

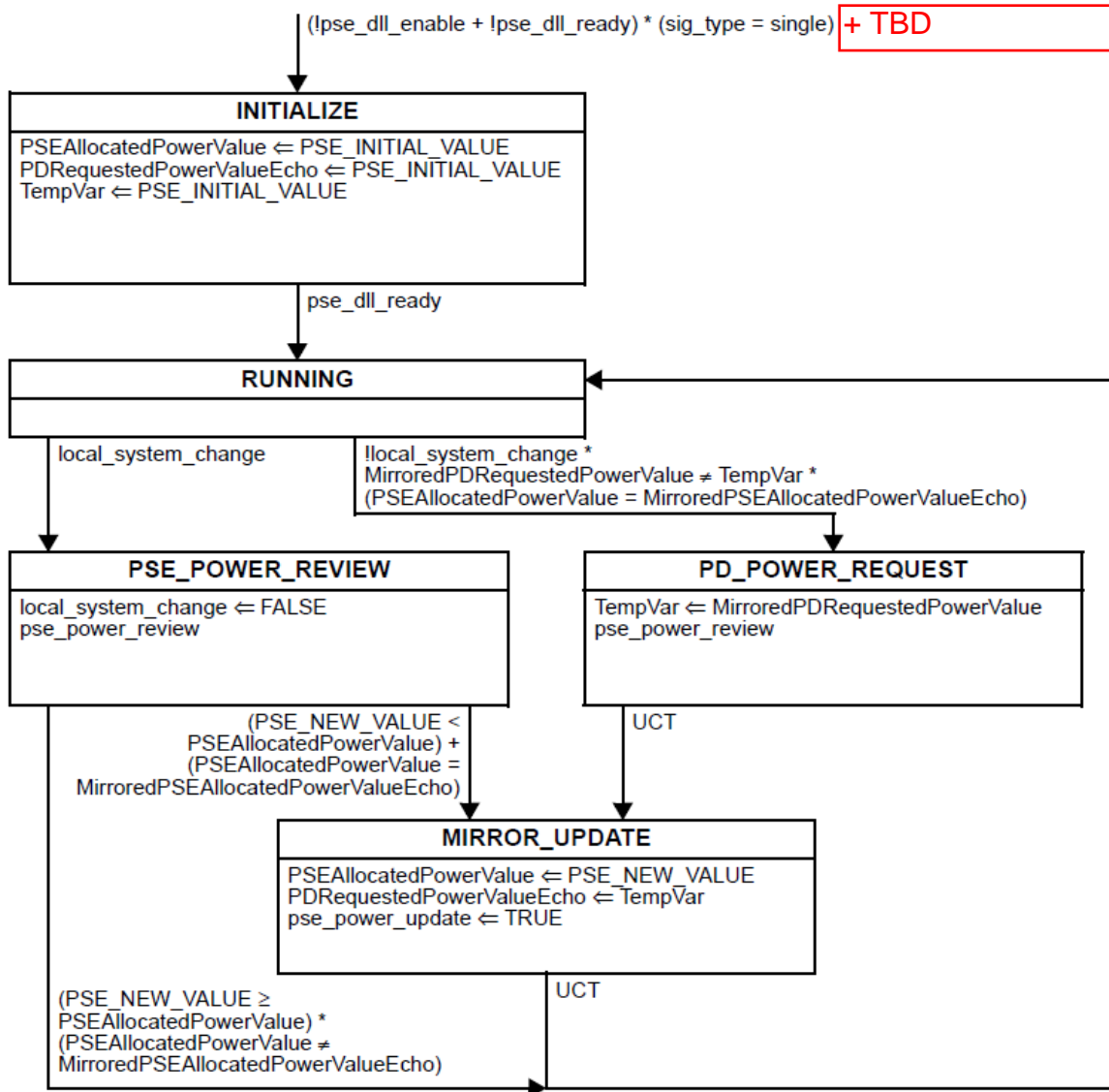
Yair: Answer to (5):

At the INITIALIZE or the IDLE state (depend on the SM), we will have input $(!pse_dll_enable_alt(X) + !pse_dll_ready_alt(X)) * (sig_type = dual) + TBD$, that when TBD is TRUE, it will initiate a function that will fill in the TLVs correctly, and will switch to the other state machine so the status of the internal variables of the previous state machine will not be important since the process (power negotiation) will start again based on the new conditions and last informations from the previous state machine

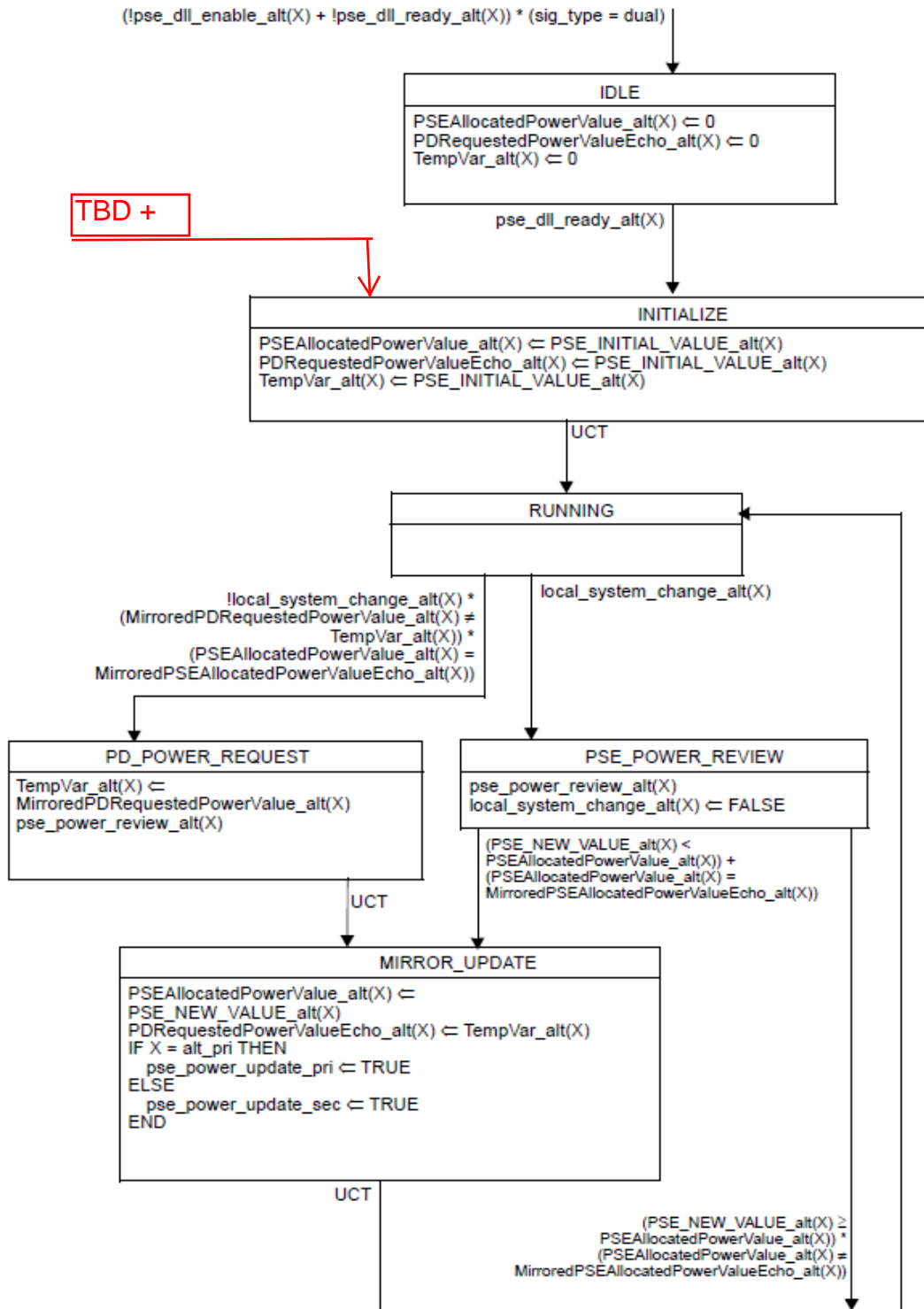
Scenario:

- 1) A Dual signature PD is powered over 2 pairs ✓
- 2) A PD_POWER_REQUEST is in progress ✓
 - a. TempVar and MirroredPDRequestedPowerValue are in use ✓
- 3) A transition to 4 pair power occurs ✓
- 4) Suddenly the alt_x versions are in force ✓
- 5) How does the state to state mapping occur when jumping from state machine to state machine? ✓

See following state machines for reference.



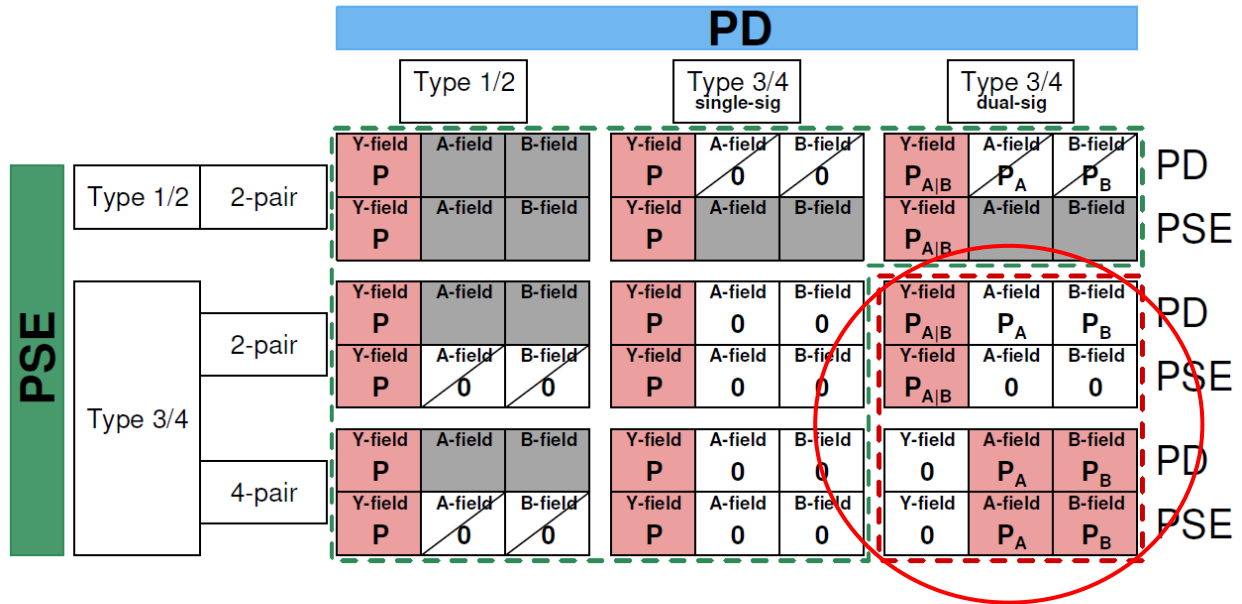
PSE Power Control SM for SS PDs



PSE Power Control SM for DS PDs

Instead is there a way to provision the SMs for DS PDs so state machine don't need to be laboriously glued together?

The root problem seems to lie in one particular set of cells in Lennart's proposal:



Can all states which assign power or mirrored power get the following additional logic:

```

If (X == A)
  If (pairsx == both)
    Y = 0
  else
    Y = A
Else
  If (pairsx == both)
    Y = 0
  else
    Y = B
  
```

Yair: How this logic works. Where is the END or Ends of this function?