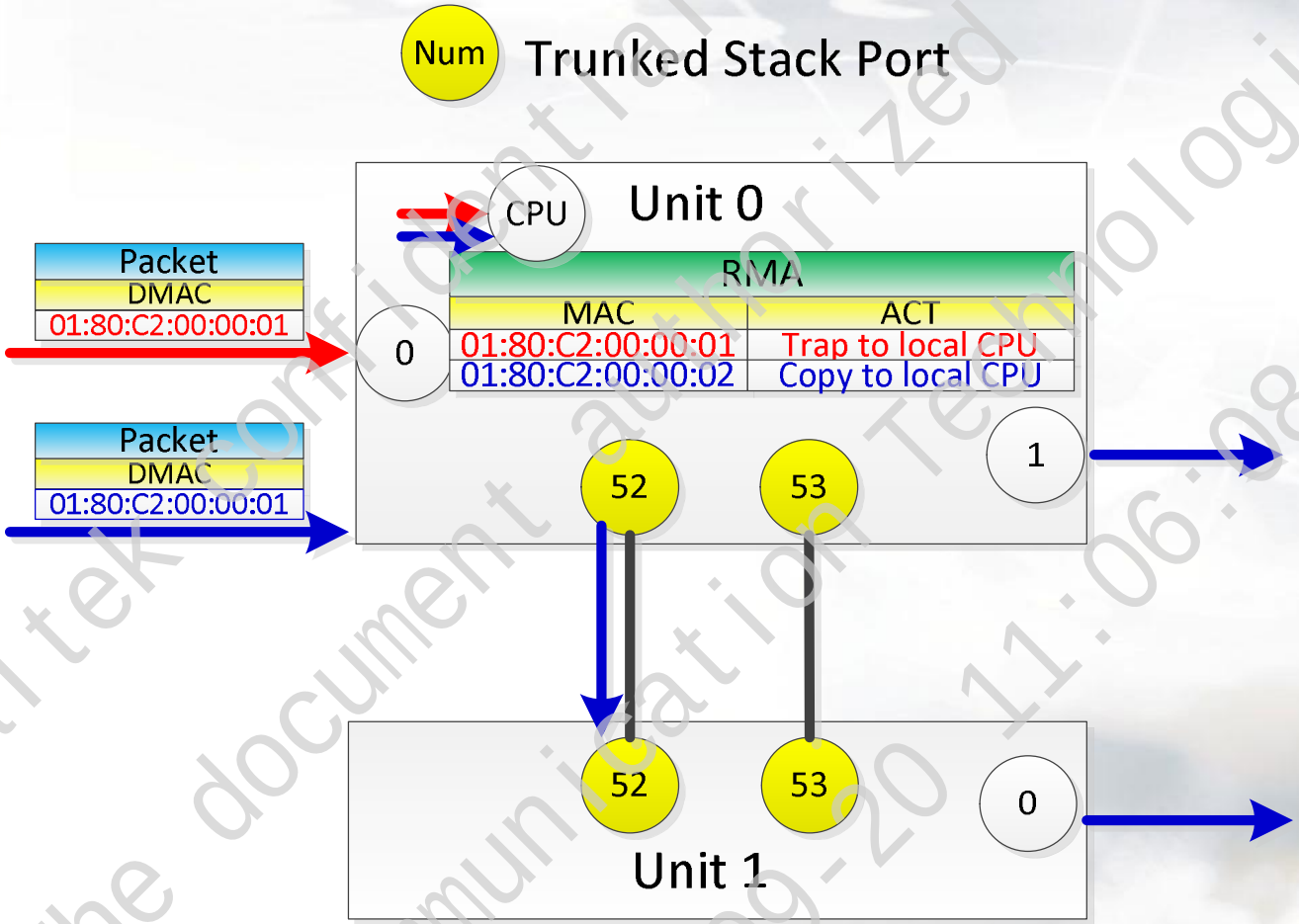
Unit 6 Configuration	
Src Unit	Blocked Ports
0	-
1	54,55
……	……

Unit 5 Configuration	
Src Unit	Blocked Ports
0	54,55
1	52,53
……	……

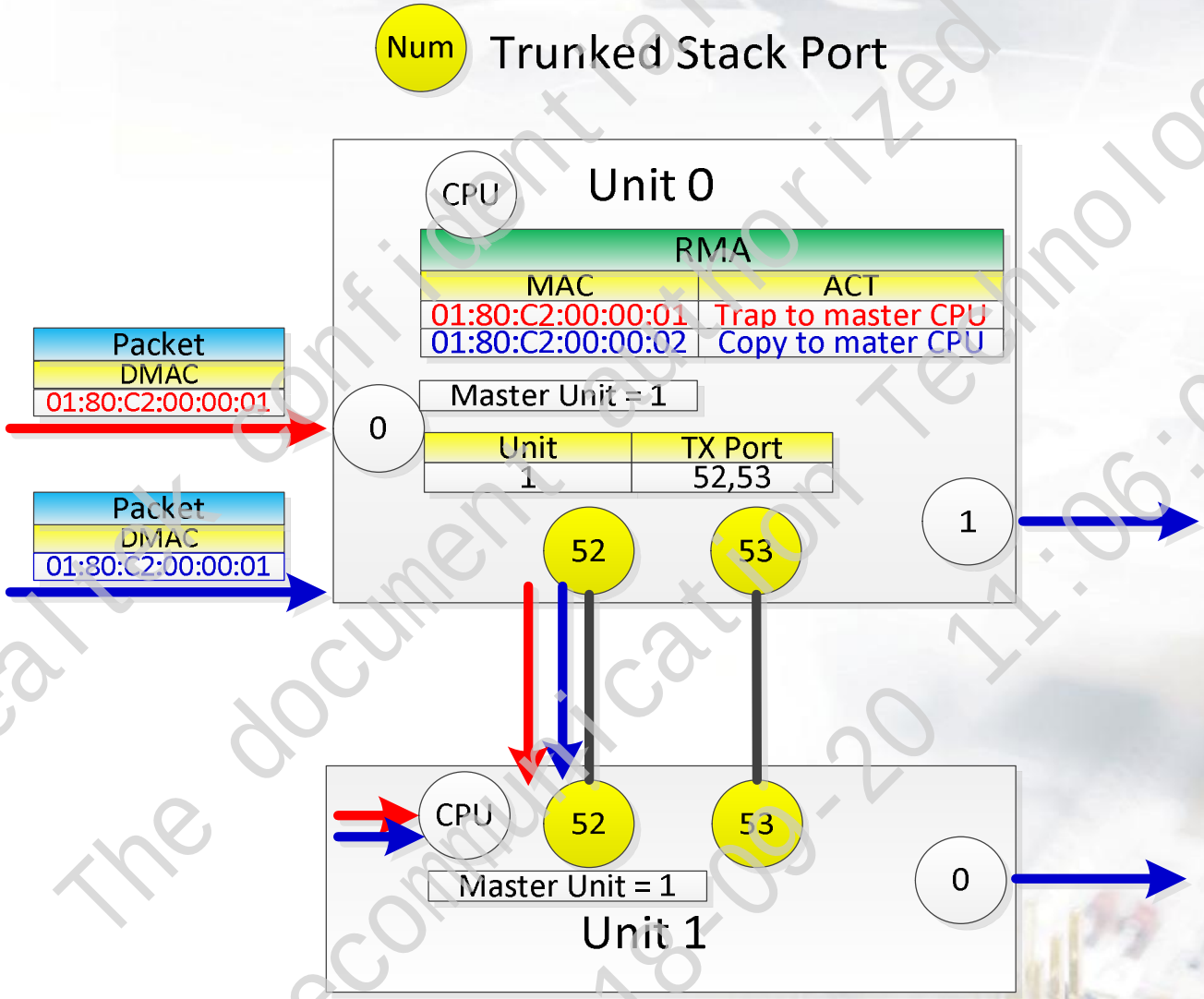
Traffic Forwarding

- Trap or Copy to Local CPU (add)
- Trap or Copy to Master CPU (add)
- Bridging
  - Know Unicast Packet
  - Know Multicast Packet
  - Unknown Packet
- Routing
  - Unicast Routing
  - Multicast Routing

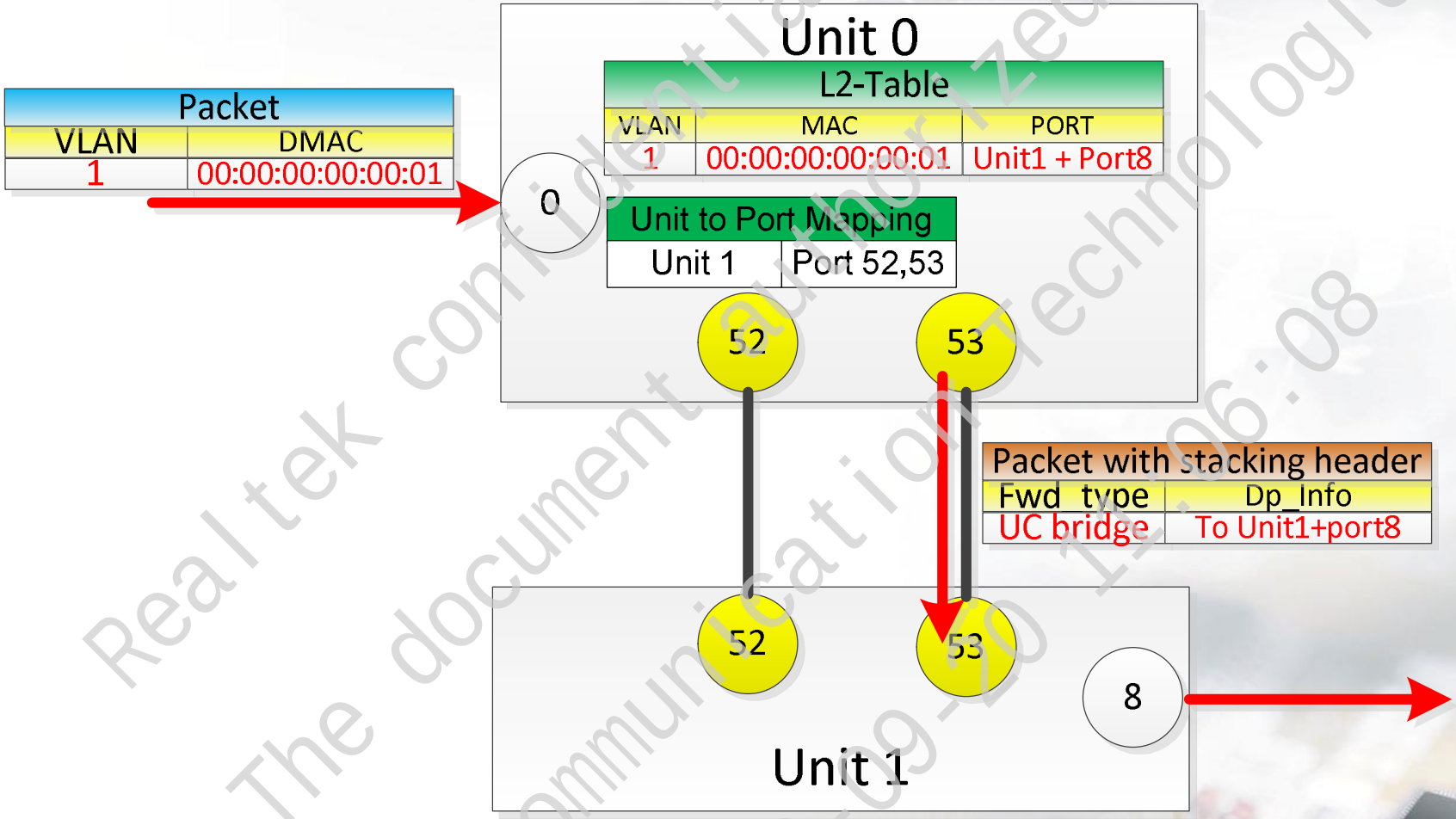
Trap/ Copy To Local CPU



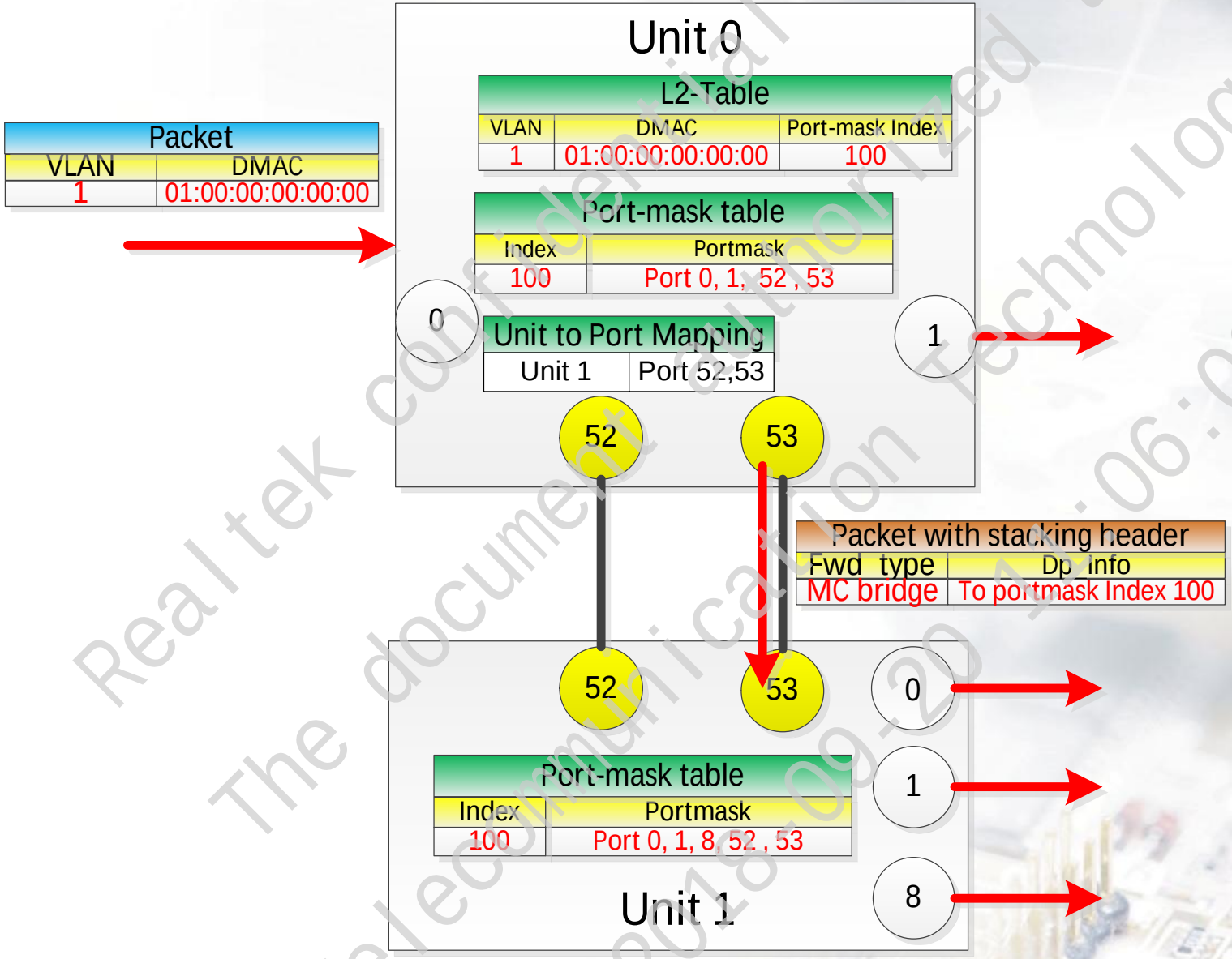
Trap/ Copy To Master CPU



Known Unicast

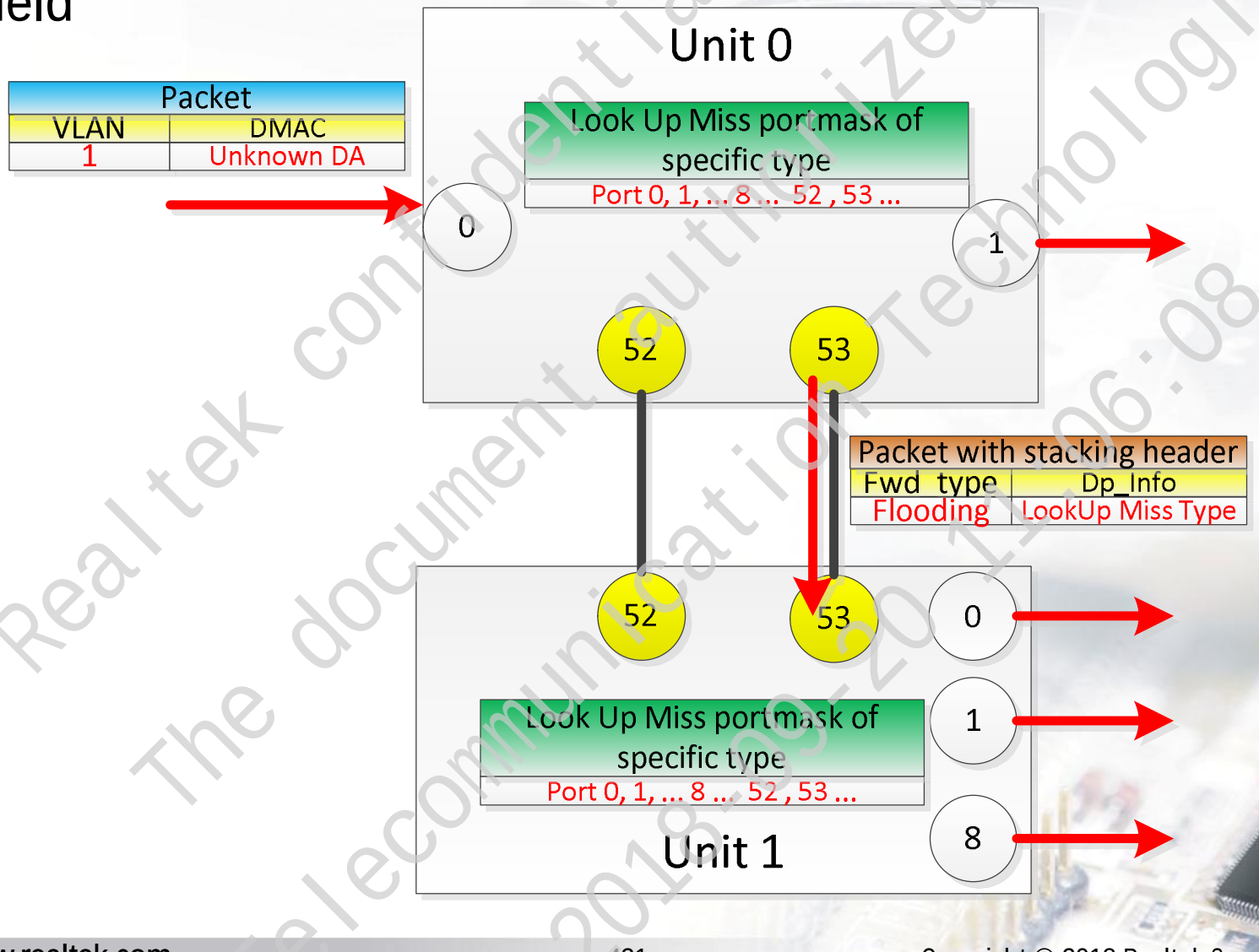


Known Multicast

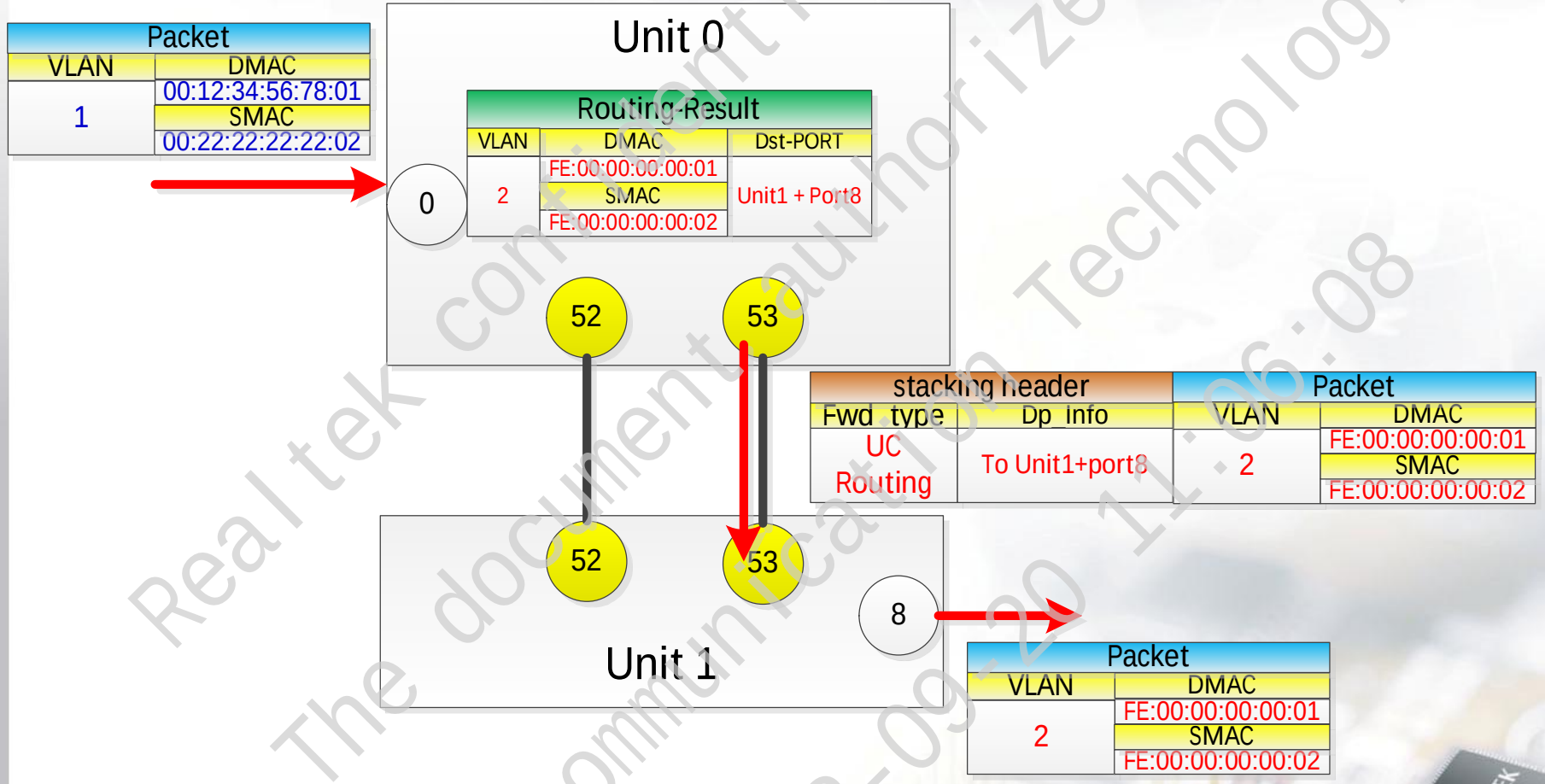


Unknown Packets

- Lookup miss packet type could be identified by stacking header field

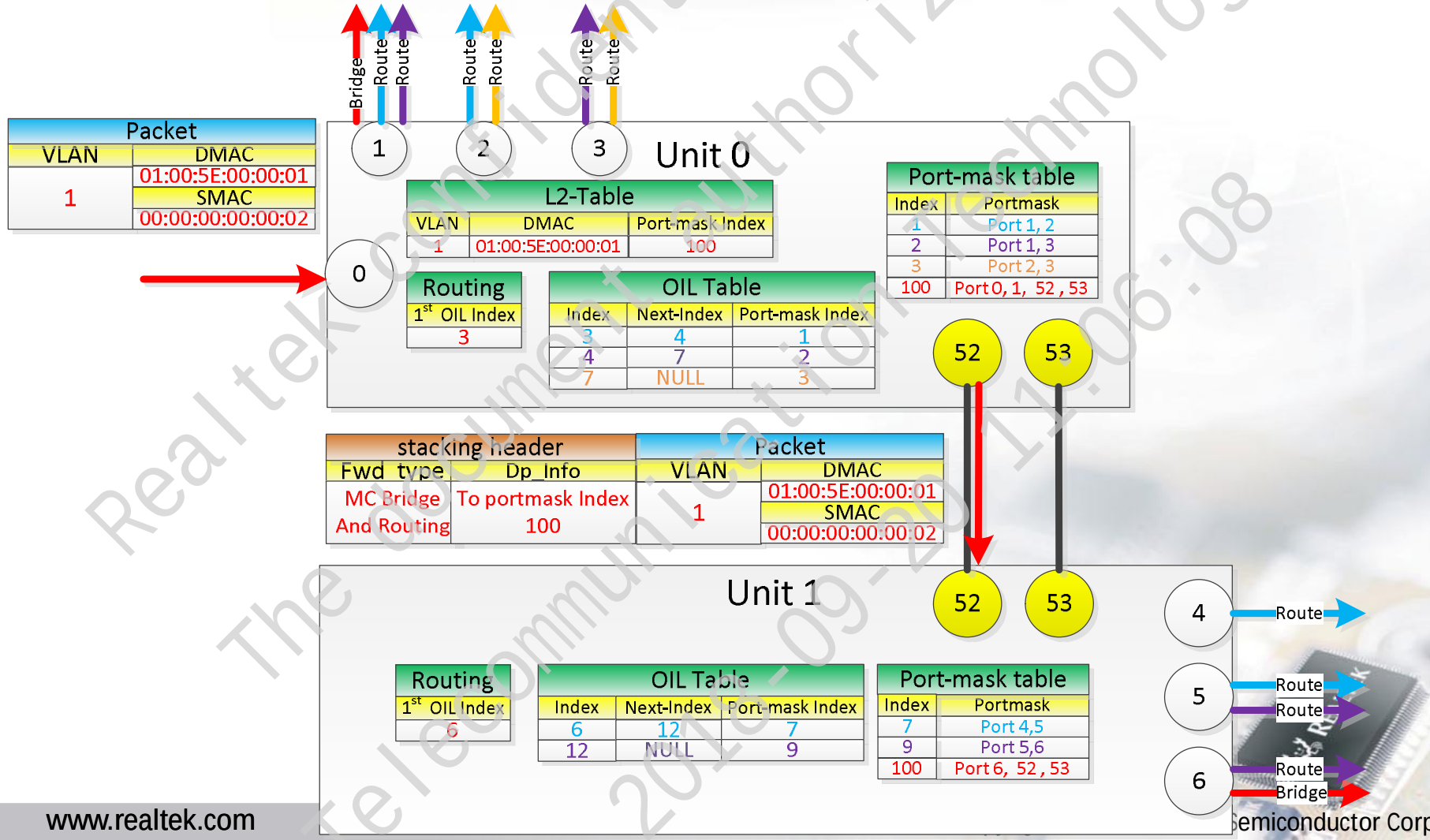


Unicast Routing



Multicast Bridging and Routing

- One packet for both multicast routing and bridging
  - First unit decides bridging multicast index and put it in stacking header
  - Each unit looks up L3 table to decide routing multicast index

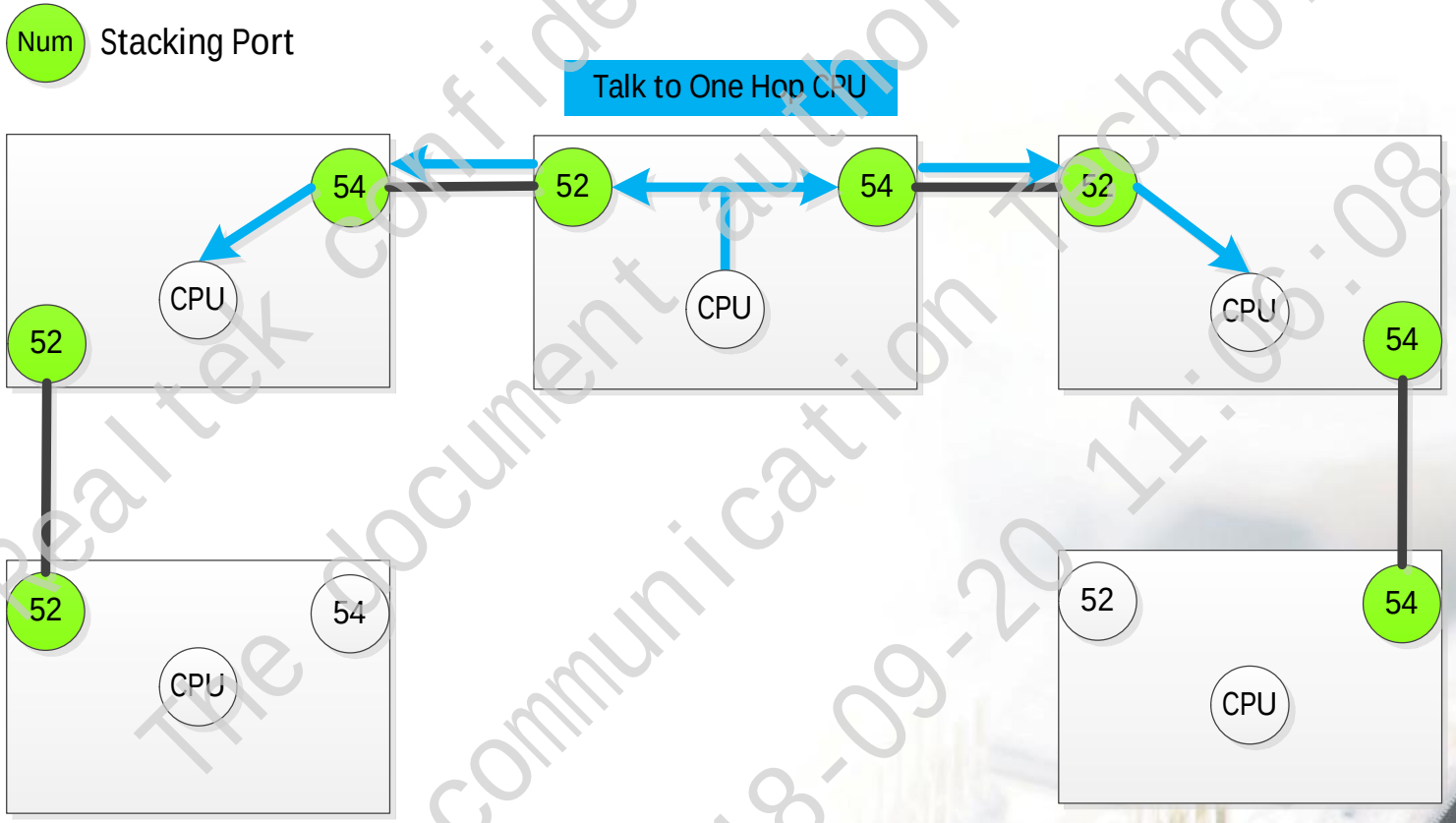


CPU Communication Packet Flow

- CPU to CPU Communication methods
  - One hop to CPU
  - Unicast to CPU
  - Broadcast to CPU

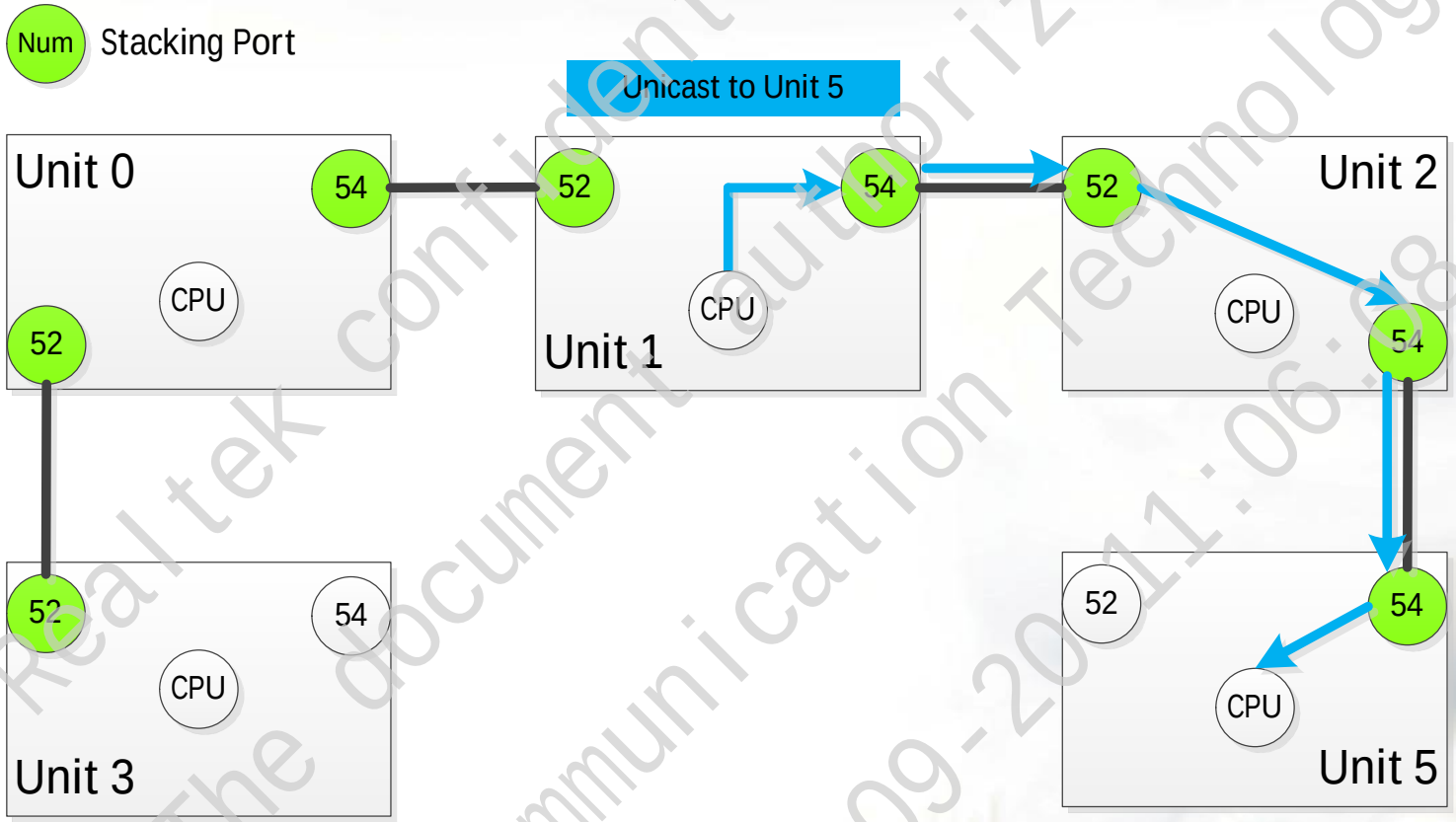
CPU Communication - One Hop to CPU

- Only send to the CPU of next hop units, used for
  - One hop hello frame
  - Initial phase communication when neighbor unit IDs are unknown



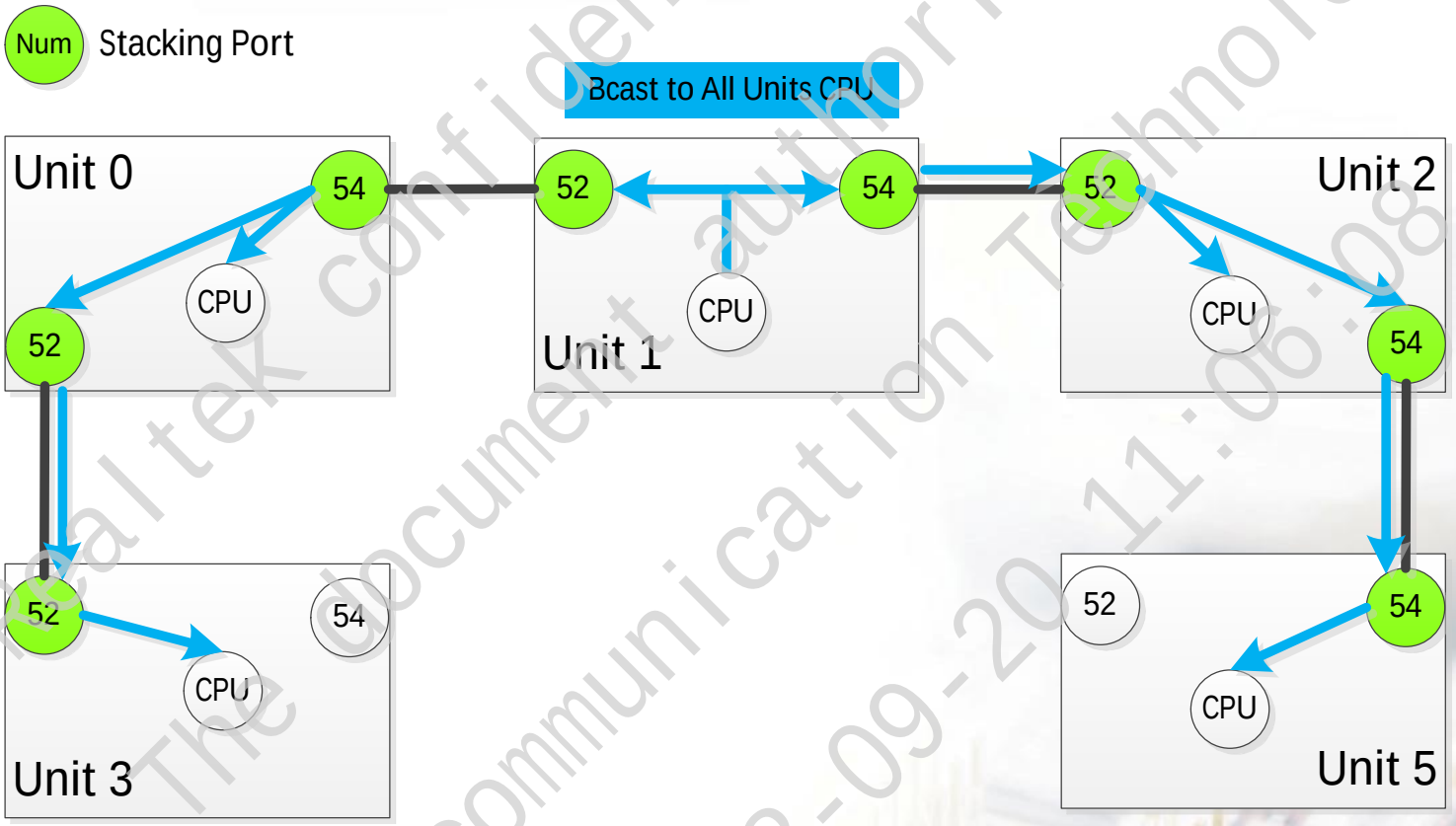
Unicast to CPU

- Communicate with a CPU of specific Unit



Broadcast to CPU

- For frames that should inform CPU of all units
- Received unit would copy the frame to CPU and forward to all stacking ports

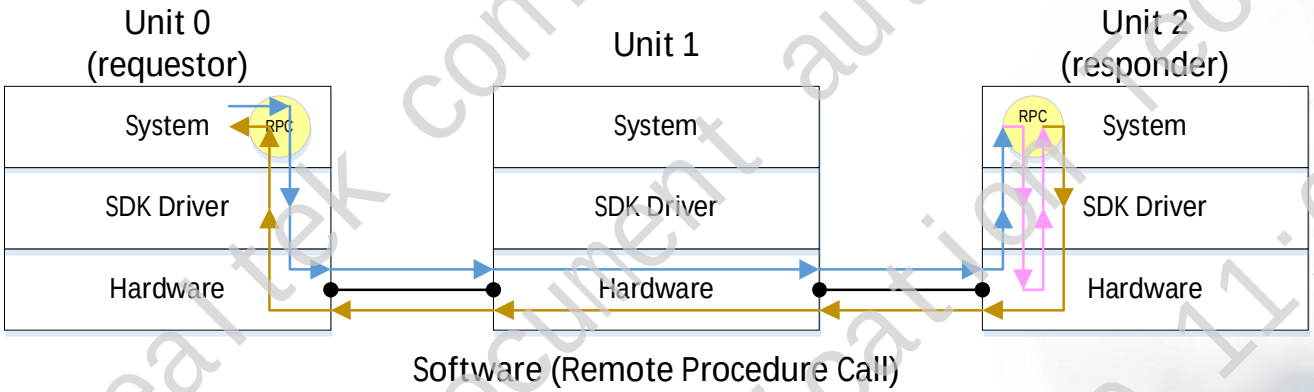


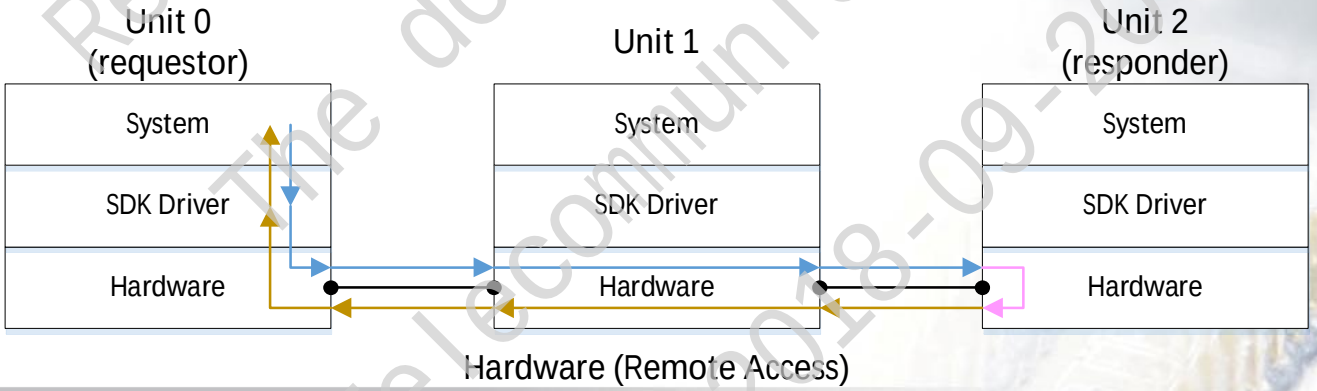
REALTEK
Realtek Semiconductor Corp.

Remote Access & Interrupt Notification

Remote Register/Table/PHY Access

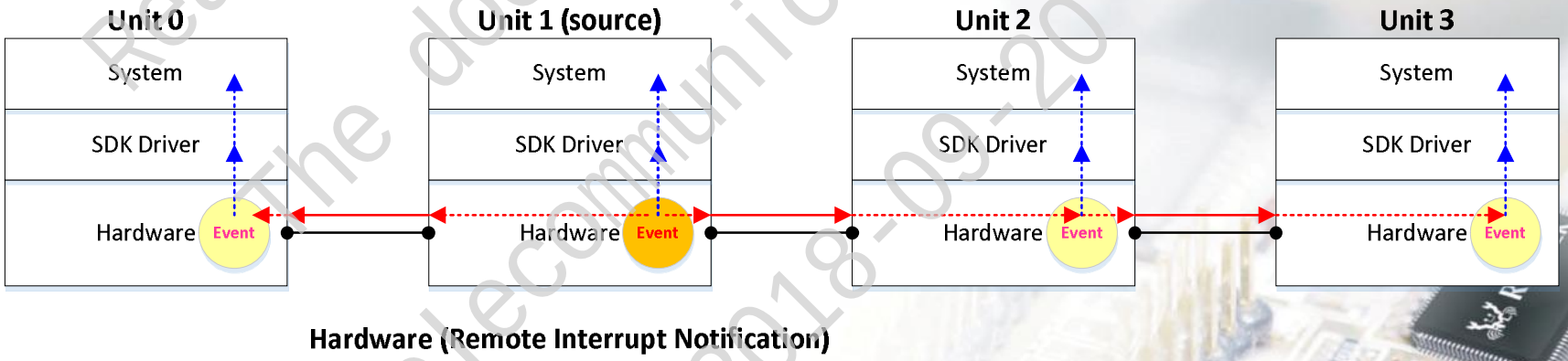
- Hardware-supported stacking remote access function
  - Register access
  - Table entry access
  - PHY access
- Higher performance than software-implemented RPC





Remote Interrupt Notification

- H/W automatically generate event notification frame to all units
  - Propagate as broadcast
  - Shorter propagation time
  - Allow each unit to be aware the interrupt events of other units
- Supported interrupt events
  - Per port Link status
  - Per CCM instance timeout
- Global configuration
  - Enable/Disable

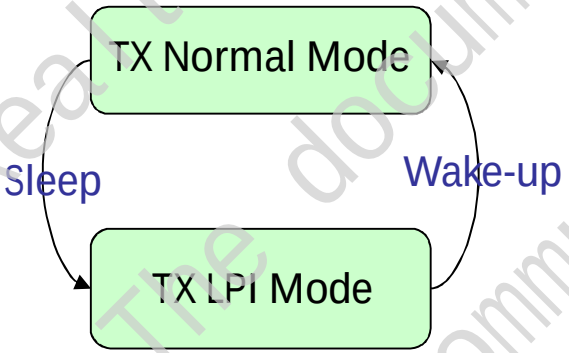


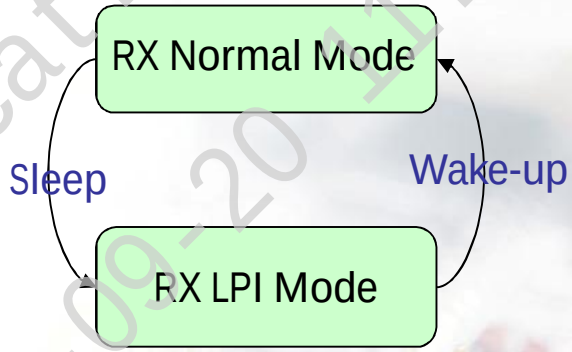
Realtek Semiconductor Corp.

EEE

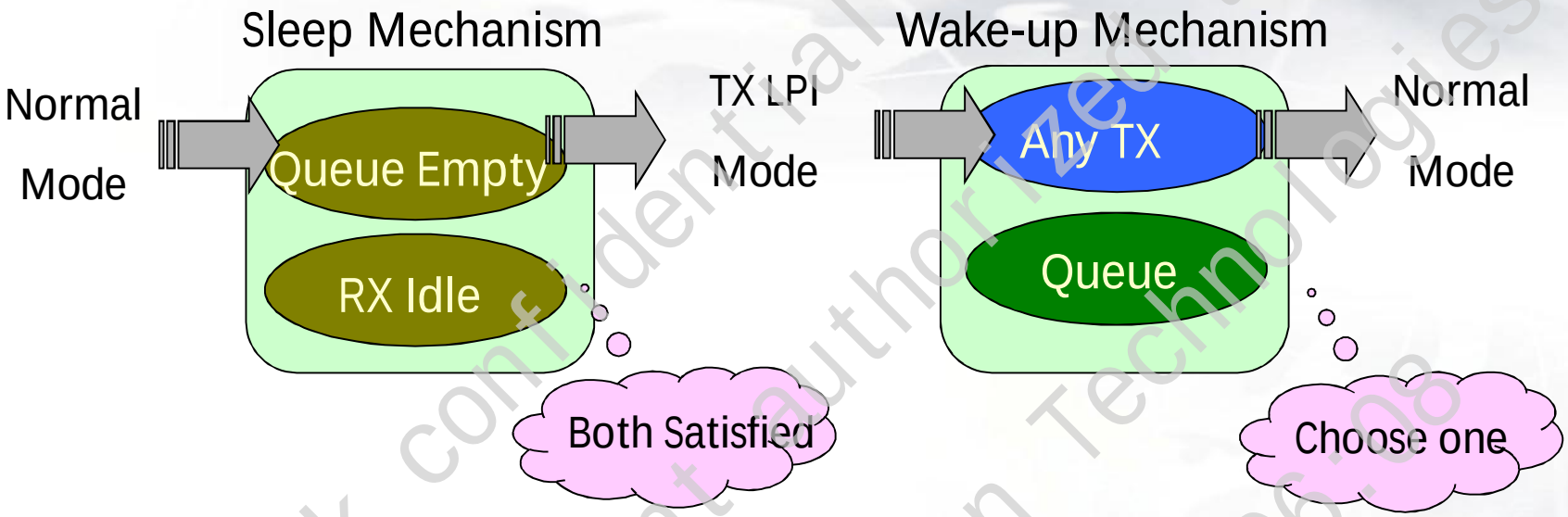
EEE - Overview

- EEE: Energy Efficient Ethernet, standardized in IEEE 802.3az
- Reduce power consumption when port is idle by shut down MAC/ PHY curcuits
  - 100BASE-Tx, 1000BASE-T, 2.5G BASE-T, 5GBASE-T, 10GBASE-T are supported
- Per port enable configurable
  - Auto-negotiate with link partner to activate EEE function if both are enabled
- Auto switch between modes
  - Normal mode
  - Low Power Idle (LPI) mode



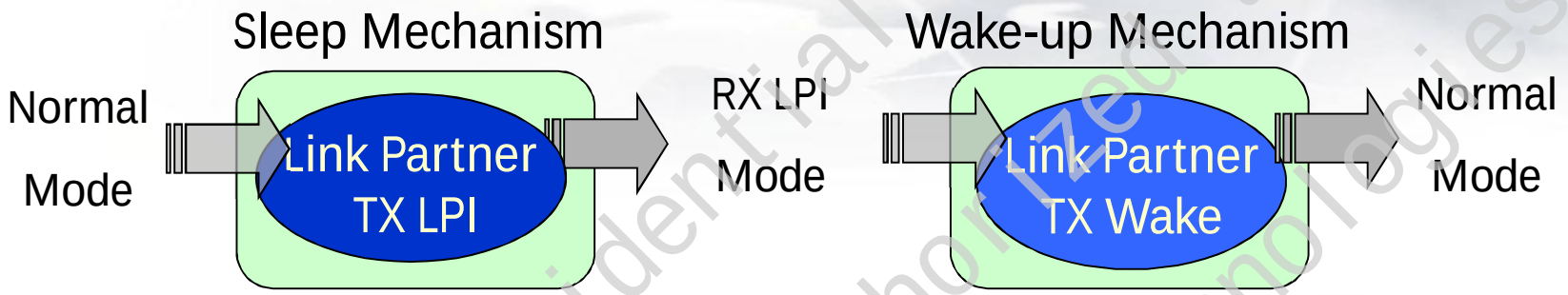


EEE – TX Sleep and Wake-up Mechanism



- Enter TX LPI mode
  - Queue Empty: TX queue is empty
  - RX Idle: RX idled exceeding a time threshold
- Exit TX LPI mode
  - Any TX: any packet needs to TX
  - Queue:
    - high queue: any packet
    - low queue: (total packet >= threshold or timer expired)

EEE – RX Sleep and Wake-up Mechanism



- Enter RX LPI mode
  - Link Partner in TX LPI mode
    - Link partner would send a signal to inform
- Exit RX LPI mode
  - Link Partner TX wake
    - Link partner would send a signal to inform

Realtek Semiconductor Corp.

LED

Overview

- 1 System LED (shared with GPIO[0])
  - Global mode: OFF/64ms/1024ms/All ON
- Maximum 60 port LED signal streams (copper + fiber)
  - Per port indicates copper and/or fiber LED signal streams (except CPU port)
  - Maximum 4 LEDs per port stream
- LED display modes
  - Serial LED mode
  - Single color scan LED mode
  - Bi-color scan LED mode
- Power-on and reset LED blinking
- Software control LED mode

Port LED configuration

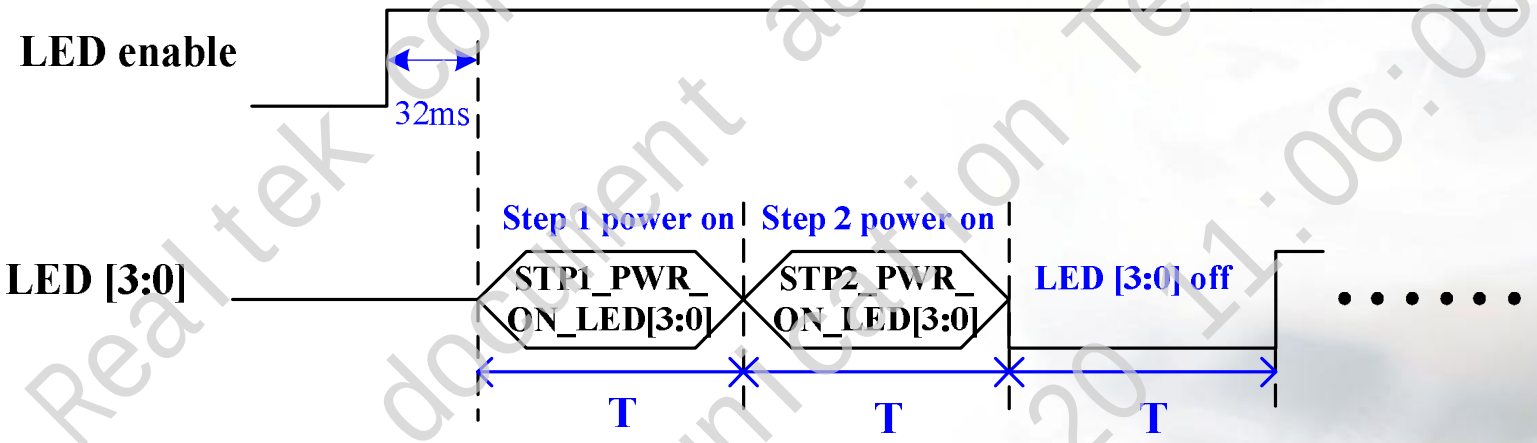
- LED polarity configuration for serial mode
  - High active / Low active
- LED streams ordering
  - Increase (Port 0 -> Port 55) / Decrease (Port 55 -> Port 0)
- Blinking time (time of a round)
  - Period: 32ms/64ms/128ms/256ms/512ms/1024ms
- Maximum 4 LEDs for each port stream
- 4 sets of LED definition profile
  - Configure the definition of LED indicator for each LED
  - Per-port, per-media (Copper/Fiber) to select the one of 4 sets
- Per-combo port configuration
  - Share the same LEDs (Using Copper LEDs)
  - Independent LEDs (Copper, Fiber use individual LEDs)

Serial mode vs Scan mode

- Serial mode
  - Working with shift registers (or RTL8231)
  - Maximum 4 LEDs for each stream
  - Better brightness
  - Simple layout
- Scan mode
  - Support by RTL8231 (LED display controller)
  - Maximum 4 LEDs for each stream in Single Color Scan mode
  - Maximum 3 LEDs for each stream in Bi-Color Scan mode
    - LED [1:0] as a Bi-Color LED
    - LED [2] as a Single Color LED
  - Reduce the BOM cost (fewer components)

Power-On Blinking

- Disabled or 400ms/800ms/1.6s (duration time T for each step)
- Three steps (3 * T):
  1. STP1_PWR_ON_LED [3:0] – to define the high/low signals for 4 LEDs on Step 1
  2. STP2_PWR_ON_LED [3:0] – to define the high/low signals for 4 LEDs on Step 2
  3. All LED off – turn off all LEDs on Step 3



Note: T is dependent on SEL_PWR_ON_BLINK[1:0]

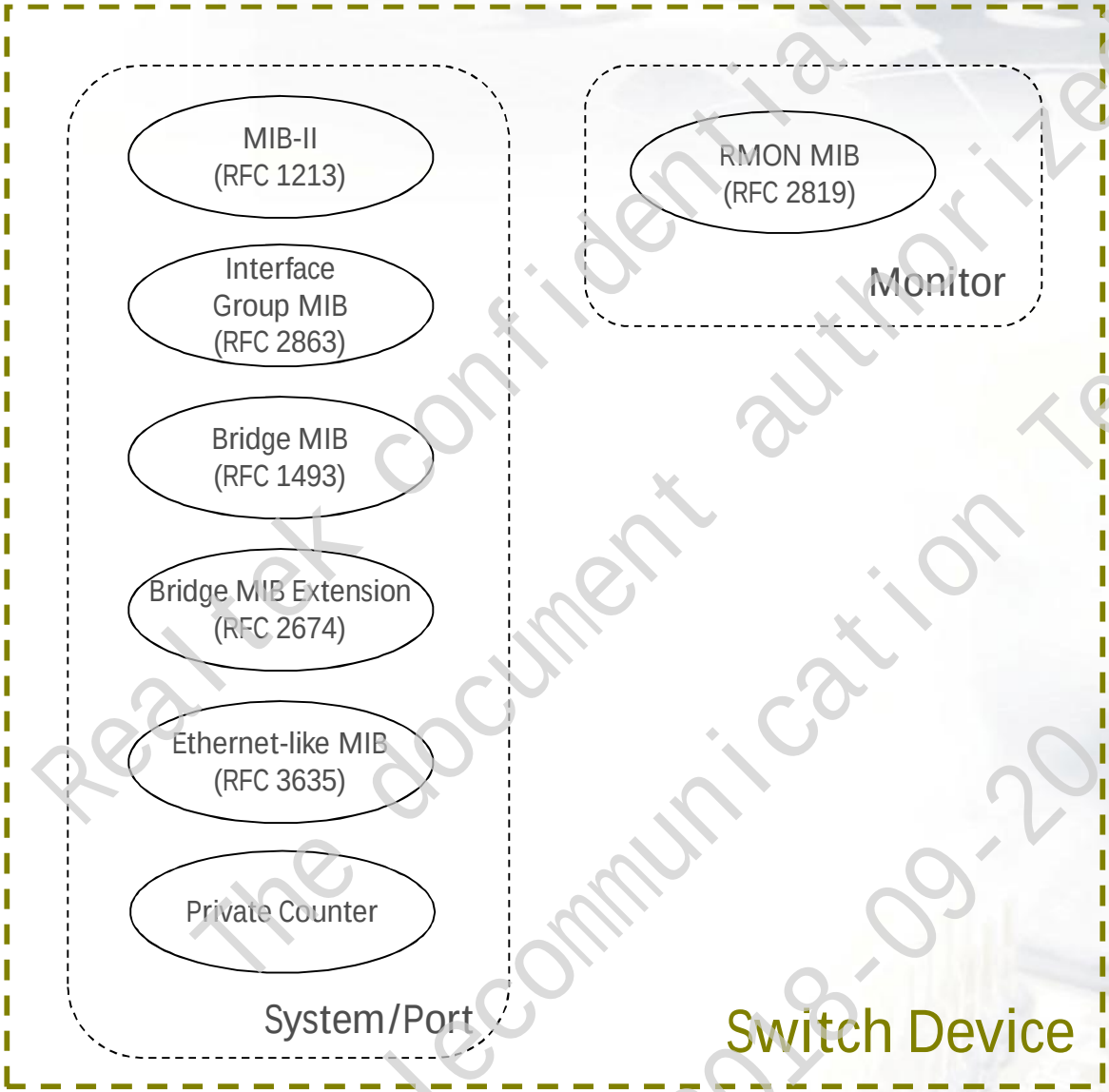
Software control LED mode

- Per-port, per-LED configuration
  - Enable/disable software control
- Per-port, per-media, per-LED configuration
  - Blinking period time
    - OFF / 32ms / 64ms / 128ms / 256ms / 512ms / 1024ms / Always ON

REALTEK
Realtek Semiconductor Corp.

MIB Statistics

Supported MIB Counter



Flexible Length Counter

- 2 sets of flexible length TX/RX good and bad packet
  - Per Set Configuration
    - Minimum packet length (1 ~ 16,383 Byte)
    - Maximum packet length (1 ~ 16,383 Byte)
  - Set 0
    - RX_etherStatsPktsFlexibleOctectSet0RT
    - RX_etherStatsPktsFlexibleOctectCRCSet0RT
    - TX_etherStatsPktsFlexibleOctectSet0RT
    - TX_etherStatsPktsFlexibleOctectCRCSet0RT
  - Set 1
    - RX_etherStatsPktsFlexibleOctectSet1RT
    - RX_etherStatsPktsFlexibleOctectCRCSet1RT
    - TX_etherStatsPktsFlexibleOctectSet1RT
    - TX_etherStatsPktsFlexibleOctectCRCSet1RT