Frequently Asked Questions: MIPI Alliance Standard for D-PHY v0.65

Version 0.65 – 26 July 2006

Questions pertaining to this document, or the terms or conditions of its provision, should be addressed to:

MIPI Alliance, Inc.
c/o IEEE-ISTO
445 Hoes Lane
Piscataway, NJ 08854
Attn: Board Secretary

Frequently Asked Questions:
MIPI Alliance Standard for D-PHY v0.65

List of FAQ items

1. D-PHY Specification version 0.65 versus 1.0
    a. Why is the first release of D-PHY called “v0.65” rather than “v1.0”?
    b. Is the number 0.65 significant?
    c. What changes are expected after v0.65?
2. Bit Error Rate (BER)
    a. Why is there no specification of BER in the D-PHY specification document?
    b. Many other PHY specifications include BER values, so why doesn’t D-PHY?
    c. How is a BER “target” different from a specified value?
    d. Does the BER target apply to the CLOCK lane as well as the DATA lane?
    e. Will a single D-PHY lane have only single bit errors or will multi-bit bursts occur?
    f. How is the target BER of 10e-12 calculated?
    g. Is a burst of external interfering noise that causes a single bit error on one data lane likely to affect other data lanes of a multi-lane D-PHY (automatically causing a multibit error in a transmission)?
    h. Would the CLOCK lane also have a BER of 10e-12?
    i. Could an error on the CLOCK lane corrupt more than one bit in the reception of the associated data packet?
3. Why are there two low power modes (LP and ULP)?
4. Why does D-PHY not specify any clock system?
5. High Speed sync sequence
    a. How robust is the high speed SoT synchronization sequence?
    b. How many bits should the receiver use to detent sync?
    c. Can I build a receiver that is tolerant of two or more bit errors in the sync sequence?
6. Will DZolp(01,10) be removed from the D-Phy specification after v0.65?
7. Normative requirements in D-PHY for End of Transmission [EOT] processing
    a. EOT processing is normative in the D-PHY specification (section 5.4.3) when this operation can be done outside the D-PHY for some protocols? EOT processing in the D-PHY requires a data buffer, which is not ideal for an I/O function.
8. Interoperability
    a. How can I calculate the maximum operating data rate for a pair of components from their documented values of Maximum Data Rate and UIinst(min)?

Q. Why is the first release of D-PHY called "v0.65" rather than "v1.0"?

A. D-PHY v0.65 represents a specification effort that is nearly complete, but not yet tested in silicon and therefore "subject to change." The MIPI Phy Working Group intends the first "one point zero" release of the D-PHY Specification to offer all MIPI members the confidence of an interoperable design proven in silicon by several semiconductor companies. Therefore, the WG prefers not to use the major milestone of version 1.0 if a member company has not yet put that Spec release through fabrication. The MIPI Board supports this approach.

Even so, the release of D-PHY v0.65 is a major milestone that offers the industry a lot of value. There is significant material in the Spec that allows a MIPI member company to begin work on a physical layer. It deserves to be released in a licensable form to allow companies to begin if they desire.

Neither the Board nor the Phy WG wants to limit exposure of the D-PHY Specification while work is ongoing toward a 1.0, which is expected during 2007. MIPI expects that some Board, Contributor, and Adopter companies will all begin work on D-PHY implementations now, and it is hoped that the experiences can be used to finalize a 1.0.

Q. Is the version number 0.65 significant? Is the Spec "65% done?"

A. The number has no relation to the state of completeness. The MIPI board considers the completeness of D-PHY v0.65 well above 65%, ready enough for member companies to begin preliminary implementations.

We are all familiar with other standards organizations that use specific version numbers to indicate specific milestones, but in general Phy WG has chosen not to proceed with this approach. The WG simply incremented the version number by 0.01 with each new draft of the spec, whether the changes were significant or minor.

Q. What changes are expected after v0.65?

A. The Phy Working Group does not expect to change any basic concepts or the functions performed by the PHY as it works toward v1.0. Nonetheless, discussion and work continues, and the WG has identified some expected changes to some details in the Spec.

Where practical, the WG has documented expected areas of change in a "D-PHY v0.65 Release Notes" companion document available to all MIPI members in the same Web directory as the D-PHY Specification. The first release of this document occurred on 1 December 2005, and the document will be updated as necessary, though infrequently, on the path to v1.0. In some cases, the document suggests the direction of a change to D-PHY v0.65 that is expected for v1.0.

Please keep in mind that until a new Specification is officially approved, the Release Notes should be considered as informative, not an authoritative list of "approved" Spec

changes. The topics in the Release Notes are actively being discussed, and any suggested modifications in the Release Notes are subject to change.

Q. Why is there no specification of Bit Error Rate (BER) in the D-PHY specification document?

A. The measurable BER of a D-PHY depends on the complete implementation. Since the Specification cannot control all implementation details without placing unusual burdens and cost on all implementers, D-PHY makes it possible for implementers to achieve a high level of robustness on the link if care is taken in all areas of the implementation.

The required level of robustness was set by cooperative discussion between MIPI PHY and protocol working groups (such as Camera and Display WG's) who discussed basic requirements for a physical layer. The PHY WG was given the target of specifying a link capable of a BER equal to or better than 10e-12.

Phy WG achieved this requirement by setting specifications on signal levels (especially the signal level difference between a logical "0" and "1") and timing margins as seen at the receiving end of an interconnect network. These specified values allow the design of a robust implementation across variations in semiconductor process, temperature and supply voltage that can easily exceed this BER target in the absence of externally coupled noise.

External noise may be coupled into the link via the power, ground or signal interconnects. Its net effect is to degrade the signal and timing margins, creating a higher probability of the receiver mis-sampling the transmitted data. D-PHY v0.65 specifies a set of interconnect values sufficient to insure transmission of signaling frequency components and attenuation of non-signaling, i.e. interfering components. However, external noise sources occur in the signaling frequency bands [depending on data rate] and will therefore introduce signal degradation and potentially errors. It is the system integrator, i.e. application designer, who must balance the cost of the interconnect, power supply integrity, ground connections, and signal shielding while considering the data-sheet noise rejection specifications of the transmitter and receiver ICs. It is not within the scope of a PHY specification to make this tradeoff. It is the consensus of the PHY WG that robust implementations of D-PHY can be designed at reasonable cost with error rates exceeding the target.

Q. Many other PHY specifications include BER values, why not D-PHY?

A. Those specifications address topics that are out of scope either for MIPI specifications, or for D-PHY in particular. For example, the MIPI Alliance does not engage in compliance testing. Some of those other PHY Specs include descriptions of necessary compliance tests with BER limits such as "when transmitting PRBS 2e31 pattern over 40cm of 50 Ohm FR4 stripline, a BER of less than X must be achieved." Compliance specifications of this form are not part of the D-PHY Specification as currently drafted, as per MIPI Board instruction.

Q. Does the BER target apply to the CLOCK lane as well as the DATA lane?

A. The short answer is that BER is not a valid concept for the CLOCK lane. Clock lane errors manifest themselves in a different form: they are a loss of coherence with the data lane. A correct implementation of a source synchronous link maintains coherence between the CLOCK and DATA lanes by launching the data with same clock edge put on the CLOCK lane so that any instantaneous jitter in the clock is transferred to the data.

In this way, when the data and clock arrive at the receiver, the jitter deviation in the CLOCK is coherent with the jitter deviation of the DATA and no timing margin is lost. This requires very low skew between the clock and data lanes due to interconnect and silicon delays. [Note that even source synchronous systems with DLL skew will adjust the jitter transmitted through the DLL due to the high jitter transfer characteristics of DLLs].

The external noise coupling mechanisms for the CLOCK lane are the same as for the DATA lane. The effect of the coupled noise as seen at the receiver is to move the edges of the clock relative to the data. A missing clock pulse is conceptually an edge that has been moved by more than one UI. A clock error occurs when a clock edge moves such that it falls outside of the allowed fractional UI period jitter specification. Many factors will determine if this will cause a data error. There is therefore a non-zero probability that a clock error will not be observable to the user.

The total error rate of the PHY is therefore:

P(data lane missampling data)	=	P(error due to data lane noise coupling) + P(clock edge moved out of spec) * P(moved clock edge causes sampling error in data lane).

Q. Will a single D-PHY lane have only single bit errors or will multi-bit bursts occur?

A. Real world noise sources produce interfering signals over finite time intervals. If this time interval is short relative to a bit time on the link, then it is likely to cause only a single bit error per lane. If it is long relative to a bit time, then multibit error bursts are likely to occur.

Therefore the error characteristic of a lane operating under identical noise conditions is dependent on the data rate. The higher the data rate, the more likely errors will have a burst characteristic. It is for this reason that high speed links use CRC type error correction consisting of polynomial, Reed-Solomon or LPC codes rather than simple linear codes like Hamming codes.

Q. How is the target BER of 10e-12 calculated?

A. Just as the acronym indicates, this is a *BIT* error rate. The total number of bits received in error is divided by the total number of bits transmitted. For this number to be useful, an implicit Gaussian error distribution is required. As discussed above, given

the noise environment in a handset, this is probably not a valid assumption. The BER target number should therefore be interpreted as a number, which would be measured by e.g. a PARBERT tester over a long period of time. It is therefore an average of all errors, single bit or burst.

Q. In a multi-lane D-PHY, Is a burst of external interfering noise that causes a single bit error on one data lane likely to affect other data lanes (thereby automatically causing a multibit error in a transmission)?

A. The short answer is probably yes. The long answer is that it depends on how the noise is coupled into the multi-lane PHY and therefore this is very application dependent. The CLOCK is the only architecturally common element in the PHY that could cause errors across the lanes of a multi-lane PHY. Also, in practice the power supplies and ground are also common to multiple lanes.

Q. Would the CLOCK lane also have a BER of 10e-12?

A. As discussed in a previous question above, the concept of a BER for the CLOCK lane is inappropriate. Movement of CLOCK sampling edges out of spec could corrupt the sampled bit on each lane of a multilane PHY at the same time. As indicated above, there is a non-zero probability that movement of a CLOCK edge will have no effect on the recovered data. The probability of any errors due to CLOCK malfunctions is much lower for a source synchronous system than for an embedded clock system due to the architected coherence between the CLOCK and DATA lanes and the architected tolerance of the receiving hardware to clock period jitter.

Q. Could an error on the CLOCK lane corrupt more than one bit in the reception of the associated data packet?

A. D-PHY traffic is organized by transmissions, i.e. bit streams over each lane, not in "packets." Layers above the PHY impose the notion of packet on these transmissions. As discussed above, it is possible for a CLOCK edge movement due to noise to corrupt more than one bit in a set of transmissions across a multilane PHY. The probability of a CLOCK lane "error" causing multiple data errors is extremely low.

There is one other extreme low-probability error case that is worth mentioning. Since the D-PHY does not require a receiver PLL, it has no tolerance for edge movement of greater than 1 UI ("loss of a clock"). Occurrence of this error would cause the deserialization hardware to be out of synch with the transmitter serializer. This would result in all subsequent bits of the transmission to be transformed into gibberish due to the bits being placed in the wrong position in the byte. Since D-PHY has no encoded byte resynchronization, this situation would exist until the next End of Transmission (EOT) Stop state.

Q. What are the pros and cons of using Ultra Low Power Mode (ULPM)?

A. D-PHY has a low-power stop state, in which both lines of the lane are "high", i.e. at a voltage level of 1.2V nominal. In this state, the line currents are zero. However, in applications where there is no native 1.2V supply in the system, the 1.2V required for the LP transmitters has to be generated from some other rail (e.g. from 1.8V using an LDO), which will draw some current from the power source.

D-PHY defines an "ultra-low power mode", in which the lines are "low", i.e. at ground level. This allows the 1.2V supply to be turned off, saving power in the circuitry that generates the 1.2V.

Though ULPM has the benefit of saving on power consumption in many applications, it comes at a cost: longer start-up times when waking up from ULPM. There is also the burden of additional complexity from implementing Escape Mode functionality, since Escape Mode is needed to enter ULPM.

Q: What is the relation between Low-Power Mode and Ultra-Low Power Mode?

A: There are only two signaling modes: High-Speed and Low-Power signaling. Within the Low-Power mode there are two FUNCTIONAL sub-modes called Low-Power Control Mode and Low-Power Escape Mode.

Low-Power line signaling sequences have different meanings in these two functional modes. The Low-Power Control Mode includes the Stop state, the Turnaround sequence, and the sequence to enter Low-Power Escape Mode. Low-Power Escape Mode then obviously covers all Escape Mode functionality. All this is depicted by the flow diagram in Figure 24 of the specification.

Stop state (LP-11) only exists in Low-Power Control mode and is the central state of the operation diagram. This is also the normal standby state on the lines if there is no HS transmission, no Turnaround, and no Escape Request or Escape Mode operation. However, in applications without separate 1.2V supply, maintaining the LP-11 Stop state still costs some static power. Therefore a special Ultra Low-Power mode (or more accurately, Ultra Low-Power state) was defined where the line levels are LP-00 and supply values are irrelevant. In order to shut down various analog biases, a significant recovery time from ULP mode/state must be reserved.

It is important to notice that the ULP mode/state is entered differently for Clock and Data Lanes. For Clock Lanes, ULP mode/state is directly entered from Control Mode, a simple process given the absence of functionality that is necessary in (some) Data Lanes. For Data Lanes, the ULP mode/state is entered via a reserved Escape mode entry code.

The Ultra-Low Power mode/state is just one specific situation within Low-Power mode. Most likely you noticed that above the words 'mode' and 'state' are mentioned simultaneously for ULP. Currently the spec calls it a “mode,” but after completing D-PHY v0.65, the Phy WG acknowledged that using the term ”mode” is confusing; the D-PHY does not operate in the ULP “mode.” Because “state” seems a more appropriate

description to denote the ULP situation, it is the intention of the working group to rename ULP mode to ULP state.

Q. Why does D-PHY not specify any clock system?

A. As mentioned in the Scope section of the D-PHY spec, the intention of omitting any detailed specification of the clock signals was to allow the implementer design trade-offs. The Phy WG felt that the clock system was internal to each chip implementation and its only manifestation to the outside was via the DATA and CLOCK signals on the PHY interface. As long as these signals conformed to the D-PHY specification, no details of the clock system used in their generation were needed.

In particular, the Phy WG wished to avoid specifying such things as Clock reference source frequency or quality, PLL lock timings etc. Also, although Spread Spectrum Clocking is allowed in the signals on the PHY interface (see lines 1248-9 of the spec), the details of the spreading scheme in the clock system are unnecessary as long as the interface signals adhere to the D-PHY specification.

Q. How robust is the high-speed Start of Transmission (SoT) synchronization sequence?

A. The sequence was selected to allow the receiver to accurately find synchronization while tolerating one bit error in the sequence.

Q. How many bits should the receiver use to detect sync? The D-PHY specification suggests that the transmitter should transmit an 8-bit synchronizing byte '00011101' at the start of transmission. At the receiver, it seems to suggest that only the last six bits (011101) need to be detected. See table 3.

A. The receiver must use a minimum of six bits (allowing any single bit error) to determine sync. Using eight bits to detect synchronization may improve the quality of the receiver. The timing around the start of transmission is such that the receiver is guaranteed to always receive 8 synchronization bits.

Q. Can I build a receiver that is tolerant of two or more bit errors in the sync sequence?

A. In general, no. If two bit errors are tolerated in any bit locations, it is not always possible to determine with confidence where the first valid data bit occurs.

Q. Will ? Zolp(01,10) be removed from the D-Phy specification after v0.65?

A. It will be removed, since this particular spec applies to a contention detection function that is no longer part of the D-PHY Specification. The mismatch between the pull-up and pull-down output resistance of the low-power drivers (? Zolp(01,10)) was originally specified to enable the detection of a LP01 or LP10 state driving into a terminated High Speed receiver.

Through subsequent discussion in 2005, the detection of this fault condition was removed because it was clear that timers at the protocol level could also handle this case reliably, more simply, and with fewer burdens on the implementer. Moreover, NOT removing the detector would have had severe implications on PHY parameter accuracy.

Although the contention detector was eliminated before v0.65 was complete, some related parameter specifications persisted in the Spec. Although Phy WG expects that the presence of the parameter specs have little implication on design complexity, these are unnecessary constraints, so Phy WG plans to remove them from later drafts of D-PHY after v0.65.

Q. Why is End of Transmission [EOT] processing normative in the D-PHY specification (section 5.4.3) when this operation can be done outside the D-PHY for some protocols? EOT processing in the D-PHY requires a data buffer, which is not ideal for an I/O function.

A. All interfaces using D-PHY must perform EOT processing to identify the end of payload data. However, if the D-PHY is embedded in a design (the usual case), with no external visibility of the PPI signals, it is the choice of the implementer whether the EOT processing is done physically in the D-PHY or in the protocol processing.

Note that even if the EOT processing is done in the D-PHY, extra data may be sent to the protocol processing in the event that bit errors cause misidentification of the true EOT.

Q. How can I calculate the maximum operating data rate for a pair of components from their documented values of Maximum Data Rate and UIinst(min)?

A. The above values give the maximum performance for an individual component. To calculate the maximum operating data rate for a pair of components, it is *strongly recommended* that component data sheets provide tables of Data Rate and corresponding UIinst(min) with sufficient resolution over the full range supported by that component.

The maximum operating data rate is that rate where the tabulated parameters of Data Rate for the receiver equals or exceeds that of the transmitter *together* with the UIinst(min) of the receiver being equal or less than that of the transmitter.

No assumption should be made about the form of variation of UIinst(min) with Data Rate as this is affected by influences (e.g. clock source, protocol, etc.) outside the scope of the D-PHY spec. acquisition.