Discussion on Maintenance Comment 201 & 202

(https://www.802-1.org/items/318 & 317)

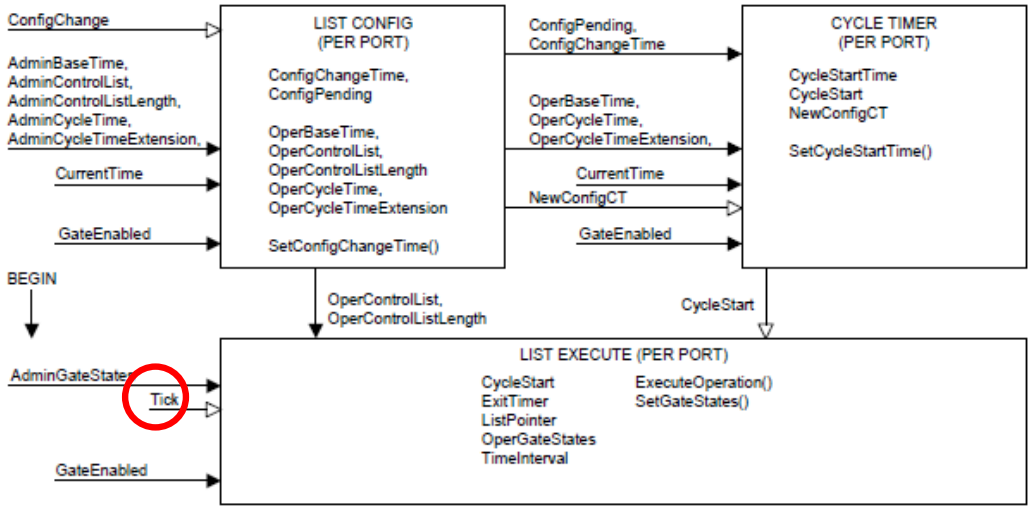
Tongtong Wang

Huawei Technologies

HUAWEI

Comment 201 : 802.1Qbv Tick granularity

- Tick input List execute FSM as time_unit signal;



- Tickgranularity is the absolute time length for one tick;

Table 12-28—The Gate Parameter Table

TickGranularity	Integer (tenths of nanoseconds)	R	BE	8.6.9.4.16

Tick granularity is used to compute timeInterval/adminCycleTime length and set TAS time gate accordingly.

Comment 201: Tick granularity

- MII Interface for 1GE/10GE/25GE MAC.

	1GE	10GE	25GE	400GE
Data path width@MII	8b	32b	64b	64b
Data path interface clocking (bit_time*bus_width)	8ns	3.2ns	2.56ns	0.16ns
MII TX_CLK	125Mhz	156.25Mhz	390.625 MHz	6250Mhz

- Interface clocking is a logic unit time, not implementation specific; (not friendly to FPGA)
- Tick granularity is the time basis of TAS scheduling port.
  - timeInterval is computed by number of ticks?! → Discuss

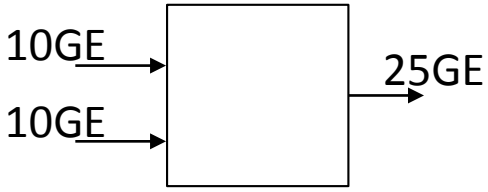
Table 8-6—Gate operations

Operation name	Parameter(s)	Action
SetGateStates	GateState, TimeInterval	The *GateState* parameter indicates a value, *open* or *closed*, for each of the Port's queues. The gates are immediately set to the states indicated in the GateState parameter. This causes gate-close events (3.1) and/or gate-open events (3.2) to occur for any queue where the new GateState represents a change of state relative to the current state of the gate. After *TimeInterval* ticks (8.6.9.4.16) has elapsed since the completion of the previous gate operation in the gate control list, control passes to the next gate operation.

- Commenter suggests to couple tick granularity with MII interface clocking
  - For better alignment between time gating with physical data clocking.

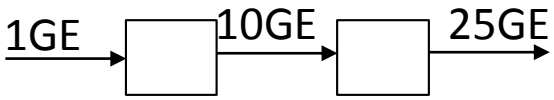
Use Case with Different Tick Granularity

- Use case 1: aggregate flows from multiple port to one



  - Tick granularity is only valid on egress port;
  - Make TAS schedule for flow 1 & 2 according to TSpec

- Use case 2: schedule one flow over multiple device w/ different port rate



  - Tick granularity are different on each device;
  - Make TAS schedule on each egress port according to TSpec

No need to care uniform tick granularity for multiple port rate.

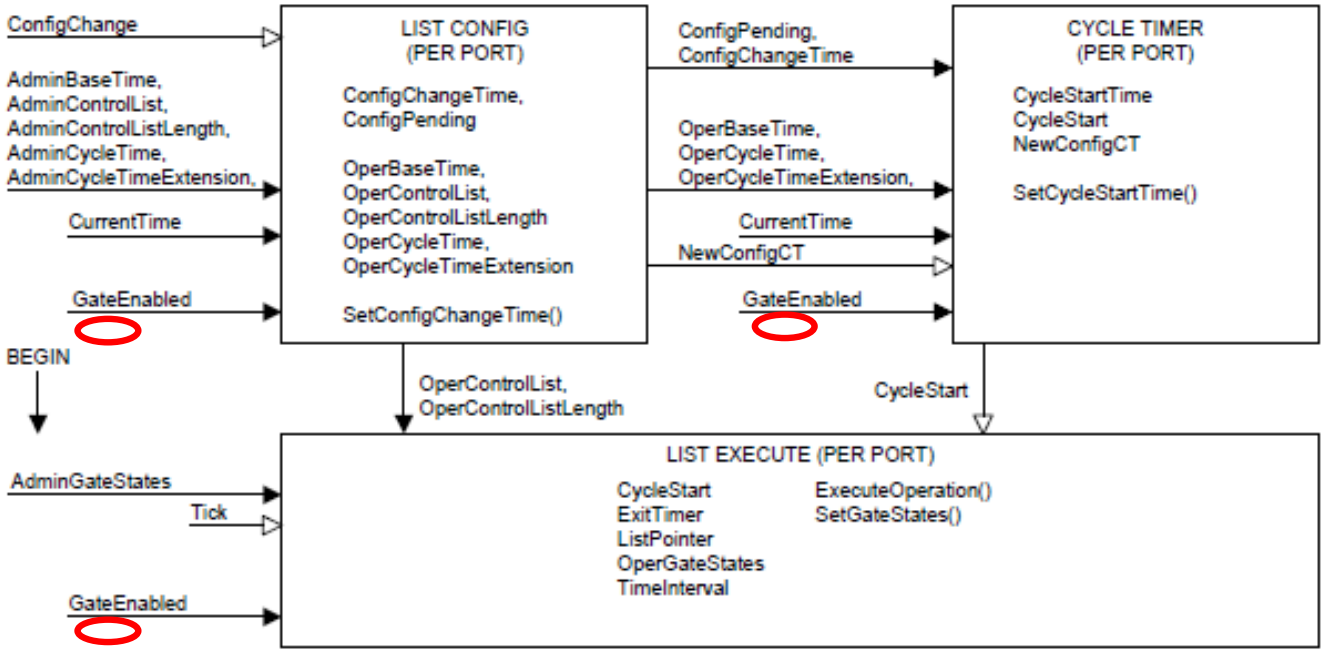
Comment 201 Discuss

- Tick granularity is the capability of physical port, declaring minimal allocable time unit to the scheduling computation function block.
- Currently minimal allocable time/bandwidth unit is expressed by 0.1ns, which varies with port rate, from 1bit(GE) to 40bit(400GE).
- Shall we change minimal allocable time window from 0.1ns to 1e-x?
  - To make it align with physical MII interface clocking

Comment 202:

Qbv ConfigChangeError counter incremented incorrectly

- GateEnable signal input to all three Qbv FSMs;



ConfigError Counter

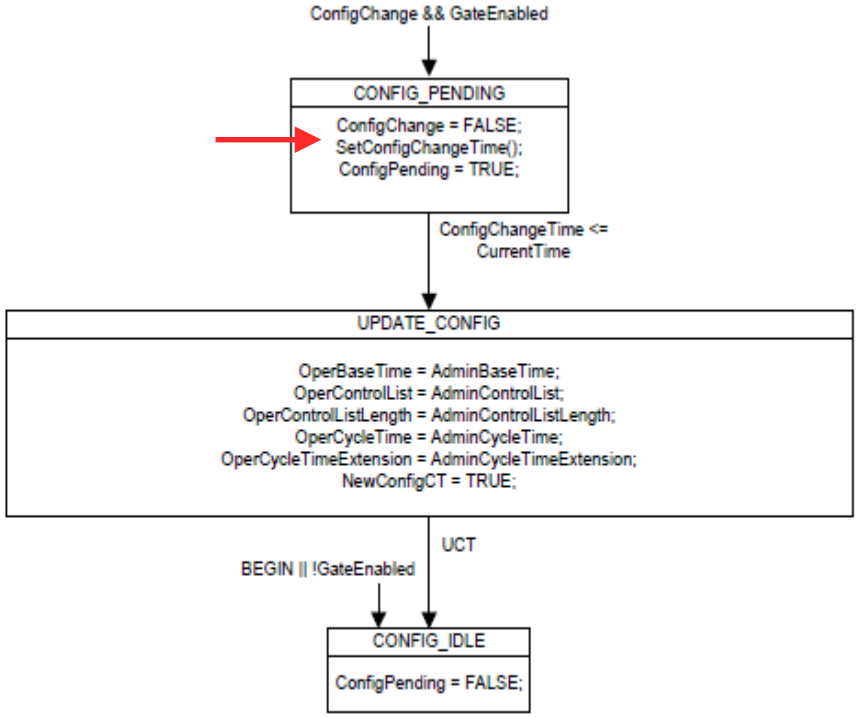


Figure 8-16—List Config state machine

a) If:
AdminBaseTime >= CurrentTime (8.6.9.4.10)
(i.e., AdminBaseTime specifies the current time or a future time)
Then:
ConfigChangeTime = AdminBaseTime

b) If:
(AdminBaseTime < CurrentTime) and (GateEnabled = FALSE)
(i.e., AdminBaseTime specifies a time in the past, and the current schedule is stopped)
Then:
ConfigChangeTime = (AdminBaseTime + N*AdminCycleTime)
where N is the smallest integer for which the relation
ConfigChangeTime >= CurrentTime
would be TRUE.

c) If:
(AdminBaseTime < CurrentTime) and (GateEnabled = TRUE)
(i.e., AdminBaseTime specifies a time in the past, and the current schedule is running)
Then:
Increment ConfigChangeError counter (12.29.1)
ConfigChangeTime = (AdminBaseTime + N*AdminCycleTime)
where N is the smallest integer for which the relation
ConfigChangeTime >= CurrentTime
would be TRUE.

- DISCUSS:
  - This erroneous scenario happens only when changing configuration during runtime and AdminBaseTime is mistakenly set.
  - Correct procedure for runtime reconfiguration is set Admin variables first and then set ConfigChange trigger.

My Views in 802.1Qbv FSM

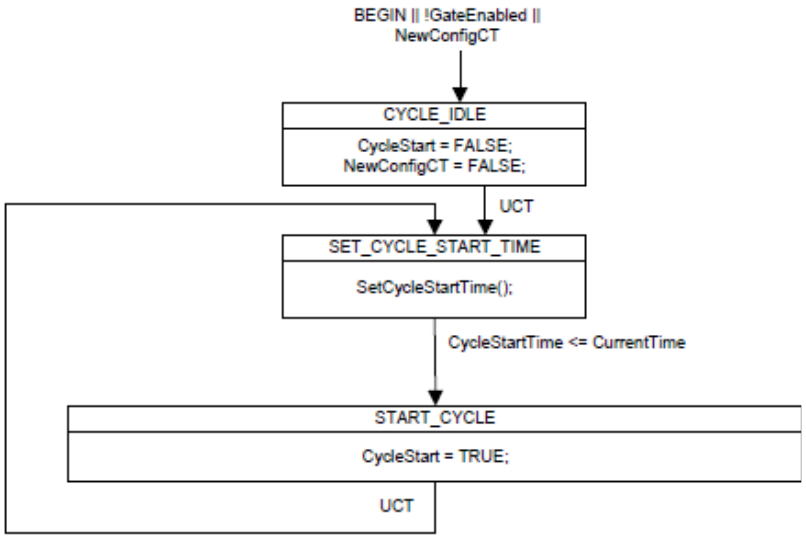


Figure 8-14—Cycle Timer state machine

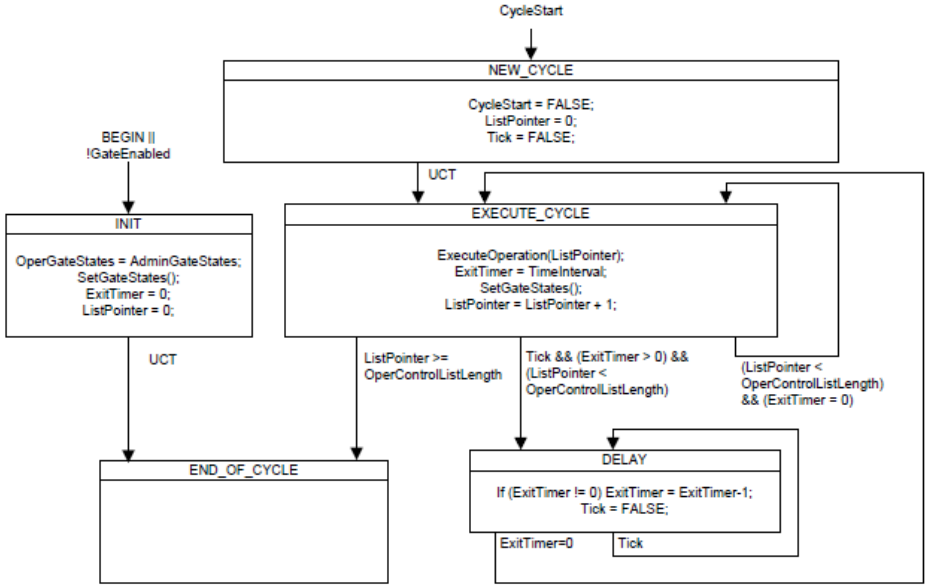
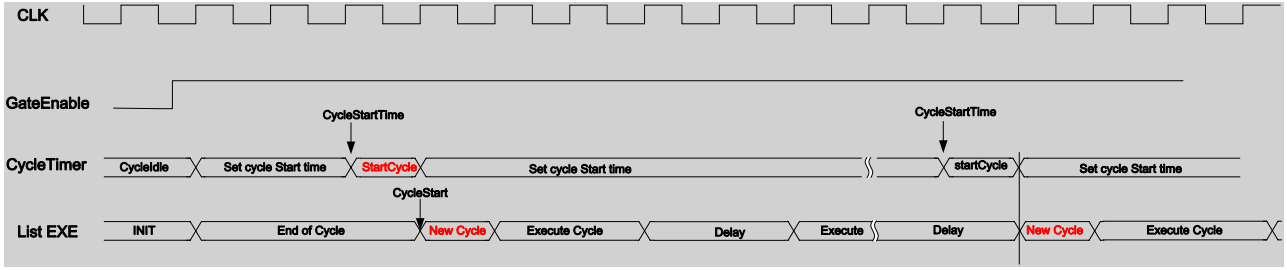


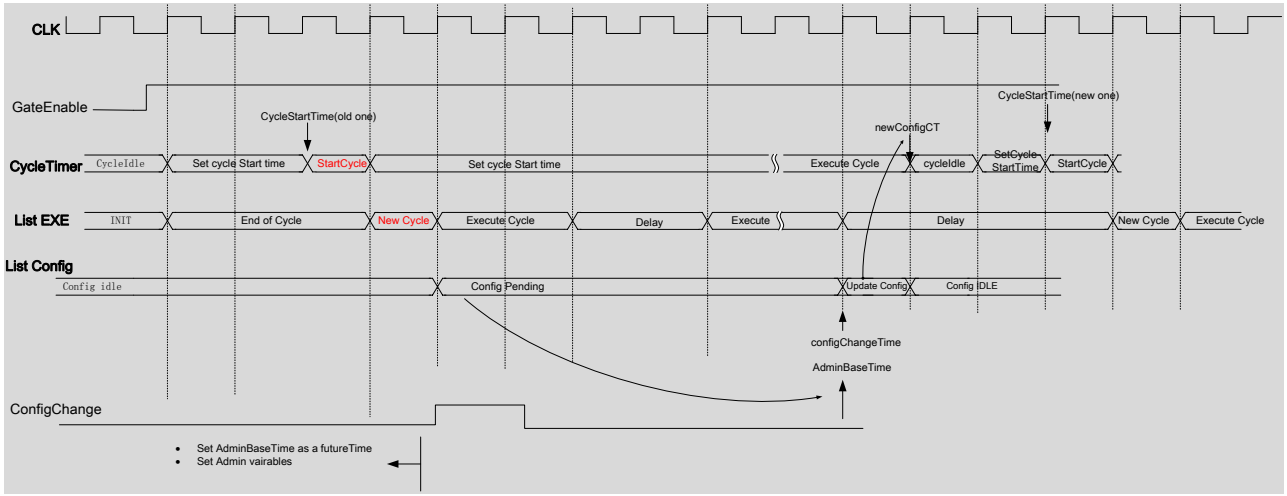
Figure 8-15—List Execute state machine



FSM in Standard is just procedural guidance, not cycle accurate.

Dynamic Reconfiguration

- The only requirement for dynamic reconfiguration is smooth changing, it is desirable to have no interference for stable flow schedule.
  - Small jitter is tolerable
  - More analysis is preferred on efficiency of dynamic reconfiguration, aiming for less wasted bandwidth and traffic bursts.



Timing diagram for dynamic reconfiguration

FSM in Standard is just procedural guidance, not cycle accurate.

Thank you

www.huawei.com

Dynamic Reconfiguration

- Dynamic reconfiguration procedure is long, it will include the following steps:
  1. Set Admin variables, and then Set ConfigChange, List config FSM enter CONFIG_PENDING
  2. Wait till ConfigChangeTime, List config FSM enter UPDATE_CONFIG, Copy Admin variables to Oper variables, and then set newConfigCT;
  3. newConfigCT will trigger cycleTimer FSM into CYCLE_IDLE state;
  4. After one UCT, cycleTimer FSM go to SET_CYCLE_START_TIME state and calculate CycleStartTime
  5. Wait till CycleStartTime, cycleTimer FSM go to START_CYCLE state and set CycleStart
  6. Finally CycleStart trigger list EXE FSM into NEW_CYCLE