*Note: Especially, no assumptions are made on mutually synchronous or asynchronous clocking of the host signals nor on the network signal being asynchronous to the host signals.*

### 7.6.2 Data Path, Network Path and Host Paths

The **Data Path** of a client encapsulation application, like the Data Path of a system interface application, consists of all involved host and media lanes, including all related module internal resources.

However, unlike the Data Path of system interface applications described in chapter 6, where the Data Path is also a unit of independent initialization, usage, and deinitialization, the Data Path of a N:1 multiplexing client encapsulation application is partitioned into N host side Host Paths (see below) and a single media side Network Path (see below), with finer granularity of initialization, usage, and deinitialization governed by a set of N+1 parallel state machines (one per Host Path and one for the Network Path), rather than by a single DPSM.

**Network Path**

The media side network signal related transmit-and-receive functions of a **multiplex** or **uniplex** client encapsulation application are realized by a **Network Path** (**NP**) that spans the segment between the internal multiplex connection points to the media lane physical interfaces.

Like the media side of a system interface Data Path, an NP can use one or more media lanes, employing single-carrier or multi-carrier transmission.

A Network Path is controlled using a **separate** set of Network Path related **registers** and its dynamic state of configuration is represented by a Network Path State Machine (**NPSM**)

**Host Path**

The host side host signal related transmit-and-receive functions of one host signal in a client encapsulation application are realized by a **Host Path** (HP) that spans the segment between internal multiplex connection points and the host lane physical interfaces.

Each **Host Path** is controlled using the host side related **Data Path registers** for its constituent lanes and its dynamic state of configuration is represented by a DPSM that is effectively restricted to act on the relevant Host Path segment of the Data Path only (by limiting the achievable states to DPDeactivated and DPInitialized).

*Note: In a certain sense, all literal, textual references to host side aspects of Data Paths in chapter 6, actually refer only to a Host Path segment of the Data Path of an NP Application, when the relevant lane is part of a NP Application and hence a HP lane. Beware of confusion. See also section 7.6.6.*

Where distinction is needed, the combination of one HP and its associated NP is called a **partial** Data Path, because the host lanes of the other HPs in the NP Application are not included.

### 7.6.3 Network Path Applications

Due to the characteristic presence of an independently operated Network Path, client encapsulation applications are also referred to as **NP Applications**. Likewise, for ease of distinction, system interface applications (without client encapsulation) are now also referred to as **DP Applications**.

A **uniplex** NP Application includes only one Host Path.

A **multiplex** NP Application includes as many Host Paths as there are multiplexed host signals.

A **homogeneous** (simple) NP Application encapsulates one type of host signals (characterized by a single HostInterfaceID).

A **mixed** (heterogeneous) NP Application encapsulates different types of host signals (each type characterized by a different HostInterfaceID).

*Note: Recall that this specification restricts the lane data rate to be identical for all Host Path lanes, not for essential reasons, but to simplify the advertisement of mixed multiplex capabilities and restrictions.*

One NP Application comprises one media side Network Path that is connected (internally, in the module) to one or more host side Host Paths (HPs) such that the information bandwidths of the Host Paths add up to the information bandwidth of the Network Path.

One NP Application instance is therefore defined as a specific association of one or more Host Interfaces comprising a group of host lanes (each associated with a separate Host Path) and a single Media Interface (associated with a single Network Path) comprising a group of media lanes.

A module may support several **alternative** NP applications, such as e.g. rate scaled multiplex applications, which are structurally identical but use a scaled host signal data rate.

A module may also support several NP Applications operating in **parallel** where each NP Application requires only a subset of the lanes available in the module.

### 7.6.4 Network Path Application Advertisement

As with Data Path Applications, Application Descriptors (see section 6.2.1.4) are used to describe all possible instantiations of an NP Application, expressed from a host side perspective in terms of host lanes.

An Application Descriptor where the information bandwidth associated with the MediaInterfaceID is a multiple of the information bandwidth associated with the HostInterfaceID is a **partial** Application Descriptor.

Using an application descriptor extension, the module explicitly advertises for each Application Descriptor, if it advertises a DP Application or an NP Application (see section 8.15.5.3).

*Note: The partial Application Descriptors required to advertise genuine multiplex applications were semantically invalid in CMIS 5.0. CMIS 5.0 hosts may therefore be confused by CMIS 5.1 modules, if they sanity test for equal data rate on host and media side, or if they do not fully match desired versus advertised Applications.*

**Uniplex NP Applications**
Structurally, the Application Descriptor for a uniplex NP Application cannot be distinguished from a DP Application Descriptor; since the meaning associated with the MediaInterfaceID does not always allow the distinction, application advertisement extensions are used to disambiguate.

**Homogeneous Multiplexing NP Applications**
The Application Descriptor for a homogeneous multiplexing NP Application can be recognized by the fact that the information bandwidth of the host side signal (identified by HostInterfaceID) is lower than the information bandwidth of the media side signal (identified by MediaInterfaceID), i.e. it is a partial Application Descriptor

*Note: Normally, the MediaInterfaceID information bandwidth is an integer multiple of the HostInterfaceID information bandwidth. When this is not the case, the host side signal cannot be used in a homogeneous multiplexing NP Application but possibly only in a mixed multiplexing NP Application.*

**Mixed Multiplexing NP Applications**
A mixed multiplexing NP Application is described by a **set** of mutually **consistent** partial Application Descriptors, one for each type of host interface being part of the multiplex structure.

Two partial Application Descriptors are **consistent** when they use the same MediaInterfaceID and when they offer at least one identical option in the MediaLaneAssignmentOptions, as well as at least one set of conflict-free HostLaneAssignmentOptions.

The module advertises separately (see section 8.15.5.5) which lane groups and bandwidth granularities can be mixed to form the Host Paths of the mixed multiplexing NP Application, subject to restrictions described in section 7.6.1.5.

### 7.6.5 Network Path Application Instance Configuration

Application Descriptors and associated extensions (see section 8.15.5.3) advertise all potential instantiations of DP or NP Applications that are supported by a module supporting NP Applications.

The configuration of a particular DP Application instance is described in sections 6.2.1 through 6.2.4, whereas the configuration and control of a particular NP Application instance is described in the following.

#### 7.6.5.1 Configuration and Control

The actual configuration of an NP Application Instance in terms of its host lanes, media lanes, and internal resources is determined by first **provisioning** (into Active Control Sets) and then **commissioning** (into hardware) a configuration previously **defined** in Staged Control Sets.

This procedure allocates the host lanes (of one or more Host Paths) and the media lanes (of a Network Path) to the NP Application Instance and internally connects the HP instances and the multiplex connection points of the NP, within the module.

The host lanes are assigned to Host Paths by using the regular Data Path related Staged Control Set registers (see section 6.2.3), and these host lanes are also assigned to the Network Path using a dedicated NP Staged Control Set (see section 8.15.2), i.e. the Host Paths and the Network Paths are associated via host lanes.

Like Data Path instances, also Host Path and Network Path instances are identified by reference to the lowest lane number of all assigned host lanes, called HP DPID and NPID, respectively.

The outputs (inputs) of a HP become associated to NP inputs (outputs) at internal multiplex connection points, in order of increasing host lane numbers (HP DPIDs), when the lanes of the HP (in the provisioned DPConfig) are also assigned to the NP (in the provisioned NPConfig).

In the overall configuration procedure, the group of Host Paths (HP) and the Network Path (NP) participating in an N:1 NP Application instance are defined, provisioned, and commissioned sequentially (i.e. in separate configuration commands), and the dynamically controlled states of all commissioned HPs (as represented in DSPM states) and of the NP (as represented in an NPSM state) are also observed and controlled separately.

The **definition**, **provisioning**, **commissioning**, and **control** of NP Application instances works as follows:

The HPs are **defined**, individually or as a group, in a Staged Control Set. Each HP is identified by reference to its first associated host lane number, called HP DPID, and all lanes of a HP must be assigned this HP DPID.

The NP is **defined** in an NP Staged Control Set. The NP is identified by reference to its first associated host lane number (of all tributary HPs), called NPID, and all lanes of NP must be assigned this NPID.

HPs and NP are then (requested to be) **provisioned** into the relevant Active Control Set or NP Active Control Set in separate steps, by writing to the associated ApplyDPInit trigger register or ApplyNPInit register, respectively. As usual, the provisioning procedure includes (partial) validation prior to copying a Staged Control Set into the associated Active Control Set.

*Note: Since the order of HP and NP provisioning is unspecified, and since the time of completing the provisioning is unknown, the module can only validate certain aspects during each of the provisioning commands.*

*Note: A future version might add a command to validate the currently provisioned configuration to allow on demand validation of a completely provisioned NP Application prior to commissioning at a time when the host knows that its provisioning is complete.*

The sequential **commissioning** of provisioned HPs and provisioned NP is **controlled** via the **DPDeinit** and **NPDeinit** registers, causing DPInitPending and NPInitPending indicators to be set, respectively.

*Note: Prior to the commissioning a configured NP Application Instance a suitable multiplex structure must be fully provisioned into the DP and NP Active Control Sets. However, some of the multiplex connection points may actually remain unused (i.e. the relevant host signals of the HP connected to the multiplex connection point are intentionally not present) by keeping the relevant DPSMs in the DPDeactivated state.*

The actually commissioned **configuration** of HPs and NP can eventually be **observed** in the Active Control Set and in the NP Active Control Set after the DPInitPending and NPInitPending indicators are cleared, while current **states** can be **observed** in the DPState and NPState registers, respectively.

*Note: See also Appendix H for configuration and control examples.*

#### 7.6.5.2 Reconfiguration

The multiplex structure of a NP can be changed by selective HP reconfiguration and, optionally, NP source reconfiguration, without disrupting the transmission of unchanged Host Paths carrying traffic.

*Note: This can be achieved because the NPSM and the HP DPSMs are decoupled.*

### 7.6.6 Modifications and Extensions of Prior Specifications for Data Path Applications

The textual specifications for DP Applications in chapter 6 are generally applicable to the Host Paths of NP Applications, with the following modifications assumed but not spelled out in the text.

Register specifications in chapter 8 referring to the host side aspects of Data Paths apply also to Host Paths.

#### 7.6.6.1 Host Lanes

All statements about the host side aspects of Data Paths apply also to Host Paths.

All register specifications referring to the host side aspects of Data Paths apply also to Host Paths.

#### 7.6.6.2 Data Path State Machine (DPSM)

Each Host Path is controlled by a slightly modified Data Path State Machine (DPSM) in which the transition signal **DPDeactivateS** is **forced to TRUE**, limiting the steady states achievable to DPDeactivated and DPInitialized.

*Note: This reflects the fact that the media side functions of a NP Application are represented by the Network Path, not by the Host Path, and therefore controlled by the NPSM.*

Formally this is achieved by adding a configuration dependent term **NPInUseT** to the DPDeactivateS equation

**DPDeactivateS** = DPReDeinitS OR DPTxDisableT OR DPTxForceSquelchT OR NPInUseT (Eq. 7-1)

where:

**NPInUseT** is defined to be TRUE for a Host Path lane and FALSE otherwise.

Formally, the presence of a Host Path (instead of a Data Path) is derived from NPInUseLane<i>, where <i> is the ID of the Data Path or Host Path (see Table 8-125).

The media side Network Path is not controlled by a DPSM, but instead by a Network Path State Machine (NPSM) as described in section 7.6.7.

#### 7.6.6.3 Media Lanes

All statements made about Media Lanes of Data Paths need to be reinterpreted as statements about Media Lanes of Network Paths.

Unlike for DP Applications, in NP Applications the conditions for Tx output signals to be valid do not depend on host signal conditions in the module.

#### 7.6.6.4 Squelching

Squelching related functions for the HPs of a NP application in the **Rx** direction are supported when they are supported for DP applications.

Core CMIS uses only physical signal defects (LOS) as triggers for automatic squelching in the Rx direction, but depending on the MediaInterfaceID, Network Paths may come with additional conditions detected in digital signal processing layers that signify the absence of a genuine host signal to be forwarded. Such conditions are not specified here but are assumed to contribute to Automatic Squelching if available, and if enabled.

Squelching related functions for the HPs of a NP application in the **Tx** direction are supported when they are supported for DP applications, with the strict exception of auto-squelching.

*Note: In an NP application, consequent actions in case of a multiplexed host signal being unavailable are assumed to be specified in the pertinent transmission standards.*

#### 7.6.6.5 Configuration

There is no intervention-free reconfiguration for Network Paths nor for Host Paths, i.e. only stepped reconfiguration functionality is available for Host Paths of multiplex Data Paths, independent of the SteppedConfigOnly setting for regular Data Paths.

### 7.6.7 Network Path State Machines (NPSM)

A **Network Path State Machine** (**NPSM**) instance describes Network Path-specific behaviors and properties that are related to the configuration of the media side **Network Path**, as managed by the host.

*Note: The NPSM state represents a* ***management*** *or* ***configuration realization status*** *of a Network Path, representing the effects of certain host configuration commands and of module reactions to those commands. It* ***does not*** *necessarily represent other behavioral or operational aspects of a Network Path, e.g. in terms of current input or output signal conditions or in terms of transmission service being provided.*

*Note: The NPSM state should neither be confused with the* ***operational status*** *of the functional resources of a Network Path (in Tx direction or in Rx direction) nor with the resulting signal* ***output status*** *of Rx host lane outputs or of Tx media lane outputs.*

**Module State and NPSM Life Cycle**

All **NPSM** instances required to represent the power-up default Application defined in the Network Path Configuration field values of the Active Control Set are initially created and set-up during the **MgmtInit** state.

After its creation, a NPSM remains in the NPDeactivated State until the Module State Machine is in the **ModuleReady** state and an exit condition from the NPDeactivated state is met.

When the host updates the Network Path Configuration fields in the Active Control Set, in either the **ModuleLowPwr** or **ModuleReady** states, the module tears down any previous NPSM that is no longer defined and then creates and sets up any newly defined NPSM.

All Network Path State Machines are torn down in the **Resetting** state.

**NPSM Purpose**

A Network Path State Machine is used by the module to represent the initialization status of the resources associated with a Network Path in response to certain host configuration settings or commands.

Although individual resources within a Network Path may complete initialization activities at different times, the module waits to report the updated NPSM state until all resources associated with the Network Path have completed the requested configuration or reconfiguration action. This synchronized status reporting across all lanes and resources in a Network Path reflects the fact that there is only one NPSM per Network Path.

**NPSMs for parallel Network Paths of Multiple Application Set Instances**

Each Network Path in a module is required to operate independently of other Network Paths: if the host changes the Network Path State of one Network Path, the other Network Paths in the module shall be unaffected and uninterrupted.

#### 7.6.7.1 NPSM State Transition Diagram and NPSM Specification

Figure 7-6 shows the state transition diagram (STD) of a NPSM representing the Network Path configuration related state of one Network Path instance.

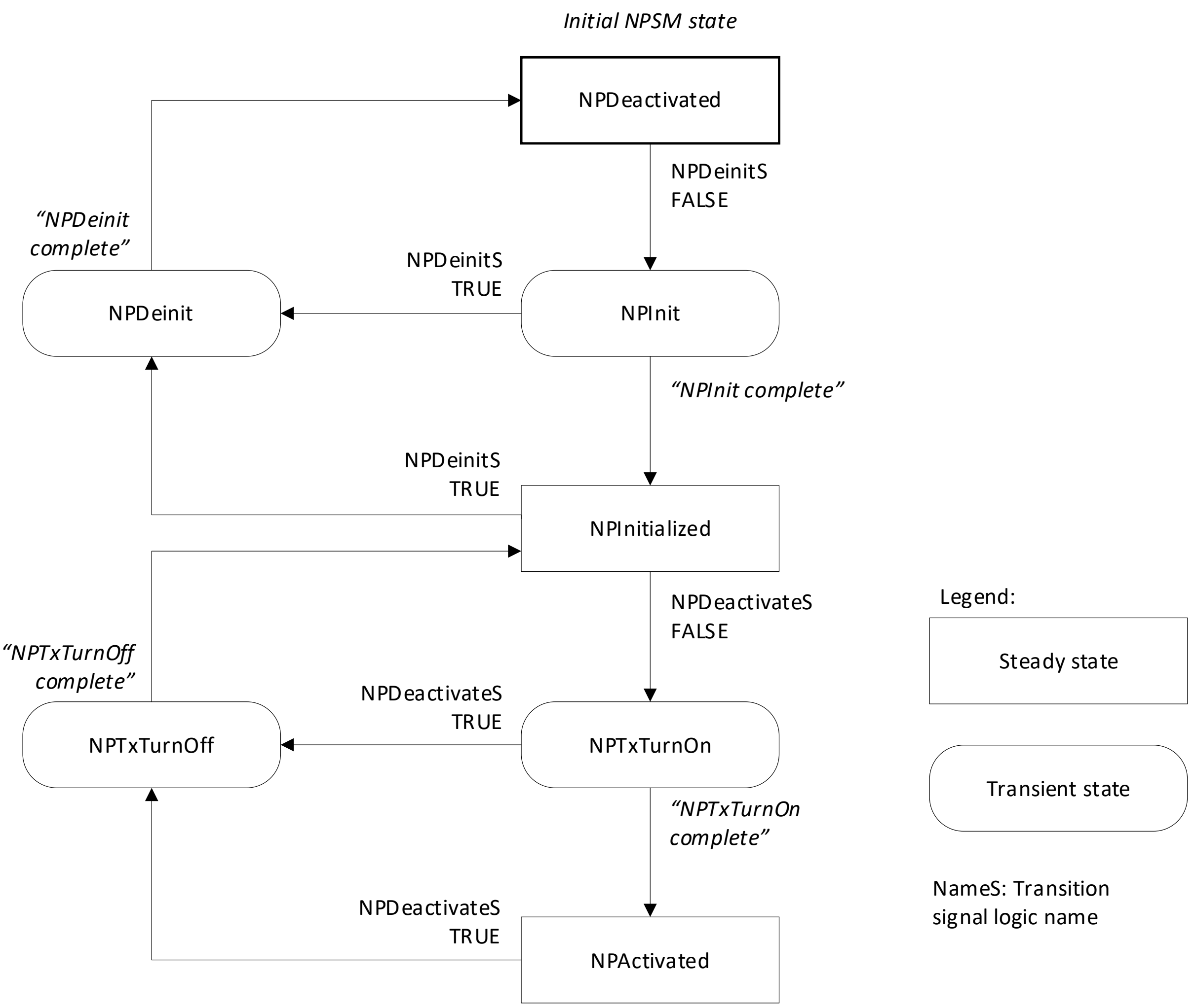


**Figure 7-6 Network Path State Machine (NPSM) State Transition Diagram**

*Note: Prior to exit from the MgmtInit module state, all Network Paths initialize to the NPDeactivated state.*

**State Exit Conditions and Transition Signals**

The state machine exits a given state when specific exit conditions are satisfied. So called **Transition Signals** (recognized by name suffix S) represent these exit conditions for steady states.

**NPDeinitS**

The NPDeinitS transition signal can also be represented by the logic equation

**NPDeinitS** = ( NOT NPInUseT ) OR ( NOT ModuleReadyT ) OR LowPwrS OR NPDeinitT (Eq. 7-2)

where:

**ModuleReadyT** = (ModuleState = ModuleReady) (Eq. 7-3)

**NPDeinitT** = NPDeinitLane<N> (Eq. 7-4)
OR NPDeinitLane<N+1>
...
OR NPDeinitLane<N+M-1>

**NPInUseT** = NPInUseLane<N> (Eq. 7-5)
OR NPInUseLane<N+1>

...
OR NPInUseLane<N+M-1>

N = first host lane in the Network Path
M = number of host lanes in the Network Path

**NPDeactivateS**

The NPDeactivateS transition signal is defined by the logic equation

**NPDeactivateS** = NPDeinitS OR NPTxDisableT OR NPTxForceSquelchT (Eq. 7-6)

where:

**NPTxDisableT** = OutputDisableTx<N> (Eq. 7-7)
OR OutputDisableTx<N+1>
...
OR OutputDisableTx<N+M-1>

**NPTxForceSquelchT** = OutputSquelchForceTx<N> (Eq. 7-8)
OR OutputSquelchForceTx<N+1>
...
OR OutputSquelchForceTx<N+M-1>

N = first media lane in the Network Path
M = number of media lanes in the Network Path

**Reaction to Module Reset**

When the MSM **ResetS** transition signal (see Table 6-11) becomes TRUE, any Network Path activities related to power down are performed in the **Resetting** module state. The NPSM state machines then ceases to exist.

*Note: Module dependent pre-reset clean up and power down activities may be implemented, possibly depending also on the reset trigger in either hardware or software.*

**Reaction to Module Fault**

When Module State Machine transitions to **ModuleFault** state, the NPSM behavior is not defined formally, but governed by the behavioral requirements of the ModuleFault state.

### 7.6.7.2 Network Path Control (Host)

A single main configuration register is provided for the host to control initialization and deinitialization of all Network Paths represented in a given Bank. This **NPDeinit** register (see Table 8-128) defines per host lane if the associated Network Path resources are determined to be unused for functional operation (and hence can be deinitialized) or if they are determined for functional operation (and hence need to be initialized).

*Note: Initialization status and behavior of Tx media lane outputs are further controlled by the host using the media lane specific control bits **OutputDisableTx**<i> and **OutputSquelchForceTx**<i>.*

A host requesting initialization or deinitialization of a Network Path ensures that the Active Control set contains the desired configuration settings and then writes the value 0 or 1, respectively, to the DPDeinit bits associated with the host lanes of that Network Path.

The host may request initialization or deinitialization of multiple Network Paths with one register access.

### 7.6.7.3 Network Path Status (Module)

The module provides information on the current state of the Network Path (NPSM **current state reporting**) and on entry to certain NPSM states (NPSM **state change indication**).

**NPSM Current State Reporting**

On entry to a NPSM state the module reports the NPSM state entered as the current NPSM state in the **NPState** status register (see Table 8-136

Table 8-135), on all lanes of the Network Path, with optional exceptions specified below.

*Note: Due to identical behavior of all lanes feeding a Network Path the host needs to read only the first lane of the Network Path to determine the Network Path state.*

**NPSM Current State Reporting Exceptions**

The module **may** suppress reporting the current NPSM state in the NPStateHostLane<i> registers when that state is known to be transitional, i.e. when it is exited immediately because its exit conditions are fulfilled on entry, or when the duration of staying in that state is known to be in the order of 1 ms or less.

*Note: The duration specification is intentionally vague. The intention for allowing exceptions in state reporting is to avoid reporting short-lived status data which the host is unlikely to read and react upon.*

**NPSM State Change Indication (Flag)**

A NPSM State Change Indication consist of the module setting a **NPStateChangedFlag** for each lane of the Network Path associated with the relevant NPSM instance.

*Note: The intention of the following specification is that the module indicates a state change only on entry to a lasting steady state and only when the transition time since the previous lasting steady state was significant.*

The maximum duration of a transient state is advertised in a MaxDuration* field (see Table 8-138) and is considered **insignificant** when the coded MaxDuration* field value is 0000b (see Table 8-43), and **significant** otherwise.

The module issues a NPSM State Change Indication on entry to a **steady** NPSM state when no exit condition of the entered **steady** state is fulfilled on state entry (state is not visited transitorily) [1].

The module does **not** perform a NPSM State Change Indication on entry to a **transient** NPSM state.

Table 6-19 defines the Flag behavior for each NPSM state entry.

**Table 7-5 Network Path State Changed Flag behaviors**

| Entered state | NPStateChangedFlag may be set * |
|---|---|
| NPInit | No |
| **NPInitialized** | **Yes** |
| NPDeinit | No |
| NPDeinit | No |
| **NPDeactivated** | **Yes** |
| NPTxTurnOn | No |
| **NPActivated** | **Yes** |
| NPTxTurnOff | No |
| NPTxTurnOff | No |
| **NPInitialized** | **Yes** |

* *Note: The Flag setting conditions are described in the main text.*

*Note: Steady state exit conditions may already be met upon entry into the steady state and lead to immediate transition to the next state (after state entry or state exit activities, if defined).*

**Flag Related Behavior**

The module does not clear any Flag due to a state change of a Network Path State Machine.

The module raises Flags only according the NPSM state-specific conformance rules defined in section 7.6.8.

#### 7.6.7.4 Detailed State Descriptions

The DPSM state descriptions in the subsections of section 6.3.3 apply similarly to the NPSM states, with the following exceptions:

Intervention-free reconfiguration of the NP is not supported.

[1] Note the deliberate difference to the DPSM behavior specification in section 6.3.3.3.

### 7.6.8 Network Path Dependent Flagging Conformance

#### 7.6.8.1 Lane-Specific Flagging Conformance per NPSM State

Network Path(s) adhere to the following NPSM flagging conformance rules.

Table 7-6 and Table 7-7 describe the Flagging conformance for media lane-specific Flags, per NPSM state.

*Note: The Network Path Flagging Conformance tables are limited to flags or flag groups and VDM observables that are related to the media interface.*

**Flag Setting Restrictions**

While in an NPSM state where a Flag is indicated as **N/A** (not allowed), the module shall not set that Flag.

All media lane-specific Flags are generally N/A throughout the **Reset** and **MgmtInit** MSM states. For all other MSM states, the NPSM State determines media lane-specific Flagging conformance.

*Note: For Flags allowed in a NPSM state, additional and more specific rules may exist*

*Note: The host can suppress undesirable Interrupts by setting the corresponding Mask bit at any time after the management interface is initialized.*

**Flag Specification Conformance**

The setting of allowed alarm and warning Flags of Network Path related monitors including associated Interrupt generation is only assured in the NPInitialized and NPActivated states.

**Table 7-6 Lane-Specific Flagging Conformance Rules per NPSM State**

| Flag / Flag Group[1] | Page | Byte | NPDeactivated | NPInit | NPDeinit | NPInitialized | NPTxTurnOn | NPTxTurnOff | NPActivated |
|---|---|---|---|---|---|---|---|---|---|
| Network Path Related Flags | | | | | | | | | |
| NPStateChangedFlag* | 17h | 128 | allowed | N/A | N/A | allowed | N/A | N/A | allowed |
| Tx Media Related Flags | | | | | | | | | |
| OpticalPowerHighAlarmFlagTx* | 11h | 139 | allowed | allowed | allowed | allowed | allowed | allowed | allowed |
| OpticalPowerLowAlarmFlagTx* | 11h | 140 | N/A | N/A | N/A | allowed | allowed | allowed | allowed |
| OpticalPowerHighWarningFlagTx* | 11h | 141 | allowed | allowed | allowed | allowed | allowed | allowed | allowed |
| OpticalPowerLowWarningFlagTx* | 11h | 142 | N/A | N/A | N/A | allowed | allowed | allowed | allowed |
| LaserBiasHighAlarmFlagTx* | 11h | 143 | allowed | allowed | allowed | allowed | allowed | allowed | allowed |
| LaserBiasLowAlarmFlagTx* | 11h | 144 | N/A | N/A | N/A | allowed | allowed | allowed | allowed |
| LaserBiasHighWarningFlagTx* | 11h | 145 | allowed | allowed | allowed | allowed | allowed | allowed | allowed |
| LaserBiasLowWarningFlagTx* | 11h | 146 | N/A | N/A | N/A | allowed | allowed | allowed | allowed |
| Rx Media Related Flags | | | | | | | | | |
| LOSFlagRx* | 11h | 147 | allowed | allowed | allowed | allowed | allowed | allowed | allowed |
| CDRLOLFlagRx* | 11h | 148 | N/A | N/A | N/A | allowed | allowed | allowed | allowed |
| OpticalPowerHighAlarmFlagRx* | 11h | 149 | allowed | allowed | allowed | allowed | allowed | allowed | allowed |
| OpticalPowerLowAlarmFlagRx* | 11h | 150 | N/A | N/A | N/A | allowed | allowed | allowed | allowed |
| OpticalPowerHighWarningFlagRx* | 11h | 151 | allowed | allowed | allowed | allowed | allowed | allowed | allowed |
| OpticalPowerLowWarningFlagRx* | 11h | 152 | N/A | N/A | N/A | allowed | allowed | allowed | allowed |

#### 7.6.8.2 VDM Flagging Conformance per NPSM State

Table 7-7 describes the Flag conformance for all Flags related to the (optional) Versatile Diagnostic Monitoring (VDM) Observables, per NPSM State. See section 7.1 and section 8.17 for information on the VDM feature.

In NPSM States where a Flag is indicated as 'Not Allowed', the module shall not set the associated Flag bit while the Network Path is in that state.

All VDM Flags are 'Not Allowed' throughout the Reset and MgmtInit module states. For all other module states where a NPSM is in use, implementers should refer to the Network Path State to determine lane-specific Flag conformance.

[1] An asterisk '*' in a name is a wildcard: All Flags matching the name pattern are referred to.

**Table 7-7 VDM Flag Conformance Rules per NPSM State**

| VDM Observable Type | NPDeactivated | NPInit | NPDeinit | NPInitialized | NPTxTurnOn<br>NPTxTurnOff<br>NPActivated |
|---|---|---|---|---|---|
| Laser Age | N/A | allowed | allowed | allowed | allowed |
| TEC Current | allowed | allowed | allowed | allowed | allowed |
| Laser Frequency Error | N/A | allowed | allowed | allowed | allowed |
| Laser Temperature | N/A | allowed | allowed | allowed | allowed |
| eSNR Media Input | allowed | allowed | allowed | allowed | allowed |
| PAM4 Level Transition Parameter (LTP) Media Input | N/A | allowed | N/A | allowed | allowed |
| Pre-FEC BER Minimum Media Input (data-path) | N/A | allowed | N/A | allowed | allowed |
| Pre-FEC BER Maximum Media Input | N/A | allowed | N/A | allowed | allowed |
| Pre-FEC BER Average Media Input | N/A | allowed | N/A | allowed | allowed |
| Pre-FEC BER Current Value Media Input | N/A | allowed | N/A | allowed | allowed |
| FERC Minimum Sample Media Input | N/A | allowed | N/A | allowed | allowed |
| FERC Maximum Sample Media Input | N/A | allowed | N/A | allowed | allowed |
| FERC Sample Average Media Input | N/A | allowed | N/A | allowed | allowed |
| FERC Current Sample Value Media Input | N/A | allowed | N/A | allowed | allowed |
| FERC Total Accumulated Media Input | N/A | allowed | N/A | allowed | allowed |

## 7.7 Unidirectional Hot Data Path Reconfiguration

Support for fast, intervention-free, **unidirectional hot reconfiguration** of certain Data Path attributes in the current DPSM state (see section 6.2.4) is motivated by requirements of certain link speed negotiation protocols, as used, e.g., in Fibre Channel applications, which require to temporarily switch unidirectionally between similar Applications at different speeds, while retaining the lane allocation of the Data Path.

*Note: Colloquially, this unidirectional reconfiguration feature is also known as Fibre Channel support.*

**Advertisement**

Support for unidirectional reconfiguration is advertised by the UnidirReconfigSupported bit (see Table 8-48).

**Commands**

The fastest intervention-free hot reconfiguration employs switching between two prepared Staged Control Sets.

*Note: Support for the 2nd Staged Control Set is advertised independently of UnidirReconfigSupported, but typical applications of unidirectional control will require two Staged Control Sets, for speed purposes.*

For the fastest intervention-free **bidirectional** hot reconfiguration, the basic ApplyImmediate commands (Table 8-63 and Table 8-69) can be used.

For **direction-specific** intervention-free hot reconfiguration, two optional **unidirectional** hot reconfiguration commands are introduced, **ApplyImmediateTx** and **ApplyImmediateRx** (see Table 8-68 and Table 8-73), that restrict the effect of a hot reconfiguration command (including provisioning and commissioning) that the host has initiated by applying a Staged Control Set, to the Tx or Rx direction, respectively.

*Note: To further speed-up the overall reconfiguration during time-critical link speed negotiations, some module form factors may even offer hardware control signals to trigger unidirectional reconfiguration commands, instead of writing to the CMIS trigger registers. The management of such 'rate or speed select Rx/Tx' functions (advertisement, administration, reporting) as well as the encoding and behavior of such control signals is outside of the scope of CMIS and is therefore left to form factor specific CMIS adjoint specifications. See section 8.8.*

Unidirectional hot reconfiguration command triggers are effective only in the **DPInitialized** or in the the **DPActivated** steady states of the relevant DPSM. The module (silently) ignores them in all other states.

*Note: This implies that the original Data Path on first entry to DPInitialized is Rx/Tx symmetrical.*

Unidirectional hot reconfiguration of a Data Path Application is **restricted** to changing the AppSel code consistently on all lanes of the Data Path, and to optionally changing Explicit Control parameters (SI attributes) on some or all lanes of the Data Path. The lane allocation to the Data Path must not be modified.

Unidirectional hot reconfiguration of NP Applications is not supported.

**Command Parameters**

The command parameters for a unidirectional target reconfiguration are defined as usual in one of the supported Staged Control Sets and are selected by using the Apply registers associated with that Staged Control Set.

**Command Validation, Execution, and Results**

The positive or negative results of a unidirectional reconfiguration command and the reconfiguration command execution status (in progress) is found in the Configuration Command Status register (see Table 8-83).

In case of command parameter validation failure, the module shall not update the Active Control Set and not commit any changes to hardware.

A host shall allow a triggered immediate reconfiguration command to complete in the current state before triggering any DPSM state transition.

**Committed Status Information**

To represent the provisioned (and eventually also commissioned) configuration per direction, two new data structures are introduced **ACS::DPConfigTx** (see Table 8-149) and **ACS::DPConfigRx** (see Table 8-148), in addition to the SI settings which are already direction specific (see Table 8-87 and Table 8-88).

The classical Active Control Set (see Table 8-86) always represents the Tx configuration (which is always identical to the Rx configuration if unidirectional reconfiguration is not used).

*Note: Hosts not using unidirectional reconfiguration facilities supported by a module may safely ignore the directional Active Control Sets and use the classical Active Control Set instead.*

## 8.15 Banked Page 16h (Network Path Functionality)

Page 16h is an optional Page supporting the optional Network Path (**NP**) functionality that is required for multiplex or uniplex client encapsulation applications.

Concept and functionality of Network Paths and NP Applications is described in section 7.6.

The module advertises support of Page 16h (and Page 17h) in Bit 01h:142.7 (see Table 8-41).

Page 16h may optionally be Banked. Each Bank of Page 16h refers to 8 lanes.

Page 16h is subdivided in subject areas as illustrated in the following table:

**Table 8-124 Page 16h Overview**

| **Byte** | **Size** (bytes) | **Subject Area** | **Description** |
|---|---|---|---|
| 128-159 | **32** | **Provisioning** | |
| 128-135 | 8 | NP Staged Control Set 0 | Lane to NP Assignments Provisioning – Staged Control Set 0 |
| 136-143 | 8 | NP Staged Control Set 1 | Lane to NP Assignments Provisioning – Staged Control Set 1 |
| 144-159 | 16 | - | **Reserved[16]** |
| 160-175 | **16** | **Control** | |
| 160 | 1 | NP Control | Network Path initialization control |
| 161 | 1 | - | **Reserved[1]** |
| 162-163 | 2 | NP Source Selectors | Signal source selection at the Network Path connection points |
| 164-175 | 12 | - | **Reserved[12]** |
| 176-191 | **16** | **Command & Response** | |
| 176 | 1 | NP Apply SCS 0 | Apply command for NP Staged Control Set 0 |
| 177 | 1 | NP Apply SCS 1 | Apply command for NP Staged Control Set 1 |
| 178-181 | 4 | Configuration Status | Status of most recent Network Path configuration command |
| 182-191 | 10 | - | **Reserved[10]** |
| 192-223 | **32** | **Status** | |
| 192-199 | 8 | NP Active Control Set | Provisioned Network Path Configuration |
| 200-203 | 4 | Network Path Status | Network Path State Machine state of each NP media lane |
| 204 | 1 | NPInitPending Condition | Commissioning status (NPInitPending condition) |
| 205-223 | 19 | - | **Reserved[19]** |
| 224-255 | **32** | **Advertisement** | |
| 224-225 | 2 | NPSM Max Durations | Maximum durations for all NPSM transient states |
| 226 | 1 | Options | Miscellaneous options |
| 227 | 1 | - | **Reserved[1]** |
| 228-247 | 20 | Mixed Multiplex Support | Advertising for mixed HP multiplexing support |
| 248-249 | 2 | Application Advertisement | Application Advertisement Extensions |
| 250-255 | 6 | - | **Reserved[4]** |

### 8.15.1 Network Path Provisioning

Network Path Provisioning fields allow the host to provision Network Paths into NP Active Control Sets, for subsequent commissioning into module hardware when the NP transits through the NPInit state of the NPSM.

*Note: See sections 6.2.3 and 6.2.4 for the core concepts and procedures of provisioning and commissioning Data Paths, which are very similar to those defined here (and in section 7.6.5), for Network Paths.*

There are two NP Staged Control Sets (see Table 8-126 and Table 8-127), both offering the same provisioning fields, allowing the host to prepare two different configurations to be provisioned on demand.

Parallel Network Paths can be provisioned when there are no host lane conflicts and when each NP carries one of the multiplex or uniplex NP Applications advertised by the module in the Application Descriptor registers (see Table 8-20, Table 8-52, and Table 8-53) and its extension (see Table 8-140), and in the Mixed Multiplex Advertisement registers (see Table 8-142 and Table 8-143).

*Note: See sections 7.6.4 and 8.15.5. for more information about NP Application advertising.*

Provisioning the Data Path of an NP Application with one Network Path serving a number N of Host Paths also requires provisioning the N Host Paths (using the Data Path configuration mechanisms). The AppSel fields of all Host Paths must refer to (partial) Application Descriptors that all advertise the same media interface ID, with throughput larger than the sum of the N host interface IDs associated with the N Data Paths.

The host provisions the NP configuration prepared in an NP Staged Control Set by writing to the Apply Trigger register of the relevant NP Staged Control Set (see Table 8-130 and Table 8-131), which triggers execution of a command to update the NP Active Control Set (see Table 8-134).

*Note: The NPID of all lanes belonging to host paths of the NP Application Data Path must be identical because they are all part of the same NP Application Data Path, and the NPInUse field of all those lanes must be set.*

The mechanism and command handling protocol to trigger execution of a provisioning command is fully analogous to the configuration of Data Paths or Host Paths (see section 6.2.3.3). However, intentionally only step by step configuration is supported (there is no NP ApplyImmediate trigger register)

The actual commissioning of a provisioned NP configuration into hardware occurs after the host initiated state transition of the NPSM from NPDeactivated to NPInit, in the NPInit state.

**Table 8-125 Network Path Provisioning per Lane (NPConfigLane<i>)**

| Lane | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| <i> | 7-4 | - | **Reserved** | RW<br>Rqd. |
| | 3-1 | NPID | **SCS<k>::NPIDLane<i>**<br>If host lane <i> feeds the Network Path of an NP Application instance, the NPIDLane<i> field stores the **Network Path ID** of that Network Path, which is defined as the number of the first host lane feeding the Network Path, decremented by one.<br><br>If lane<i> is unused (NPInUseLane<i> = 0), the value of NPIDLane<i> is ignored.<br><br>*Note: All lanes of the Network Path of an Application that spans multiple host lanes have the same NPID.*<br>*Note: For example, the NPID of a Network Path carrying the HP of host lane 1 is 0 and the NPID of a NP where host lane 5 is the lowest lane number feeding the NP is 4.* | |
| | 0 | NPInUse | **SCS<k>::NPInUseLane<i>**<br>0b: host lane <i> is either part of the Data Path of a DP Application or it is unused<br>1b: host lane <i> is part of a Host Path that feeds the Network Path identified by the NPID field in a NP Application | |

**Table 8-126 Staged Control Set 0, Network Path Configuration (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 128 | 7-4 | - | **SCS0::NPConfigLane1**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane1 | | |
| | 0 | NPInUseLane1 | | |
| 129 | 7-4 | - | **SCS0::NPConfigLane2**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane2 | | |
| | 0 | NPInUseLane2 | | |
| 130 | 7-4 | - | **SCS0::NPConfigLane3**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane3 | | |
| | 0 | NPInUseLane3 | | |
| 131 | 7-4 | - | **SCS0::NPConfigLane4**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane4 | | |
| | 0 | NPInUseLane4 | | |
| 132 | 7-4 | - | **SCS0::NPConfigLane5**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane5 | | |
| | 0 | NPInUseLane5 | | |
| 133 | 7-4 | - | **SCS0::NPConfigLane6**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane6 | | |
| | 0 | NPInUseLane6 | | |
| 134 | 7-4 | - | **SCS0::NPConfigLane7**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane7 | | |
| | 0 | NPInUseLane7 | | |
| 135 | 7-4 | - | **SCS0::NPConfigLane8**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane8 | | |
| | 0 | NPInUseLane8 | | |

**Table 8-127 Staged Control Set 1, Network Path Configuration (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 136 | 7-4 | - | **SCS1::NPConfigLane1**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane1 | | |
| | 0 | NPInUseLane1 | | |
| 137 | 7-4 | - | **SCS1::NPConfigLane2**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane2 | | |
| | 0 | NPInUseLane2 | | |
| 138 | 7-4 | - | **SCS1::NPConfigLane3**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane3 | | |
| | 0 | NPInUseLane3 | | |
| 139 | 7-4 | - | **SCS1::NPConfigLane4**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane4 | | |
| | 0 | NPInUseLane4 | | |
| 140 | 7-4 | - | **SCS1::NPConfigLane5**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane5 | | |
| | 0 | NPInUseLane5 | | |
| 141 | 7-4 | - | **SCS1::NPConfigLane6**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane6 | | |
| | 0 | NPInUseLane6 | | |
| 142 | 7-4 | - | **SCS1::NPConfigLane7**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane7 | | |
| | 0 | NPInUseLane7 | | |
| 143 | 7-4 | - | **SCS1::NPConfigLane8**<br>See Table 8-125 | RW<br>Rqd. |
| | 3-1 | NPIDLane8 | | |
| | 0 | NPInUseLane8 | | |

### 8.15.2 Network Path Control

The operational status of the provisioned Network Paths (see the NP Active Control Set described in section 8.15.4) is determined by the content of the **NPDeinit** register.

The NPDeinit register controls the initialization of the media lanes in all configured Network Paths that are associated with the 8 lanes represented in a Bank.

The module evaluates the NPDeinit register only in the fully operational Module State ModuleReady. When the Module State is ModuleReady, the Network Path associated with lanes whose NPDeinit bits are cleared will transition to the NPInit state and begin the media lane initialization process.

*Note: By default, all configured Network Paths will begin initializing when the Module State reaches ModuleReady. The host can prevent this auto-initialization behavior by setting the NPDeinit bits while the module is in the ModuleLowPwr state.*

At time of initialization (commissioning), the multiplex structure of the NP application must be provisioned in the Active Control set, in terms of HP definitions, but the HPs can be in any achievable DPSM state.

Parallel Network Paths are mutually independent. They may be initialized or deinitialized by one command or by a sequence of commands.

**Table 8-128 Network Path Initialization Control (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 160 | 7 | NPDeinitLane8 | **NPDeinitLane<i>**<br>Initialization control for the Network Path fed by host lane <i><br>0b: Initialize the Network Path associated with host lane <i><br>1b: Deinitialize the Network Path associated with host lane <i><br><br>All host lanes feeding one Network Path must have the same NPDeinitLane<i> value set<br>*Note: These bits represent static requests, not trigger events* | RW<br>Rqd. |
| | 6 | NPDeinitLane7 | | |
| | 5 | NPDeinitLane6 | | |
| | 4 | NPDeinitLane5 | | |
| | 3 | NPDeinitLane4 | | |
| | 2 | NPDeinitLane3 | | |
| | 1 | NPDeinitLane2 | | |
| | 0 | NPDeinitLane1 | | |

Supporting configurable client replacement signal data insertion in Tx direction at NP inputs from HP outputs, replacing received host signal data to be mapped into NP inputs, is optional and hence advertised.

Supporting configurable client replacement signal insertion in Rx direction at HP inputs from NP outputs, replacing received and demapped host signal data of NP outputs, is optional and hence advertised.

*Note: It is assumed that the pertinent specifications of an NP application specify well defined consequent actions for the case when no valid host signal is available to be forwarded. The default configuration is therefore to forward the received signal or the signal resulting from a consequent action, as this allows for intervention-free power up. See section 7.6.1.2 and 8.15.5.2 for more information on client replacement signals*

**Table 8-129 Network and Host Path Signal Source Selection (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 162 | 7 | HPSourceRx8 | **HPSourceRx<i>**<br>Controls the signal source feeding host lane <i> in Rx direction<br>0b: Signal received from Network Path<br>1b: Internally generated client replacement signal<br>All host lanes belonging to the same HP must have the same HPSourceRx<i> value set<br><br>Advertisement: 16h:226.0 | RW<br>Cond. |
| | 6 | HPSourceRx7 | | |
| | 5 | HPSourceRx6 | | |
| | 4 | HPSourceRx5 | | |
| | 3 | HPSourceRx4 | | |
| | 2 | HPSourceRx3 | | |
| | 1 | HPSourceRx2 | | |
| | 0 | HPSourceRx1 | | |
| 163 | 7 | NPSourceTx8 | **NPSourceTx<i>**<br>Controls the signal source feeding the Network Path input related to the HP containing host lane <i><br>0b: Signal received from Host Path<br>1b: Internally generated client replacement signal<br>All host lanes belonging to the same HP must have the same NPSourceTx<i> value set<br>Advertisement: 16h:226.1 | RW<br>Cond. |
| | 6 | NPSourceTx7 | | |
| | 5 | NPSourceTx6 | | |
| | 4 | NPSourceTx5 | | |
| | 3 | NPSourceTx4 | | |
| | 2 | NPSourceTx3 | | |
| | 1 | NPSourceTx2 | | |
| | 0 | NPSourceTx1 | | |

### 8.15.3 Network Path Commands

Triggering the execution of a command to **provision** a configuration prepared in an NP Staged Control Set into the NP Active Control Set by writing to an **ApplyNPInit** trigger register and the associated command handling protocol using the **NPConfigStatus** status register is fully analogous to DP command handling for Data Paths described in section 6.2.4.

The ApplyNPInit trigger register allows the host to trigger the execution of the Network Path provisioning or re-provisioning commands for the lanes selected via the lane bit mask in the value written to the trigger register.

*Note: Unlike the name might suggest, writing to ApplyNPInit causes a provisioning step to be executed but does not itself cause the NPInit state to be entered, as this step is governed by the NPSM.*

*Note: The ApplyNPInit register is a stateless trigger registers with write-only access type. This implies that the value read from the register is not specified. Modules may use the bits in these registers for any purpose, including to signal command execution or acceptance status, e.g. for debug purposes.*

**Command acceptance**: The module (silently) ignores a set trigger bit for lanes where execution of a previously triggered provisioning command is still in progress, as indicated in the associated NPConfigStatus field. Conversely, when the module accepts a triggered command for a lane, it immediately sets the NPConfigStatus field of that lane to ConfigInProgress.

**Command validation**: After setting ConfigInProgress, the module first validates the configuration to be provisioned.

*Note: Full multiplex structure validation at time of provisioning may be difficult if not impossible for the module because Host Path provisioning and Network Path provisioning are separate steps with undefined order.*

**Command execution**: After successful validation and after subsequent successful copy from the relevant NP Staged Control Set to the NP Active Control Set, the module sets the bits of the provisioned lanes in the **NPInitPending** register, indicating that the commissioning of the NP Active Control Set during the NPInit state is still outstanding.

**Command termination**: Finally the module updates the NPConfigStatus.

**Table 8-130 Staged Control Set 0, Apply Triggers (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 176 | 7 | ApplyNPInitLane8 | **SCS0::ApplyNPInitLane<i>** | WO |
| | 6 | ApplyNPInitLane7 | 0b: No action for host lane <i> | Rqd. |
| | 5 | ApplyNPInitLane6 | 1b: Trigger the **Provision** procedure using the NP Staged | |
| | 4 | ApplyNPInitLane5 | Control Set **0** settings for host lane <i>, with feedback | |
| | 3 | ApplyNPInitLane4 | provided in the associated NPConfigStatusLane<i> field | |
| | 2 | ApplyNPInitLane3 | | |
| | 1 | ApplyNPInitLane2 | Restriction: This byte must be written in a single-byte WRITE | |
| | 0 | ApplyNPInitLane1 | | |

**Table 8-131 Staged Control Set 1, Apply Triggers (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 177 | 7 | ApplyNPInitLane8 | **SCS1::ApplyNPInitLane<i>** | WO |
| | 6 | ApplyNPInitLane7 | 0b: No action for host lane <i> | Rqd. |
| | 5 | ApplyNPInitLane6 | 1b: Trigger the **Provision** procedure using the NP Staged | |
| | 4 | ApplyNPInitLane5 | Control Set **0** settings for host lane <i>, with feedback | |
| | 3 | ApplyNPInitLane4 | provided in the associated NPConfigStatusLane<i> field | |
| | 2 | ApplyNPInitLane3 | | |
| | 1 | ApplyNPInitLane2 | Restriction: This byte must be written in a single-byte WRITE | |
| | 0 | ApplyNPInitLane1 | | |

The NPConfigStatus register provides feedback on the command handling status (**in-progress**, **ready**) and the result status (**success**, **rejection** due to validation failure)

**Table 8-132 NP Configuration Command Status registers (Page 16h)**

| Byte | Bit | Field Name | Field Description | Type |
|---|---|---|---|---|
| 178 | 7-4 | NPConfigStatusLane2 | **NPConfigStatusLane<i>**<br>Provisioning Command Execution / Result Status for the Network Path of host lane <i>, during and after the most recent configuration command.<br>See Table 8-133 for the encoding of values.<br>*Note: There is no feedback to the host when an Apply trigger is ignored after failed readiness test (when another configuration is still in progress)* | RO Rqd. |
| | 3-0 | NPConfigStatusLane1 | | |
| 179 | 7-4 | NPConfigStatusLane4 | | RO Rqd. |
| | 3-0 | NPConfigStatusLane3 | | |
| 180 | 7-4 | NPConfigStatusLane6 | | RO Rqd. |
| | 3-0 | NPConfigStatusLane5 | | |
| 181 | 7-4 | NPConfigStatusLane8 | | RO Rqd. |
| | 3-0 | NPConfigStatusLane7 | | |

The status codes in Table 8-133 represent both the current command handling status (**in-progress**, **ready**) and the result status information (**success**, **rejection** due to validation failure), whereby the **ready** command handling status is implicitly indicated by the presence of result status information.

**Table 8-133 NP Configuration Command Execution and Result Status Codes (Page 16h)**

| Encoding | Name | Value Description |
|---|---|---|
| 0h | ConfigUndefined | No status information available (initial register value) |
| 1h | ConfigSuccess | **Positive Result Status**: The last accepted configuration command has been completed successfully |
| 2h | ConfigRejected | **Negative Result Status (2h-Bh. Dh-Fh)**:<br>Configuration rejected: unspecific validation failure |
| 3h | ConfigRejectedInvalidAppSel | Configuration rejected: invalid AppSel codes |
| 4h | ConfigRejectedInvalidNetworkPath | Configuration rejected: invalid set of lanes for AppSel |
| 5h | - | **Reserved** |
| 6h | ConfigRejectedLanesInUse | Configuration rejected: some lanes not in NPDeactivated |
| 7h | ConfigRejectedPartialNetworkPath | Configuration rejected: lanes are only subset of Network Path |
| 8h | - | **Reserved** (other validation failures) |
| 9h | - | |
| Ah | - | |
| Bh | - | |
| Ch | ConfigInProgress | **Execution Status**: A configuration command is still being processed by the module; a new configuration command is **ignored** for this lane while ConfigInProgress. |
| Dh | - | **Custom** Configuration rejected for custom reasons |
| Eh | - | |
| Fh | - | |

### 8.15.4 Network Path Status

The Active Control Set represents the currently commissioned NP configuration, except transiently during a reconfiguration when the NPInitPending status is active.

**Table 8-134 NP Active Control Set, Network Path Configuration (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 192 | 7-4 | - | **ACS::NPConfigLane1** | RO |
| | 3-1 | NPIDLane1 | See Table 8-125 | Rqd. |
| | 0 | NPInUseLane1 | | |
| 193 | 7-4 | - | **ACS::NPConfigLane2** | RO |
| | 3-1 | NPIDLane2 | See Table 8-125 | Rqd. |
| | 0 | NPInUseLane2 | | |
| 194 | 7-4 | - | **ACS::NPConfigLane3** | RO |
| | 3-1 | NPIDLane3 | See Table 8-125 | Rqd. |
| | 0 | NPInUseLane3 | | |
| 195 | 7-4 | - | **ACS::NPConfigLane4** | RO |
| | 3-1 | NPIDLane4 | See Table 8-125 | Rqd. |
| | 0 | NPInUseLane4 | | |
| 196 | 7-4 | - | **ACS::NPConfigLane5** | RW |
| | 3-1 | NPIDLane5 | See Table 8-125 | Rqd. |
| | 0 | NPInUseLane5 | | |
| 197 | 7-4 | - | **ACS::NPConfigLane6** | RW |
| | 3-1 | NPIDLane6 | See Table 8-125 | Rqd. |
| | 0 | NPInUseLane6 | | |
| 198 | 7-4 | - | **ACS::NPConfigLane7** | RW |
| | 3-1 | NPIDLane7 | See Table 8-125 | Rqd. |
| | 0 | NPInUseLane7 | | |
| 199 | 7-4 | - | **ACS::NPConfigLane8** | RW |
| | 3-1 | NPIDLane8 | See Table 8-125 | Rqd. |
| | 0 | NPInUseLane8 | | |

The current state of the NPSM associated with a host lane (if any) is indicated as follows, using the encoding defined in Table 8-136

**Table 8-135 Lane-associated Network Path States (Page 16h)**

| Byte | Bit | Field Name | Register Description (NPStateHostLane<i>) | Type |
|---|---|---|---|---|
| 200 | 7-4 | NPStateHostLane2 | Network Path State of host lane 2 (see Table 8-136) | RO |
| | 3-0 | NPStateHostLane1 | Network Path State of host lane 1 (see Table 8-136) | Rqd. |
| 201 | 7-4 | NPStateHostLane4 | Network Path State of host lane 4 (see Table 8-136) | RO |
| | 3-0 | NPStateHostLane3 | Network Path State of host lane 3 (see Table 8-136) | Rqd. |
| 202 | 7-4 | NPStateHostLane6 | Network Path State of host lane 6 (see Table 8-136) | RO |
| | 3-0 | NPStateHostLane5 | Network Path State of host lane 5 (see Table 8-136) | Rqd. |
| 203 | 7-4 | NPStateHostLane8 | Network Path State of host lane 8 (see Table 8-136) | RO |
| | 3-0 | NPStateHostLane7 | Network Path State of host lane 7 (see Table 8-136) | Rqd. |

**Table 8-136 Network Path State Encoding**

| Encoding | State |
|---|---|
| 0h | Reserved |
| 1h | NPDeactivated (or unused lane) |
| 2h | NPInit |
| 3h | NPDeinit |
| 4h | NPActivated |
| 5h | NPTxTurnOn |
| 6h | NPTxTurnOff |
| 7h | NPInitialized |
| 8h-Fh | Reserved |

**Table 8-137 Network Path Conditions (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 204 | 7 | NPInitPendingLane8 | **NPInitPendingLane<i>**<br>0b: NPInit not pending<br>1b: Commissioning the NP Active Control Set during NPInit has not yet been executed after a successful ApplyNPInit, hence the NP Active Control Set content may still deviate from the actual hardware configuration.<br><br>*Note: The setting SteppedConfigOnly is irrelevant for the NPSM.* | RO<br>Rqd. |
| | 6 | NPInitPendingLane7 | | |
| | 5 | NPInitPendingLane6 | | |
| | 4 | NPInitPendingLane5 | | |
| | 3 | NPInitPendingLane4 | | |
| | 2 | NPInitPendingLane3 | | |
| | 1 | NPInitPendingLane2 | | |
| | 0 | NPInitPendingLane1 | | |

### 8.15.5 Network Path Related Advertisements (Capabilities and Restrictions)

#### 8.15.5.1 Maximum Durations Advertisement

The maximum duration of transient NPSM states is advertised by the module as follows

**Table 8-138 NPSM Durations Advertising (Page 16h)**

| Byte | Bit | Field Name | Field Description | Type |
|---|---|---|---|---|
| 224 | 7-4 | MaxDurationNPDeinit | Maximum duration of the NPDeinit state (encoded as per Table 8-43) | RO Rqd. |
| | 3-0 | MaxDurationNPInit | Maximum duration of the NPInit state (encoded as per Table 8-43) | |
| 225 | 7-4 | MaxDurationNPTxTurnOff | Encoded maximum duration of the NPTxTurnOff state (see Table 8-43) | RO Rqd. |
| | 3-0 | MaxDurationNPTxTurnOn | Encoded maximum duration of the NPTxTurnOn state (see Table 8-43) | |

#### 8.15.5.2 Miscellaneous Options

A module advertises if the host can configure a host replacement signal to be forwarded in downstream direction, instead of the (possibly but not necessarily failed) host signal received from an upstream source.

*Note: Replacement signal insertion is intended to be used when no host signal is connected to a multiplex connection point of the NP (Tx direction) or when no host signal is expected from a multiplex connection point of the NP (Rx direction), i.e. always when a host signal is known to be intentionally missing or "unequipped".*

*Note: It is assumed that the pertinent transmission specifications of an NP application provide transmit and forwarding specifications in case of an upstream host signal failure, such as automatic AIS or LF insertion. Reactive modifications of the data stream, automatically performed by the module, prior to forwarding the data stream are often called "consequent actions".*

*Note: When configurable replacement signals are not supported, a deliberately unconnected host signal would be treated by the module as a host signal failure, while supporting configurable replacement signals may allow to distinguish an intentionally unconnected signal from a signal failure leading to consequent actions.*

**Table 8-139 Miscellaneous Options (Page 16h)**

| Byte | Bit | Field Name | Field Description | Type |
|---|---|---|---|---|
| 226 | 7-2 | - | **Reserved** | RO Rqd. |
| | 1 | ReplaceHPSignalTxSupported | 0b: not supported<br>1b: Tx replacement signals for NP inputs are supported | |
| | 0 | ReplaceHPSignalRxSupported | 0b: not supported<br>1b: Rx replacement signals for NP outputs are supported | |

#### 8.15.5.3 Application Advertisement Extensions

As an extension to the basic application advertisement described in section 8.2.11 a module supporting NP Applications advertises for each Application Descriptor (identified by its AppSel code) if the advertised application is a DP Application or a NP Application.

This application advertisement extension is necessary for robust distinction of system interface applications (DP Applications) and uniplex applications (NP Applications).

*Note: An Application Descriptor for a genuine homogeneous multiplex application can also be recognized from the data rates associated with the MediaInterfaceID and the HostInterfaceID: The number of homogeneously multiplexed Host Paths is the quotient of the larger information rate of the media interface (MediaInterfaceID) and the smaller information rate of the host interface (HostInterfaceID) in the Application Descriptor.*

*Note: The extension to the Application Descriptor is required (for modules supporting NP Applications) because the distinction of a uniplex NP Application and a DP Application cannot always be derived from the meaning of the advertised MediaInterfaceID, and hence the distinction must be indicated explicitly.*

**Table 8-140 NP Extended Application Advertisement (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 248 | 7 | ExtAppDescriptor15 | **ExtAppDescriptor<i>**<br>The Application Descriptor identified by AppSel<i>:<br>0b: describes a supported DP Application<br>1b: (partially) describes a supported NP Application<br><br>*Note: The value is irrelevant when the Application Descriptor identified by AppSel<i> is unused.* | RW<br>Rqd. |
| | 6 | ExtAppDescriptor14 | | |
| | 5 | ExtAppDescriptor13 | | |
| | 4 | ExtAppDescriptor12 | | |
| | 3 | ExtAppDescriptor11 | | |
| | 2 | ExtAppDescriptor10 | | |
| | 1 | ExtAppDescriptor9 | | |
| | 0 | ExtAppDescriptor8 | | |
| 249 | 7 | ExtAppDescriptor7 | | |
| | 6 | ExtAppDescriptor6 | | |
| | 5 | ExtAppDescriptor5 | | |
| | 4 | ExtAppDescriptor4 | | |
| | 3 | ExtAppDescriptor3 | | |
| | 2 | ExtAppDescriptor2 | | |
| | 1 | ExtAppDescriptor1 | | |
| | 0 | - | | |

### 8.15.5.4 Multiplex and Uniplex Application Advertisement

NP Applications are advertised by one or more Application Descriptors (see Table 8-20, Table 8-52, and Table 8-53), each of which indicates a HP type (HostInterfaceID) and the NP type (MediaInterfaceID), together with the Application Descriptor Extensions (see Table 8-140) distinguishing NP Applications and DP Applications.

**Uniplex NP and Homogeneous Multiplex Application Advertisement**

A uniplex or homogeneous multiplex NP Application is advertised in one Application Descriptor with AppSel=<i> where the associated ExtAppDescriptor<i> bit is set.

*Note: One characteristic resulting from a single Application Descriptor being used is that the host lane data rates of all tributaries are identical, simply because they all use the same HostInterfaceID.*

**Mixed Multiplex NP Application Advertisement**

A mixed (heterogenous) multiplex application is advertised via a **set** of mutually **consistent** Application Descriptors with all the associated ExtAppDescriptor<i> bits set, together with the additional Mixed Multiplex Descriptor registers (see Table 8-142 and Table 8-143 and section 8.15.5.5).

Application Descriptors are mutually **consistent** when they all advertise the same MediaInterfaceID with at least one common option in the MediaLaneAssignmentOptions, and when there is at least one combination of HostLaneAssignmentOptions in the Application Descriptors that assigns different host lanes to each of them.

A non-trivial **uniform lane data rate restriction** for a **mixed multiplex** application requires that the host lane data rates of all tributaries must be identical. See also subsection 7.6.1.5.

*Note: In other words, all tributaries of a multiplex application are multi-lane signals with possibly different number of lanes but with uniform lane data rate. As specified below in section 8.15.5.5, this lane data uniformity requirement extends even across parallel multiplex applications.*

*Note: The rationale for this apparently undesired restriction is to keep accurate advertising both of capabilities and of restrictions at a reasonable level of advertisement complexity.*

### 8.15.5.5 Constraints and Advertisements for Parallel NP Applications

The **uniform lane data rate restriction** applies also across **parallel** multiplexing applications (of any kind).

*Note: The rationale for this apparently undesired restriction is to keep accurate advertising both of capabilities and of restrictions, as defined below, at a reasonable level of advertisement complexity.*

The ordered list of Host Path lane groups feeding one single Network Path, in lane number order, is called the **multiplex structure** of the Network Path.

The ordered list of all Host Path lane groups (feeding any of possibly several parallel Network Paths) is called the **global multiplex structure**.

The **multiplexing granularity** of a multiplex structure is defined by a HostInterfaceID value that indicates both the common host lane data rate and the multiplexing rate granularity of a mixed multiplex structure.

A particular global multiplex structure is supported when

- The data rate of each Host Path is a **power-of-two multiple** of one selected multiplexing granularity
- The information rate of each Network Path is the **sum of the** information **rates** of all its Host Paths
- The **lane grouping** of the host lanes into Host Paths is advertised as **supported** by the module

Table 8-141 lists all possible lane groupings (global multiplex structures) together with their global multiplex structure encodings (Multiplex Structure IDs) for mixed Host Path widths in a multiplex application.

Table 8-141 also shows the Host Path DPIDs (HP DPIDs) for the Host Paths of a particular global multiplex structure, independent of any NP they belong to.

When parallel Network Paths are supported, the first NP is associated with one or more HPs using a first group of host lanes, and the $n^{th}$ NP is associated with one or more HPs using the $n^{th}$ group of host lanes.

*Note. This table has the same hand-crafted structure as Table 8-34, which is not scalable to fewer or more lanes. It is reused here because a look-up table is expected to be used in implementations. The ID numbering (0-25) is different (off by one) because here the multiplex structure ID selects a bit position in a bit mask.*

**Table 8-141 Multiplex Lane Grouping Advertisement**

| Multiplex Structure | | | HP DPID per Host Lane # | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **ID** | # of HPs | HP Widths | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 0 | 8 | 1, 1, 1, 1, 1, 1, 1, 1 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 8 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 2 | 4, 4 | 1 | 1 | 1 | 1 | 5 | 5 | 5 | 5 |
| 3 | 4 | 2, 2, 2, 2 | 1 | 1 | 3 | 3 | 5 | 5 | 7 | 7 |
| 4 | 3 | 4, 2, 2 | 1 | 1 | 1 | 1 | 5 | 5 | 7 | 7 |
| 5 | 4 | 4, 2, 1, 1 | 1 | 1 | 1 | 1 | 5 | 5 | 7 | 8 |
| 6 | 4 | 4, 1, 1, 2 | 1 | 1 | 1 | 1 | 5 | 6 | 7 | 7 |
| 7 | 5 | 4, 1, 1, 1, 1 | 1 | 1 | 1 | 1 | 5 | 6 | 7 | 8 |
| 8 | 3 | 2, 2, 4 | 1 | 1 | 3 | 3 | 5 | 5 | 5 | 5 |
| 9 | 4 | 2, 1, 1, 4 | 1 | 1 | 3 | 4 | 5 | 5 | 5 | 5 |
| 10 | 4 | 1, 1, 2, 4 | 1 | 2 | 3 | 3 | 5 | 5 | 5 | 5 |
| 11 | 4 | 1, 1, 1, 1, | 1 | 2 | 3 | 4 | 5 | 5 | 5 | 5 |
| 12 | 5 | 2, 2, 2, 1, 1 | 1 | 1 | 3 | 3 | 5 | 5 | 7 | 8 |
| 13 | 5 | 2, 2, 1, 1, 2 | 1 | 1 | 3 | 3 | 5 | 6 | 7 | 7 |
| 14 | 5 | 2, 1, 1, 2, 2 | 1 | 1 | 3 | | 5 | 5 | 7 | 7 |
| 15 | 5 | 1, 1, 2, 2, 2 | 1 | 2 | 3 | 3 | 5 | 5 | 7 | 7 |
| 16 | 6 | 2, 2, 1, 1, 1, 1 | 1 | 1 | 3 | 3 | 5 | 6 | 7 | 8 |
| 17 | 6 | 2, 1, 1, 2, 1, 1 | 1 | 1 | 3 | 4 | 5 | 5 | 7 | 8 |
| 18 | 6 | 2, 1, 1, 1, 1, 2 | 1 | 1 | 3 | 4 | 5 | 6 | 7 | 7 |
| 19 | 6 | 1, 1, 2, 2, 1, 1 | 1 | 2 | 3 | 3 | 5 | 5 | 7 | 8 |
| 20 | 6 | 1, 1, 2, 1, 1, 2 | 1 | 2 | 3 | 3 | 5 | 6 | 7 | 7 |
| 21 | 6 | 1, 1, 1, 1, 2, 2 | 1 | 2 | 3 | 4 | 5 | 5 | 7 | 7 |
| 22 | 6 | 2, 1, 1, 1, 1, 1, 1 | 1 | 1 | 3 | 4 | 5 | 6 | 7 | 8 |
| 23 | 7 | 1, 1, 2, 1, 1, 1, 1 | 1 | 2 | 3 | 3 | 5 | 6 | 7 | 8 |
| 24 | 7 | 1, 1, 1, 1, 2, 1, 1 | 1 | 2 | 3 | 4 | 5 | 5 | 7 | 8 |
| 25 | 7 | 1, 1, 1, 1, 1, 1, 2 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 7 |

Based on this coded enumeration of global multiplex structures, the module advertises the supported global multiplex options by a list of up to four **Multiplex Descriptors** as follows:

Each Multiplex Descriptor advertises a multiplex granularity **MuxGranularity** (bandwidth) and a set of supported global multiplex structures **MuxStructsSupported** for that granularity.

For reasons of access efficiency, the two parts of a Multiplex Descriptor (granularity and list of global multiplex structures) are stored in two separate register arrays, as follows:

**Table 8-142 Multiplex Granularities Advertisement (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 228 | 7-0 | MuxGranularity1 | **U8 MuxGranularity<i>**<br>0: Not supported (end of granularity list: after a zero value, all following MuxGranularity<i> fields are zero as well)<br>>0: HostInterfaceID indicating the lane data rate and the multiplex rate granularity: all multiplexed signals have the same lane data rate and a data rate that is a power of two multiple of the multiplex rate granularity | RO<br>Rqd. |
| 229 | 7-0 | MuxGranularity2 | | |
| 230 | 7-0 | MuxGranularity3 | | |
| 231 | 7-0 | MuxGranularity4 | | |

**Table 8-143 Global Multiplex Structures Advertisement (Page 16h)**

| Byte | Bits | Field Name | Register Description | Type |
|---|---|---|---|---|
| 232-235 | 31-0 | MuxStructsSupported1 | U32 **MuxStructsSupported<i>** contains a bit mask where each bit <j> set indicates support for the multiplex structure ID <j> as defined in Table 8-141 | RO<br>Rqd. |
| 236-239 | 31-0 | MuxStructsSupported2 | | |
| 240-243 | 31-0 | MuxStructsSupported3 | | |
| 244-247 | 31-0 | MuxStructsSupported4 | | |