New CNU Notification

www.huawei.com

Author/ Email: Duane Remein

Version: V1.0(20YYMMDD)

HUAWEI TECHNOLOGIES CO., LTD.

HUAWEI

Contents

- Problem statement
- Two Possible solutions
- Conclusions

Problem Statement

- **How does the EPoC PHY interact with the Upper Layers to assign CNU_IDs and notify the Upper Layers a new CNU has been registered on the network**
  - Who assigns the CNU_ID; PHY or Upper Layers?
  - How many CNUs can be registered without notifying the Upper Layers? One at a time or do we allow parallel processing?
  - What information should be passed to the Upper Layers when a CNU is found?

- **There is an impact to how MDIO registers are defined depending on how new CNU notification is specified.**

Who Assigns the CNU_ID?

PHY Assigns CNU_ID

- **PHY/Upper layers more independent**
- **CNU_ID probably not equal to LLID as they would be assigned independently**

Upper Layers Assign CNU_ID

- **Requires more coordination between layers**
- **CNU_ID can easily equal primary LLID (i.e., 1st assigned by MAC CTRL)**
  - May simplify software slightly

How many CNUs can be added before notification to upper Layers?

- **The “longest pole” item to download to a CNU for registration is the Equalizer settings.**
  - 8k registers
  - Transport takes several 100’s of ms
  - Assume can achieve 4/second
- **If PHY could register 8 CNUs at a time then the upper layers could limit polling to once every couple of seconds**
  - This establishes an upper bound without performance degradation
  - Could be lower but may have slight impact when linking numerous CNUs (network “cold start”)

What information should be passed to Upper Layers about a new CNU?

- **A new CNU_ID has been linked – only must have**
- **Value of assigned CNU_ID**
- **Range information would be very useful**
  - Helpful during MAC Discovery to limit MAC Discovery Window size
  - PHY Ranging should be a more accurate determination of RTT than MAC Discovery can resolve due to TQ jitter
    - **Would need to know delay through PHY**
- **MAC address might be useful**
  - Could be used to target MAC Discovery to individual CNUs

- **Additional information on CNU capabilities etc. could be gathered by the MAC after completion of MAC Discovery and an eOAM link has been established**

Possible solutions – MDIO Register structure

PHY assigned CNU_ID

- **Table of 8(?) register sets including CNU_ID + info**
  - New flag (1b) – set by PHY cleared by Upper Layers
    - **Set – table entry valid for new CNU registered by PHY**
    - **Clear – Table entry is available for use by PHY, cleared by Upper Layers**
  - CNU_ID assigned (15b)
  - Range (TBD)
  - MAC Add (48b)(64b?)

- **Assuming 16b Range is OK then this will use107 mdio registers per CNU or 56 registers**

Upper layer assigned CNU_ID

- **Table 1 of 8(?) assignable CNU_IDs**
  - Assigned flag (1b)) – set by PHY cleared by Upper Layers
    - **Set – table entry used for new CNU registered by PHY**
    - **Clear - Table entry is available for use by PHY, cleared by Upper Layers**
  - Allowed CNU_ID (15b)
- **Table 2 of 8(?) register sets**
  - Range (TBD)
  - MAC Add (48b)(64b?)

- **Assuming 16b Range is OK then this will use 10 mdio registers per CNU or 56 registers**

Example 1 PHY assigned CNU_ID

PHY Process

1. **Finds new CNU**
2. **Creates CNU_ID & sends to new CNU**
3. **Determines Range, MAC address, etc**
4. **Writes 1 table entry with;**
   - CNU_ID assigned
   - Range of new CNU
   - MAC Address of new CNU
5. **Changes "New" flag from False to True**

Upper Layer Process

1. **poll New flag**
2. **If New flag = True; Read**
   - CNU_ID
   - Range
   - MAC Address
3. **Begins MAC Discovery for new CNU**
4. **Clears Register set**
   - Changing New flag to False notifies the PHY that the register set is available for another new CNU

Example 2 Upper Layer assigned CNU_ID

PHY Process

1. **Finds new CNU**
2. **Reads CNU_ID from Table 1 & sends to new CNU**
   - Any entry with “Assigned” flag = false
3. **Determines Range, MAC address, etc**
4. **Writes corresponding Table 2 entry with;**
   - Range of new CNU
   - MAC Address of new CNU
5. **Sets “Assigned” flag**

Upper Layer Process

1. **Loads CNU_ID table with allowable values & enables PHY Discovery**
2. **Polls “Assigned” flag**
3. **If “Assigned” flag = T; Read**
   - Range
   - MAC Address
4. **Clears entry in Table 2**
5. **Set next allowable CNU_ID value in Table 1 & clears “Assigned” flag**

Conclusion

- **There is no significant complexity difference between the two methods outlined**
- **Allowing the Upper Layer to assign the CNU_ID may have some advantages**

- **Not addressed: CNU de-registering at PHY level**
  - How does Upper Layer interact with PHY on MAC de-registration?
  - How does PHY interact with Upper layers if it de-links a CNU?

Thank you

www.huawei.com

Straw Poll

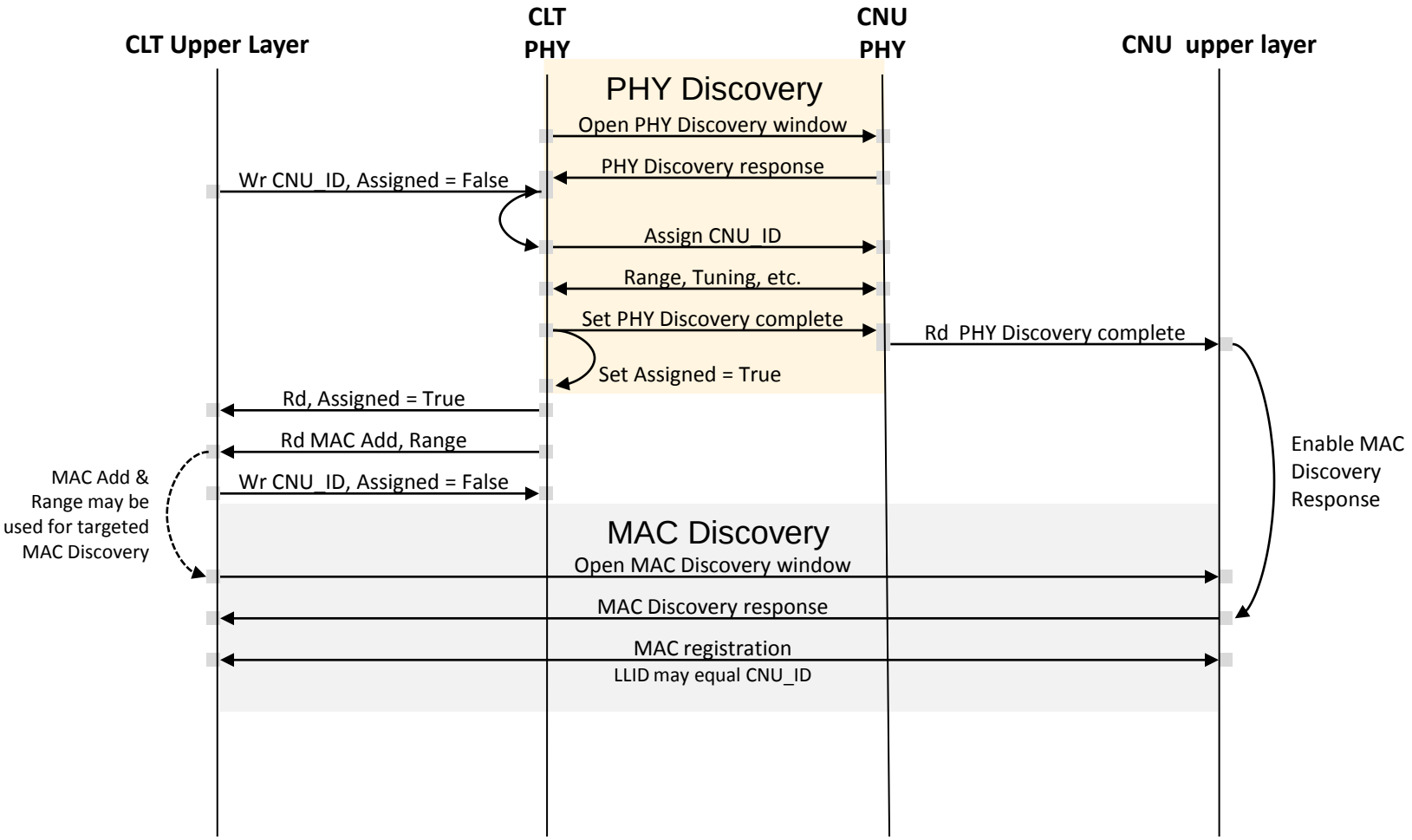
- **I agree with creating registers and process description to allow the Upper Layers to pre-assign CNU_IDs for up to 8 CNUs at a time as described in remein_01_1406.pdf**
  - Yes _____
  - No _____
  - Don’t care _____

- **Backup**

www.huawei.com

HUAWEI

Process Diagram



PHY SD for Upper Layer assigned CNU_ID

Variables

RcvDisRsp
- TYPE: Boolian
- When True indicates a valid PHY Discovery Response is being processed

TblIdx
- TYPE: Integer between 0 & 7
- Index into Table 1 & 2 (see 45.x.x.x & 45.x.x.y)

NewCNU_ID
- TYPE: 15b binary
- A value indication the identifier for the new CNU_ID

NCNU_Reg
- TYPE: Boolian
- When True indicates that the new CNU has been fully registered with the PHY

Range
- TYPE: TBD
- This variable indicates the distance (in TBD) between the CLT and newly discovered CNU

NewMAC
- TYPE: 40b binary
- This variable reflects the MAC address of the new CNU as received in the Discovery Response message.

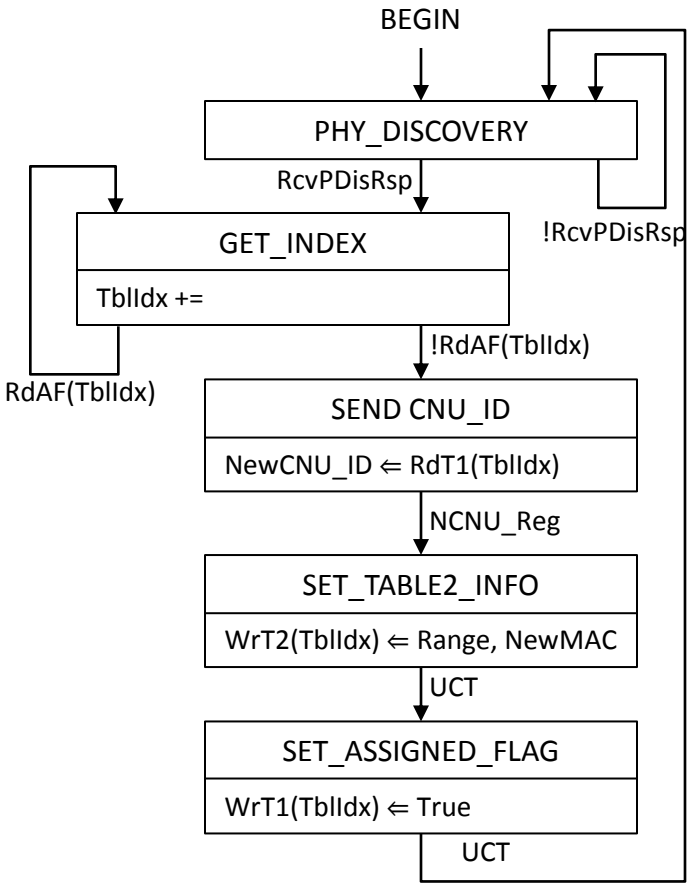
Functions

RdAF(x)
- This function performs a read of row x of the Assigned flag from Table 1 bit 15 (see 45.x.x.x)

RdT1(x)
- This function performs a read at row x of a valid CNU_ID from Table 1 bits 14:0 (see 45.x.x.x)

WrT2(x)
- This function performs a write at row x of Table 2 (see 45.x.x.y)

WrT1(x)
- This function performs a write to the Assigned flag at row x of Table 1 bit 15 (see 45.x.x.x)



Clause 45 additions

- **Table 1**

10GPASS-XR New CNU control registers 1-8 bit definitions			
1.19x1.16	CNU_ID Assigned flag 1	The Allowed CNU_ID 1 has been assigned to a new CNU	R/W
1.19x1.14:0	Allowed CNU_ID 1	A new CNU may be assigned this value for CNU_ID if the associated CNU_ID Assigned flag is false	R/W
1.19x2..19x7	…	…	R/W
1.19x8.16	CNU_ID Assigned flag 8	The Allowed CNU_ID 8 has been assigned to a new CNU	R/W
1.19x8.14:0	Allowed CNU_ID 8	A new CNU may be assigned this value for CNU_ID if the associated CNU_ID Assigned flag is false	R/W

- **Table 2**

10GPASS-XR New CNU Info registers 1-48 bit definitions			
1.19a0.15	Reserved	Ignore on read	RO
1.19a0.14:0	NewCNU1Range	The range of NewCNU1 as determined by the PHY Discovery process	R/W
1.19a1..19a6	NewCNU1MAC	The MAC address of the NewCNU1 as determined by the PHY Discovery process	R/W
1.19b0.15	Reserved	Ignore on read	RO
1.19b0.14:0	NewCNU2Range	The range of NewCNU1 as determined by the PHY Discovery process	R/W
1.19b1..19b6	NewCNU2MAC	The MAC address of the NewCNU1 as determined by the PHY Discovery process	R/W
1.19c0..19g6	…	…	R/W
1.19h0.15	Reserved	Ignore on read	RO
1.19h0.14:0	NewCNU8Range	The range of NewCNU1 as determined by the PHY Discovery process	R/W
1.19h1..19h6	NewCNU8MAC	The MAC address of the NewCNU1 as determined by the PHY Discovery process	R/W

Place these registers at 45.2.1.116 between current registers 19gg 10GPASS-XR PHY Discovery control register and 10GPASS-XR PHY frame counter bit definitions. Renumber subsequent registers as required.