Configuring Synchronous Ethernet

The Cisco ME 1200 NID support Synchronous Ethernet (SyncE), which is the PHY-layer frequency-synchronization solution for IEEE 802.3 links. It is an evolution of the conventional Ethernet and Ethernet + SDH and SONET-based synchronization. SyncE is used to synchronize and send clock information to remote sites on the network. Each network element along the synchronization path must support SyncE. SyncE provides only frequency synchronization, not related to time or space.

Prerequisites for Configuring SyncE

- NID must have an IP address.

Restrictions for Configuring SyncE

- The port number three (3) cannot be nominated to source number one (1).

Information About Synchronous Ethernet

This chapter describes the Synchronous Ethernet features, standards, and limitations in the Cisco ME 1200 Series Carrier Ethernet Access Device. This chapter also describes procedures to configure Synchronous Ethernet.

Synchronous Ethernet Overview

A separate external time-division multiplexing (TDM) circuit is required to provide synchronized timing to multiple remote network elements (NEs) for packet transport networks like Cisco Carrier Packet Transport system. The Synchronous Ethernet (SycnE) feature addresses this requirement by providing effective timing to the remote NEs through a packet network without using an external circuit for timing.

With Ethernet equipment gradually replacing existing Synchronous Optical Networking (SONET) and Synchronous Digital Hierarchy (SDH) equipment in service-provider networks, frequency synchronization is required to provide high-quality clock synchronization over Ethernet ports. The SyncE feature provides the required synchronization at the physical level. Operation messages maintain SyncE links and ensure that a node always derives timing from the most reliable source. SyncE uses the Ethernet Synchronization Message Channel (ESMC) to enable traceability of the best clock source to correctly define the timing source and prevent a timing loop.

The Cisco ME 1200 Series Carrier Ethernet Access Device supports Synchronous Ethernet (SyncE), which is the physical layer frequency-synchronization solution for IEEE 802.3 links. SyncE is defined by the ITU-T standards such as G.8261, G.8262, G.8264, and G.781. It is an evolution of the conventional Ethernet and Ethernet + SDH and SONET-based synchronization. SyncE is used to synchronize and send clock information to remote sites on the network. For SyncE to work, each network element along the synchronization path must support SyncE. SyncE provides only frequency synchronization, not related to time or space.

Understanding SyncE

SyncE provides the Ethernet physical layer network (PHY) level frequency distribution of known common precision frequency references. Clocks for use in SyncE are compatible with the clocks used in the SONET/SDH synchronization network. To achieve network synchronization, synchronization information is transmitted through the network via synchronous network connections with performance of egress clock. In SONET/SDH the communication channel for conveying clock information is SSM, and in SyncE it is the ESMC.

SyncE is a standard for distribution of frequency over Ethernet links. Other standards (IEEE Std. 1588 Precision Time Protocol [PTP], IETF Network Time Protocol [NTP], and so on) have been and are being developed or enhanced for high-quality time distribution and Adaptive Clock Recovery (ACR) requirements.

To maintain the timing chain in SONET/SDH, operators often use SSM. Information provided by SSM Quality Levels (SSM-QL) helps a node derive timing from the most reliable source and prevent timing loops. The SONET/SDH header has a QL information present in the S1 bytes of its header. Hence, the SONET/SDH does not require any specific channel for QL information exchange. As the Ethernet does not have the QL information in its header, it requires ESMC for QL information. Because Ethernet networks are not required to be synchronous on all links or in all locations, a specific channel, the ESMC channel defined in G.8264, provides this service. ESMC is composed of the standard Ethernet header for an organization-specific slow protocol, the ITU-T OUI; a specific ITU-T subtype; an ESMC-specific header; a flag field; and a type, length, value (TLV) structure: the use of flags and TLVs aimed at improving the management of Synchronous Ethernet links and the associated timing change.

For more information, see Configuring Synchronous Ethernet.

SyncE Standards

- ITU-T G.8261: Timing and synchronization aspects in packet network
- ITU-T G.8262: Timing characteristics of Synchronous Ethernet equipment slave clock
- ITU-T G.8264: Distribution of timing through packet networks
- ITU-T G.781: Synchronization layer functions

Understanding SyncE Protocols

Network clocking uses the Synchronization Status Messages (SSM) mechanism to exchange the Quality Level (QL) of the clock between the network elements. In Ethernet, Ethernet Synchronization Message Channel (ESMC) is used for SSM exchange.

The two important protocols used for SyncE are:

- Synchronization Status Messages (SSM)
- Ethernet Synchronization Messaging Channel (ESMC)

Synchronization Status Messages (SSM)

Network elements use Synchronization Status Messages (SSM) to inform the neighboring elements about the Quality Level (QL) of the clock. The non-ethernet interfaces such as optical interfaces and SONET/T1/E1 SPA framers uses SSM. The key benefits of the SSM functionality:

- Prevents timing loops.
- Provides fast recovery when a part of the network fails.
- Ensures that a node derives timing from the most reliable clock source.

Ethernet Synchronization Messaging Channel (ESMC)

To maintain a logical communication channel in synchronous network connections, ethernet relies on a channel called Ethernet synchronization Messaging Channel (ESMC). This is based on IEEE 802.3 Organization Specific Slow Protocol standards. ESMC relays the SSM code that represents the Quality Level (QL) of the Ethernet Equipment Clock (EEC) in a physical layer.

The ESMC packets are received only for those ports configured as clock sources and transmitted on all the SyncE interfaces in the system. These packets are then processed by the Clock selection algorithm and are used to select the best clock. The Tx frame is generated based on the QL value of the selected clock source and sent to all the enabled SyncE ports.

Understanding SyncE Clocks

Clock Selection Algorithm

The clock selection algorithm selects the best available synchronization source from the nominated sources. This algorithm exhibits nonrevertive behavior among the clock sources with the same QL value, and always selects the signal with the best QL value. For clock option SDH, the default is revertive, and for clock option SONET, the default is nonrevertive.

The following parameters contribute to the selection process:

- Quality level (QL)
- Signal fail through QL-FAILED
- Priority
- External commands (Manual, Auto-revertive and so on)

Clock Selection Modes

A clock selection is said to be the best, when the clock source is configured with the highest QL and with the highest priority (for the ones with equal QL).

The following are different clock selection modes:

- **Manual—**the clock selector is manually set to the chosen clock source. If the manually selected clock source fails, then, the clock selector goes to the holdover state.
- **Selected—**the clock selector selects the clock manually, however, the highest priority selected clock source becomes the Source.
- **NonRevertive—**the clock selector selects the best clock source only done when the selected clock fails.
- **Revertive—**the selection of the best clock source is constantly searched for.
- **Holdover—**the clock selector is forced to the holdover state.
- **Freerun—**the clock selector is forced to the free run state.

Manual mode is used to force selection of a specific source. It is also used to switch back to the primary source if auto-nonrevertive mode is selected and the failure is cleared. Selected mode is used to freeze the current clock source, in case of a failure on switchover.

How to Configure SyncE

Configuring SyncE Global Defaults

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**setSyncEglobalDefaultConfig set-global-default-config** **Example:** `Switch(SyncE)# setSyncEglobalDefaultConfig set-global-default-config`	Sets the global configuration to defaults. This means that the SyncE feature is not configured on the device.
Step 3	**setSyncEglobalDefaultConfig review** **Example:** `Switch(SyncE)# setSyncEglobalDefaultConfig review`	Displays the configuration.
Step 4	**setSyncEglobalDefaultConfig commit** **Example:** `Switch(SyncE)# setSyncEglobalDefaultConfig commit`	Sends the configuration to the NID.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# setSyncEglobalDefaultConfig set-global-default-config
Switch(SyncE)# setSyncEglobalDefaultConfig review

Commands in queue:
    setSyncEglobalDefaultConfig set-global-default-config

Switch(SyncE)# setSyncEglobalDefaultConfig commit

    SetSyncEglobalDefaultConfig Commit Success!!!

Switch(SyncE)# exit
```

Viewing SyncE Global Defaults

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**getSyncEglobalconfig get-global-config** **Example:** `Switch(SyncE)# getSyncEglobalconfig get-global-config`	Displays the SyncE global configuration details.
Step 3	**getSyncEglobalconfig review** **Example:** `Switch(SyncE)# getSyncEglobalconfig review`	Displays the configuration that are in queue.
Step 4	**getSyncEglobalconfig commit** **Example:** `Switch(SyncE)# getSyncEglobalconfig commit`	Sends the configuration to the NID.
Step 5	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# getSyncEglobalconfig get-global-config
Switch(SyncE)# getSyncEglobalconfig review

Commands in queue:
    getSyncEglobalConfig get-global-config

Switch(SyncE)# getSyncEglobalconfig commit

   GetSyncEglobalConfig_Output.synce_global_conf.clock_select_config.t = 5
   GetSyncEglobalConfig_Output.synce_global_conf.clock_select_config.u.revertive = ''
   GetSyncEglobalConfig_Output.synce_global_conf.wait_to_restore = 5
   GetSyncEglobalConfig_Output.synce_global_conf.SSM_QL_for_holdover.t = 1
   GetSyncEglobalConfig_Output.synce_global_conf.SSM_QL_for_holdover.u.QL_NONE = ''
   GetSyncEglobalConfig_Output.synce_global_conf.SSM_QL_for_freerun.t = 1
   GetSyncEglobalConfig_Output.synce_global_conf.SSM_QL_for_freerun.u.QL_NONE = ''
   GetSyncEglobalConfig_Output.synce_global_conf.EEC_Option.t = 1
   GetSyncEglobalConfig_Output.synce_global_conf.EEC_Option.u.EEC1 = ''

   GetSyncEglobalConfig Commit Success!!!

Switch(SyncE)# exit
```

Configuring SyncE Clock Defaults

This task configures the SyncE configurations to defaults.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**setSyncEclockDefaultConfig set-synce-clock-config-defaults-req** **Example:** `Switch(SyncE)# setSyncEclockDefaultConfig set-synce-clock-config-defaults-req`	Set SyncE default Clock configurations.
Step 3	**setSyncEclockDefaultConfig review** **Example:** `Switch(SyncE)# setSyncEclockDefaultConfig review`	Displays the configuration.
Step 4	**setSyncEclockDefaultConfig commit** **Example:** `Switch(SyncE)# setSyncEclockDefaultConfig commit`	Sends the configuration to the NID.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# setSyncEclockDefaultConfig set-synce-clock-config-defaults-req
Switch(SyncE)# setSyncEclockDefaultConfig review

Commands in queue:
    setSyncEclockDefaultConfig set-synce-clock-config-defaults-req

Switch(SyncE)# setSyncEclockDefaultConfig commit

    SetSyncEclockDefaultConfig Commit Success!!!

Switch(SyncE)# exit
```

Viewing SyncE Clock Defaults

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**getSyncEclockdefaultConfig get-synce-clock-config-defaults-req** **Example:** `Switch(SyncE)# getSyncEclockdefaultConfig get-synce-clock-config-defaults-req`	Displays the SyncE default Clock configurations.
Step 3	**getSyncEclockdefaultConfig review** **Example:** `Switch(SyncE)# getSyncEclockdefaultConfig review`	Displays the configuration.
Step 4	**getSyncEclockdefaultConfig commit** **Example:** `Switch(SyncE)# getSyncEclockdefaultConfig commit`	Sends the configuration to the NID.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# getSyncEclockdefaultConfig get-synce-clock-config-defaults-req
Switch(SyncE)# getSyncEclockdefaultConfig review

Commands in queue:
getSyncEclockDefaultConfig get-synce-clock-config-defaults-req

Switch(SyncE)# getSyncEclockdefaultConfig commit

  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].state = false
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].port = 1
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].priority = 0
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].SSM_overwrite.t = 1
 GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].SSM_overwrite.u.QL_NONE
 = ''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].hold_off.t = 1
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].hold_off.u.disabled
 = ''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].aneg_mode.t = 1
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[0].aneg_mode.u.none =
''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].state = false
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].port = 2
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].priority = 0
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].SSM_overwrite.t = 1
 GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].SSM_overwrite.u.QL_NONE
 = ''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].hold_off.t = 1
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].hold_off.u.disabled
 = ''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].aneg_mode.t = 1
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[1].aneg_mode.u.none =
''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].state = false
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].port = 3
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].priority = 0
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].SSM_overwrite.t = 1
 GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].SSM_overwrite.u.QL_NONE
 = ''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].hold_off.t = 1
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].hold_off.u.disabled
 = ''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].aneg_mode.t = 1
  GetSyncEclockDefaultConfig_Output.clock_sel_config.source_configs[2].aneg_mode.u.none =
''
  GetSyncEclockDefaultConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_1_UNI
 = false
  GetSyncEclockDefaultConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_2_UNI
 = false
  GetSyncEclockDefaultConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_3_UNI
 = false
  GetSyncEclockDefaultConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_4_UNI
 = false
  GetSyncEclockDefaultConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_5_UNI
 = false
  GetSyncEclockDefaultConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_6_UNI
```

```
 = false

  GetSyncEclockDefaultConfig Commit Success!!!

Switch(SyncE)# exit
```

Configuring Clock Source

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **port** *Physical port* } **Example:** `Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1 setSyncEclockConfig clock-sel-config occur source-configs port 2`	Configures the clock source on the port. • **source-configs—**Specifies the source configurations. ◦ *Physical port***—**Physical port. The range is from 1 to 6. • **port—**Specifies the physical port. ◦ *source_configs***—**nominate a port number to be the clock source. The range is from 1 to 2.
Step 3	**setSyncEclockConfig review** **Example:** `Switch(SyncE)# setSyncEclockConfig review`	Displays the configuration.
Step 4	**setSyncEclockConfig commit** **Example:** `Switch(SyncE)# setSyncEclockConfig commit`	Sends the configuration to the NID.
Step 5	**exit** **Example:** `Switch(syncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2
Switch(SyncE)# setSyncEclockConfig review
```

```
Commands in queue:
  setSyncEclockConfig clock-sel-config source-configs 1
clock-sel-config occur source-configs port 2

Switch(SyncE)# setSyncEclockConfig commit

  SetSyncEclockConfig Commit Success!!!

Switch(SyncE)# exit
```

Viewing Clock Configurations

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**getSyncEclockConfig get-clock-config** **Example:** `Switch(SyncE)# getSyncEclockConfig get-clock-config`	Displays clock configuration.
Step 3	**setSyncEclockConfig review** **Example:** `Switch(SyncE)# setSyncEclockConfig review`	Displays the configuration.
Step 4	**getSyncEclockConfig commit** **Example:** `Switch(SyncE)# getSyncEclockConfig commit`	Sends the configuration to the NID.
Step 5	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# getSyncEclockConfig get-clock-config
Switch(SyncE)# setSyncEclockConfig review

Commands in queue:
 getSyncEclockConfig get-clock-config

Switch(SyncE)# getSyncEclockConfig commit

   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].state = true
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].port = 4
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].priority = 1
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].SSM_overwrite.t = 2
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].SSM_overwrite.u.QL_PRC =
```

```
'0'
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].hold_off.t = 2
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].hold_off.u.value = 800
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].aneg_mode.t = 1
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[0].aneg_mode.u.none = ''
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].state = true
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].port = 3
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].priority = 0
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].SSM_overwrite.t = 2
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].SSM_overwrite.u.QL_PRC =
'0'
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].hold_off.t = 2
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].hold_off.u.value = 1000
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].aneg_mode.t = 1
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[1].aneg_mode.u.none = ''
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].state = false
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].port = 3
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].priority = 0
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].SSM_overwrite.t = 1
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].SSM_overwrite.u.QL_NONE =
 ''
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].hold_off.t = 1
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].hold_off.u.disabled = ''
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].aneg_mode.t = 1
   GetSyncEClockConfig_Output.clock_sel_config.source_configs[2].aneg_mode.u.none = ''
   GetSyncEClockConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_1_UNI =
false
   GetSyncEClockConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_2_UNI =
false
   GetSyncEClockConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_3_UNI =
true
   GetSyncEClockConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_4_UNI =
true
   GetSyncEClockConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_5_UNI =
false
   GetSyncEClockConfig_Output.clock_sel_config.ssm_enable_ports.GigabitEthernet_6_UNI =
false

   GetSyncEClockConfig Commit Success!!!

Switch(SyncE)# exit
```

Overwriting the Quality Level (QL)

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**setSyncEclockConfig clock-sel-config** {**source-configs** | **ssm-enable-ports** {**GigabitEthernet-1-UNI** | **GigabitEthernet-2-UNI** | **GigabitEthernet-3-UNI** | **GigabitEthernet-4-UNI** | **GigabitEthernet-5-UNI** | **GigabitEthernet-6-UNI**} {**disable** | **enable**}}} **Example:** `Switch(SyncE)# setSyncEclockConfig clock-sel-config ssm-enable-ports GigabitEthernet-4-UNI enable`	Enters the SyncE clock configuration to select the UNI ports. Here selecting the physical port 4: • **GigabitEthernet-1-UNI—**Physical port 1. • **GigabitEthernet-2-UNI—**Physical port 2. • **GigabitEthernet-3-UNI—**Physical port 3. • **GigabitEthernet-4-UNI—**Physical port 4.

	Command or Action	Purpose
		• **GigabitEthernet-5-UNI—**Physical port 5. • **GigabitEthernet-6-UNI—**Physical port 6. • **disable—**Disables the SSM on the configured port. • **enable—**Enables the SSM on the configured port.
Step 3	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **port** *Physical port* } **Example:** `Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1 port 4`	Configures the clock source on the port: • *source-configs***—**Nominates a clock source, either 1 or 2. • *Physical port***—**Physical port. The range is from 1 to 6.
Step 4	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **SSM-overwrite** {**QL-DNU** | **QL-EEC1** | **QL-EEC2** | **QL-INV** | **QL-NONE** | **QL-PRC** | **QL-SSUA QL-SSUB**}} **Example:** `Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1 SSM-overwrite QL-PRC`	Selects QL value to overwrite any received QL in an SSM message • *source-configs***—**Nominate a port number to be the clock source. The range is from 1 to 2.
Step 5	**setSyncEclockConfig review** **Example:** `Switch(SyncE)# setSyncEclockConfig review`	Displays the configuration.
Step 6	**setSyncEclockConfig commit** **Example:** `Switch(SyncE)# setSyncEclockConfig commit`	Sends the configuration to the NID.
Step 7	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# setSyncEclockConfig clock-sel-config ssm-enable_ports GigabitEthernet-4-UNI
 enable
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1 port 4
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1 SSM-overwrite QL-PRC
Switch(SyncE)# setSyncEclockConfig review

Commands in queue:
   setSyncEclockConfig clock-sel-config ssm-enable_ports GigabitEthernet-4-UNI enable
   setSyncEclockConfig clock-sel-config source-configs 1 port 4
   setSyncEclockConfig clock-sel-config source-configs 0 SSM-overwrite QL-PRC
```

```
Switch(SyncE)# setSyncEclockConfig commit

   SetSyncEclockConfig Commit Success!!!

Switch(SyncE)# exit
```

Understanding Clock Redundancy

On the Cisco ME 1200 NID, it is possible to configure up to two clock sources. Any Ethernet port can act as a clock source. For the Cisco ME 1200 NID, external clock input does not exit. Based on the priority and Quality level (QL) of the clock sources, the best source is selected.

To select the best source, nominate the clock sources, and then set priorities for each of them. Enable SSM on ports used for synchronization. Note that QL overwrites the priority. That means, if port 2 receives QL-PRC and port 1 receives only QL-EEC1, and even though port 1 has higher priority than port 2, the port 2 is selected as QL overwrites.

Configuring Clock Redundancy

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**setSyncEclockConfig clock-sel-config** {**source-configs** | **ssm-enable-ports** {**GigabitEthernet-1-UNI** | **GigabitEthernet-2-UNI** | **GigabitEthernet-3-UNI** | **GigabitEthernet-4-UNI** | **GigabitEthernet-5-UNI** | **GigabitEthernet-6-UNI**} {**enable** | **disable**}}} **Example:** `Switch(SyncE)# setSyncEclockConfig clock-sel-config ssm-enable-ports GigabitEthernet-5-UNI enable`	Enters the SyncE clock configuration to select the UNI ports. Here selecting the physical port 5. • **GigabitEthernet-1-UNI—**Physical port 1. • **GigabitEthernet-2-UNI—**Physical port 2. • **GigabitEthernet-3-UNI—**Physical port 3. • **GigabitEthernet-4-UNI—**Physical port 4. • **GigabitEthernet-5-UNI—**Physical port 5. • **GigabitEthernet-6-UNI—**Physical port 6. • **disable—**Disables the SSM on the configured port. • **enable—**Enables the SSM on the configured port.
Step 3	**setSyncEclockConfig clock-sel-config** {**source-configs** | **ssm-enable-ports** {**GigabitEthernet-1-UNI** | **GigabitEthernet-2-UNI** | **GigabitEthernet-3-UNI** | **GigabitEthernet-4-UNI** | **GigabitEthernet-5-UNI** | **GigabitEtherne-6-UNI**} {**enable** | **disable**}}}	Enters the SyncE clock configuration to select the UNI ports. Here selecting the physical port 6. • **GigabitEthernet-1-UNI—**Physical port 1. • **GigabitEthernet-2-UNI—**Physical port 2.

	Command or Action	Purpose
	Example: `Switch(SyncE)# setSyncEclockConfig clock-sel-config ssm-enable-ports GigabitEthernet-6-UNI enable`	• **GigabitEthernet-3-UNI—**Physical port 3. • **GigabitEthernet-4-UNI—**Physical port 4. • **GigabitEthernet-5-UNI—**Physical port 5. • **GigabitEthernet-6-UNI—**Physical port 6. • **disable—**Disables the SSM on the configured port. • **enable—**Enables the SSM on the configured port.
Step 4	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **port** *Physical port*} **Example:** `Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1 port 5`	Configures the clock source on the port. Here the configuration is done on port 5, and the *source-config* is set to 1. • *source-configs***—**Nominate a port number to be the clock source. The range is from 1 to 2. • *Physical port***—**Physical port. The range is from 1 to 6.

Step 5	**setSyncEclockConfig clock-sel-config** {**source-configs** {**priority** *priority*}}
	Example:
	`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1`
	`setSyncEclockConfig clock-sel-config occur source-configs port 2 priority 0`
		Sets the clock priority. Here the clock priority is set to 0.
		• *priority***—**Clock priority value. Either 0 or 1.

Step 6	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **state** {**enable**	**disable**}}
		Example:
		`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1`
		`setSyncEclockConfig clock-sel-config occur source-configs port 2 state enable`
			Enables or Disables the clock source.
			• *source-configs***—**nominate a port number to be the clock source. The range is from 1 to 2.

Step 7	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **hold-off** {**disabled**		**value** {**300 msec to 1800 msec**}}}
			Example:
			`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1`
			`setSyncEclockConfig clock-sel-config occur source-configs port 2  hold-off value 1000`
				Sets the Hold-off timer value. Active loss of clock source is delayed by the selected amount of time. The clock selector changes the clock source if the loss of clock condition is cleared within this time.
				• *source-configs***—**nominate a port number to be the clock source. The range is from 1 to 2.

	Command or Action	Purpose

Step 8	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **port** *Physical port*}
	
	Example:
	`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1`
	`setSyncEclockConfig clock-sel-config occur source-configs port 2`
		Configures the clock source on the port. Here the configuration is done on port 6, and the source_config is set to 0.
		• *source-configs*—Nominate a port number to be the clock source. The range is from 1 to 2.
		• *Physical port*—Physical port. The range is from 1 to 6.

Step 9	**setSyncEclockConfig clock-sel-config** {**source-configs** {**priority** *priority* }}
	
	Example:
	`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1`
	`setSyncEclockConfig clock-sel-config occur source-configs port 2 priority 1`
		Sets the clock priority. Here the clock priority is set to 1.
		• *priority*—Clock priority value. Either 0 or 1.

Step 10	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **state** {**enable** | **disable**}}
	
	Example:
	`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1`
	`setSyncEclockConfig clock-sel-config occur source-configs port 2 state enable`
		Enables or Disables the clock source.
		• *source-configs*—Nominate a port number to be the clock source. The range is from 1 to 2.

Step 11	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs***hold-off** {**disabled** | **value** {**300 msec to 1800 msec**}}}
	
	Example:
	`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1`
	`setSyncEclockConfig clock-sel-config occur source-configs port 2 hold-off value 800`
		Sets the Hold-off timer value. Active loss of clock source is delayed by the selected amount of time. The clock selector changes the clock source if the loss of clock condition is cleared within this time.
		• *source-configs*—Nominate a port number to be the clock source. The range is from 1 to 2.

Step 12	**setSyncEclockConfig review** **Example:** `Switch(SyncE)# setSyncEclockConfig review`	Displays the configuration.
Step 13	**setSyncEclockConfig commit** **Example:** `Switch(SyncE)# setSyncEclockConfig commit`	Sends the configuration to the NID.
Step 14	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# setSyncEclockConfig clock-sel-config ssm-enable-ports GigabitEthernet-5-UNI
 enable
Switch(SyncE)# setSyncEclockConfig clock-sel-config ssm-enable-ports GigabitEthernet-6-UNI
 enable
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2 priority 0
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2 state enable
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2 hold-off value 1000
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2 priority 1
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2 state enable
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
setSyncEclockConfig clock-sel-config occur source-configs port 2 hold-off value 800

Switch(SyncE)# setSyncEclockConfig review

Commands in queue:
 setSyncEclockConfig clock-sel-config ssm-enable-ports GigabitEthernet-5-UNI enable
 setSyncEclockConfig clock-sel-config ssm-enable-ports GigabitEthernet-6-UNI enable
 setSyncEclockConfig clock-sel-config source-configs 1 port 2
 setSyncEclockConfig clock-sel-config source-configs 1 priority 0
 setSyncEclockConfig clock-sel-config source-configs 1 state enable
 setSyncEclockConfig clock-sel-config source-configs 1 hold-off value 1000
 setSyncEclockConfig clock-sel-config source-configs 1 port 2
 setSyncEclockConfig clock-sel-config source-configs 1 priority 1
 setSyncEclockConfig clock-sel-config source-configs 1 state enable
 setSyncEclockConfig clock-sel-config source-configs 1 hold_off value 800

Switch(SyncE)# setSyncEclockConfig commit

 SetSyncEclockConfig Commit Success!!!

Switch(SyncE)# exit
```

Understanding SyncE Timers

You can manage syncE timers by changing the priority of the clock sources. You can also influence selection by modifying the following timers:

- WTR (Wait to restore) Timer
- Hold-off Timer

WTR Timer

The WTR time is activated on the falling edge of a clock source failure (in Revertive mode). This means that the clock source is first available for clock selection after WTR Time (can be cleared).

Hold-off Timer

In the Hold-off timer, the active loss of clock source is delayed by the selected amount of time. The clock selector does not change the clock source if the loss of clock condition is cleared within this time.

Configuring SyncE Timers

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**setSyncEglobalConfig synce-global-conf** {**EEC-Option** | **SSM-QL-for-freerun** | **SSM-QL-for-holdover** | **clock-select-config** | **wait-to-restore**}	Enters the SyncE global configuration. • **EEC-Option—**Selects PLL EEC option. • **SSM-QL-for-freerun—**Transmits SSM QL value when clock selector is in Free Run Mode. • **SSM-QL-for-holdover—**Transmits SSM QL value when clock selector is in Hold Over State. • **clock-select-config—**Selection mode of nominated clock sources. • **wait-to-restore—**Select the wait to restore time.
Step 3	**setSyncEglobalConfig synce-global-conf wait-to-restore** *wait to restore time* **Example:** `Switch(SyncE)# setSyncEglobalConfig synce-global-conf wait-to-restore 1`	Enters the wait to restore time. • *wait to restore time*—Restore time. The range is from 0 to 12 minutes; enter the value zero to disable.
Step 4	**setSyncEglobalConfig synce-global-conf clock-select-config** {**freerun** | **holdover** | **manual** *manually set* | **nonrevertive** | **revertive** | **selected**} **Example:** `Switch(SyncE)# setSyncEglobalConfig synce-global-conf clock-select-config revertive`	Enters the selection mode of nominated clock sources. • **freerun—**Selector is forced in free run. • **holdover—**Selector is forced in holdover. • **manual—**Selector is manually set to chosen clock source. ◦ *manually set*—Clock source.The range is from 1 to 2. • **nonrevertive—**Automatic clock selection, selecting best clock source nonrevertively. • **revertive—**Automatic clock selection, selecting best clock source revertively. • **selected—**Manual clock selection, selecting pt selected clock source.
Step 5	**setSyncEglobalConfig synce-global-conf SSM-QL-for_holdover** {**QL-DNU** | **QL-EEC1** |	Transmits SSM QL value when clock selector is in Hold Over State.

	Command or Action	Purpose
	QL-EEC2 | **QL-INV** | **QL-NONE** | **QL-PRC** | **QL-SSUA QL-SSUB**} **Example:** `Switch(SyncE)# setSyncEglobalConfig synce-global-conf SSM-QL-for-holdover QL-EEC1`	• **QL-DNU—**SSM QL value is QL-DNU. • **QL-EEC1—**SSM QL value is QL-EEC1. • **QL-EEC2—** SSM QL value is QL-EEC2. • **QL-INV—**SSM QL value is QL-INV. • **QL-NONE—**SSM QL value is QL-NONE. • **QL-PRC—**SSM QL value is QL-PRC. • **QL-SSUA—**SSM QL value is QL-SSUA. • **QL-SSUB—**SSM QL value is QL-SSUB.
Step 6	**setSyncEglobalConfig synce-global-conf SSM-QL-for-freerun** {**QL-DNU** | **QL-EEC1** | **QL-EEC2** | **QL-INV** | **QL-NONE** | **QL-PRC** | **QL-SSUA QL-SSUB**} **Example:** `Switch(SyncE)# setSyncEglobalConfig synce-global-conf SSM-QL-for-freerun QL-EEC2`	Transmits SSM QL value when clock selector is in Free Run Mode. • **QL-DNU—**SSM QL value is QL_DNU. • **QL-EEC1—**SSM QL value is QL_EEC1. • **QL-EEC2—** SSM QL value is QL_EEC2. • **QL-INV—**SSM QL value is QL_INV. • **QL-NONE—**SSM QL value is QL_NONE. • **QL-PRC—**SSM QL value is QL_PRC. • **QL-SSUA—**SSM QL value is QL_SSUA. • **QL-SSUB—**SSM QL value is QL_SSUB.
Step 7	**setSyncEglobalConfig synce-global-conf EEC-Option**{**EEC1** | **EEC2**} **Example:** `Switch(SyncE)# setSyncEglobalConfig synce-global-conf EEC-Option EEC2`	Selects PLL EEC option. • **EEC1—**DPLL bandwidth is 3.5 Hz. • **EEC2—**DPLL bandwidth is 0.1 Hz.
Step 8	**setSyncEglobalConfig review** **Example:** `Switch(SyncE)# setSyncEglobalConfig review`	Displays the configuration.
Step 9	**setSyncEglobalConfig commit** **Example:** `Switch(SyncE)# setSyncEglobalConfig commit`	Sends the configuration to the NID.
Step 10	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# setSyncEglobalConfig synce-global-conf wait-to-restore 1
Switch(SyncE)# setSyncEglobalConfig synce-global-conf clock-select-config revertive
Switch(SyncE)# setSyncEglobalConfig synce-global-conf SSM-QL-for-holdover QL-EEC1
Switch(SyncE)# setSyncEglobalConfig synce-global-conf SSM-QL-for-freerun QL-EEC2
Switch(SyncE)# setSyncEglobalConfig synce-global-conf EEC-Option EEC2
Switch(SyncE)# setSyncEglobalConfig review

Commands in queue:
    setSyncEglobalConfig synce-global-conf wait-to-restore 1
    setSyncEglobalConfig synce-global-conf clock-select-config revertive
    setSyncEglobalConfig synce-global-conf SSM-QL-for-holdover QL-EEC1
    setSyncEglobalConfig synce-global-conf SSM-QL-for-freerun QL-EEC2
    setSyncEglobalConfig synce-global-conf EEC-Option EEC2

Switch(SyncE)# setSyncEglobalConfig commit

    SetSyncEglobalConfig Commit Success!!!

Switch(SyncE)# exit
```

Viewing SyncE Timers

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**getSyncEglobalConfig get-global-config** **Example:** `Switch(SyncE)# getSyncEglobalConfig get-global-config`	Displays the SyncE global configuration.
Step 3	**getSyncEglobalConfig review** **Example:** `Switch(SyncE)# getSyncEglobalConfig review`	Sends the configuration to the NID.
Step 4	**getSyncEglobalConfig commit** **Example:** `Switch(SyncE)# getSyncEglobalConfig commit`	Sends the configuration to the NID.
Step 5	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# getSyncEglobalConfig get-global-config
Switch(SyncE)# getSyncEglobalConfig review

Commands in queue:
  getSyncEglobalConfig get-global-config

Switch(SyncE)# getSyncEglobalConfig commit

  GetSyncEglobalConfig_Output.synce_global_conf.clock_select_config.u.revertive = '0'
  GetSyncEglobalConfig_Output.synce_global_conf.wait_to_restore = 1
  GetSyncEglobalConfig_Output.synce_global_conf.SSM_QL_for_holdover.t = 6
  GetSyncEglobalConfig_Output.synce_global_conf.SSM_QL_for_holdover.u.QL_EEC1 = '0'
  GetSyncEglobalConfig_Output.synce_global_conf.SSM_QL_for_freerun.t = 1
  GetSyncEglobalConfig_Output.synce_global_conf.SSM_QL_for_freerun.u.QL_NONE = ''
  GetSyncEglobalConfig_Output.synce_global_conf.EEC_Option.t = 1
  GetSyncEglobalConfig_Output.synce_global_conf.EEC_Option.u.EEC1 = ''

  GetSyncEglobalConfig Commit Success!!!

Switch(SyncE)# exit
```

Understanding ANEG Mode

The Auto-negotiation (ANEG) mode is relevant for 1000BaseT ports only. To recover the clock from a port, the clock must be negotiated to the Slave mode. To distribute the clock, the port must be negotiated to the Master mode.

Following are the different ANEG modes that can be activated on a clock source port:

- **Prefer Slave—**the port negotiates to the Slave mode.
- **Prefer Master—**the port negotiates to the Master mode.
- **Forced Slave—**the port is forced to the Master mode.

Note The port in the **Locked** state always remains negotiated to the **Slave**.

Configuring ANEG mode

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the syncE mode.

	Command or Action	Purpose

Step 2	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **port** *Physical port*}
	
	Example:
	`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1`
	`setSyncEclockConfig clock-sel-config occure source-configs 2`
		Configures the clock source on the port.
		• *source_configs*—Nominate a port number to be the clock source. The range is from 1 to 2.
		• *Physical port*—Physical port. The range is from 1 to 6.

Step 3	**setSyncEclockConfig clock-sel-config** {**source-configs** *source-configs* **aneg-mode** {**forced-slave** | **none** | **prefer-master** | **prefer-slave**}}
	
	Example:
	`Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1 aneg-mode prefer-master`
	`setSyncEclockConfig clock-sel-config occure source-configs 2 aneg-mode prefer-master`
		Configures the ANEG mode that is relevant to ports1 and 2, which are 1000 base T.

Step 4	**setSyncEclockConfig review** **Example:** `Switch(SyncE)# setSyncEclockConfig review`	Displays the configuration.
Step 5	**setSyncEclockConfig commit** **Example:** `Switch(SyncE)# setSyncEclockConfig commit`	Sends the configuration to the NID.
Step 6	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1
    setSyncEclockConfig clock-sel-config occur source-configs port 2
Switch(SyncE)# setSyncEclockConfig clock-sel-config source-configs 1 aneg-mode prefer-master

   setSyncEclockConfig clock-sel-config occur source-configs port 2 aneg-mode prefer-master
Switch(SyncE)# setSyncEclockConfig review

Commands in queue:
  setSyncEclockConfig clock-sel-config source-configs 1
    setSyncEclockConfig clock-sel-config occur source-configs port 2
  ssetSyncEclockConfig clock-sel-config source-configs 1 aneg-mode prefer-master
   setSyncEclockConfig clock-sel-config occur source-configs port 2 aneg-mode prefer-master


Switch(SyncE)# setSyncEclockConfig commit

 SetSyncEclockConfig Commit Success!!!
```

```
Switch(SyncE)# exit
```

Verifying SyncE Status

DETAILED STEPS

	Command or Action	Purpose
Step 1	**syncE** **Example:** `Switch# SyncE`	Enters the SyncE mode.
Step 2	**showNetworkClock show-synce-status** **Example:** `Switch(SyncE)# showNetworkClock show-synce-status`	Displays the SyncE status.
Step 3	**exit** **Example:** `Switch(SyncE)# exit`	Exits the SyncE mode.

Configuration Example

```
Switch# SyncE
Switch(SyncE)# showNetworkClock show-synce-status

  ShowNetworkClock_Output.show_network_clock.selector_state.t = 2
  ShowNetworkClock_Output.show_network_clock.selector_state.u.holdover = ''
  ShowNetworkClock_Output.show_network_clock.alarm_state[0].clock_source = 1
  ShowNetworkClock_Output.show_network_clock.alarm_state[0].LOCS = false
  ShowNetworkClock_Output.show_network_clock.alarm_state[0].SSM = false
  ShowNetworkClock_Output.show_network_clock.alarm_state[0].WTR = false
  ShowNetworkClock_Output.show_network_clock.alarm_state[1].clock_source = 2
  ShowNetworkClock_Output.show_network_clock.alarm_state[1].LOCS = true
  ShowNetworkClock_Output.show_network_clock.alarm_state[1].SSM = false
  ShowNetworkClock_Output.show_network_clock.alarm_state[1].WTR = false
  ShowNetworkClock_Output.show_network_clock.alarm_state[2].clock_source = 3
  ShowNetworkClock_Output.show_network_clock.alarm_state[2].LOCS = true
  ShowNetworkClock_Output.show_network_clock.alarm_state[2].SSM = false
  ShowNetworkClock_Output.show_network_clock.alarm_state[2].WTR = false

  ShowNetworkClock Commit Success!!!

Switch(SyncE)# exit
```