Proposed Draft

Serial ATA International Organization

Version 10
June 28, 2010

Serial ATA Revision 3.0 TP 029
Title : Clarification of Speed Negotiation

Author Information

Author Name	Company	Email address
Serge Bedwani	Intel	Serge.bedwani@intel.com
Chuck Hill	Alta Engineering	cphill@altaeng.com

Workgroup Chair Information

Workgroup (Phy, Digital, etc…)	Chairperson Name	Email address
Phy	Chuck Hill	cphill@altaeng.com

Document History

Version	Date	Comments
1	4 February 2010	Initial release.
2	17 February 2010	Added transition out of HP6 to clarify asynchronous reset
3	24 February 2010	Add another transition out of HP6 to clarify optional reset for speed negotiation
4	3 March 2010	Remove additional transitions out of HP6. Edit note 6 on HP6.
5	17 March 2010	Further edits on note 6, get diagrams edited from rev 3.0 and vote approval.
6	21 June 2010	Add description for figure describing the speed negotiation using COMRESET.
7	23 June 2010	Edits made during phy call. Added Reset Speed Negotiation term.
8		Invalid
9	28 June 2010	Editorial changes.

1 Introduction

1.1 Problem Statement

Documentation of existing methods of speed negotiation.

1.2 Solution Summary

1. Make more clear the spec allows an alternate method for Hosts to change speed (during HR_Reset), avoiding the 53.3nS limitation.

1.3 Background (optional)

The scope of the process of speed negotiation has grown beyond what was originally intended in response to implementation issues

The specification places a higher burden on Hosts than Devices in that 1) Hosts must determine the speed of incoming Aligns 2) Hosts must switch their speed in 53.3nS and 3) Hosts must not induce a condition where Device receivers cannot reacquire. Host designers have met these burdens by different means.

The process of changing speed can involve a simple switching of clocks and thus achieve the 53.3nS, or it may require changing a PLL which may need 100uS or more to change. Device designers are allowed 375uS (indirectly set by a 874uS timeout) ~~which allows the PLLs~~. However, Host designers faced with a 53.3nS limitation, implemented a reset based scheme to allow them enough time to change speed. This is currently allowed by the spec.

The spec currently allows a speed negotiation scheme using a COMRESET command to achieve software control of the speed negotiation process.

2 Technical Specification Changes

Table 51 – State Diagram Host Phy Initialization State Machine

HP6: HR_AwaitAlign	Host transmits D10.2 characters at lowest supported rate[2,5]	
1. If reset speed negotiation is not supported and $ALIGN_P$ detected from device (at any supported speed)[3]. If reset speed negotiation is supported and $ALIGN_P$ detected from device at expected speed.	→	HR_AdjustSpeed
2. $ALIGN_P$ not detected from device and 873.8 us (32768 Gen1 Dwords) has elapsed since entry to HR_AwaitAlign. If reset speed negotiation is supported and $ALIGN_P$ is detected at a lower speed.	→	HR_Reset[1,4,6]
3. $ALIGN_P$ not detected from device and less than 873.8 us (32768 Gen1 Dwords) has elapsed since entry to HR_AwaitAlign.	→	HR_AwaitAlign

NOTES:
1. Host retries the power-on sequence indefinitely unless explicitly turned off by the Application layer.
2. Host shall start transmitting D10.2 characters no later than 533 ns (20 Gen1 Dwords) after COMWAKE is negated as specified in the OOB signaling section.
3. Host designers should be aware that the device is allowed 53.3 ns (2 Gen1 Dwords) after releasing COMWAKE (by holding the idle condition for more than 175 ns) to start sending characters. Until this occurs, the bus is at an idle condition and may be susceptible to crosstalk from other devices. Care should be taken so that crosstalk during this window doesn't result in a false detection of an $ALIGN_P$. For example: a compliant host may detect the negation of COMWAKE in as little as 112 ns, such a host should wait at least 116.3 ns (175+53.3-112) after detecting the release of COMWAKE to start looking for $ALIGN_P$ primitives.
4. The Host Phy Initialization state machine may use the transition to HR_Reset as a method of speed negotiation.
5. The Device may respond with D10.2 when out of lock (see DR_SendAlign).
6. If $ALIGN_P$ is detected at a lower speed, the host may transition to HR_Reset before the 873.8us timeout thus reducing the time for speed negotiation.(RSN)

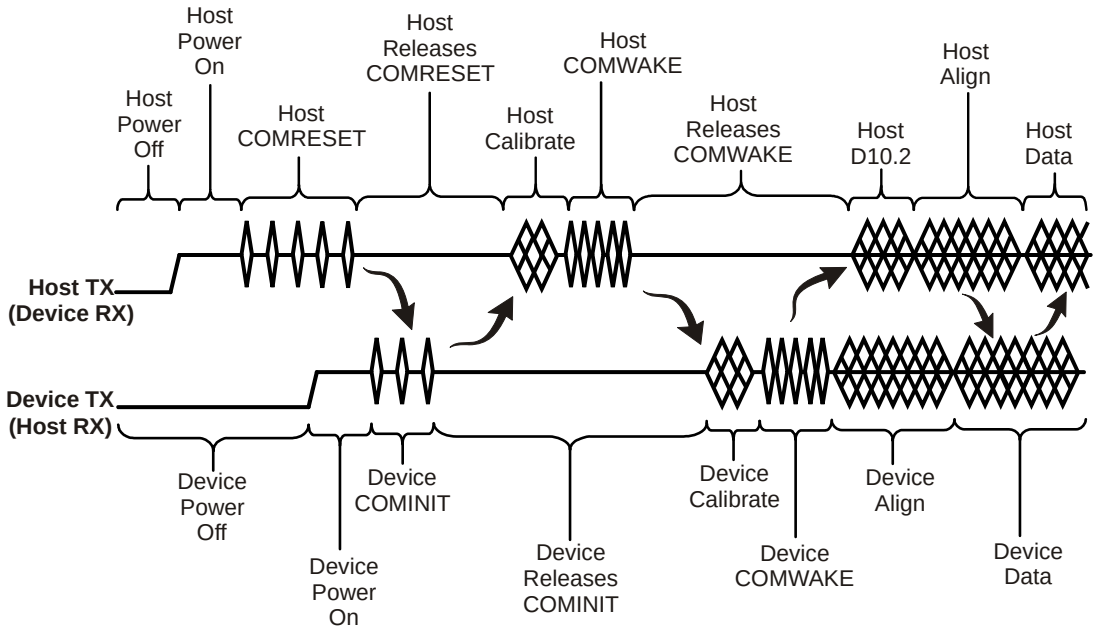

Figure 187 – Power-On Sequence

Description:

1. Host/device power-off - Host and device power-off.
2. Power is applied - Host side signal conditioning pulls TX and RX pairs to neutral state (common mode voltage).
3. Host issues COMRESET
4. Host releases COMRESET. Once the power-on reset is released, the host releases the COMRESET signal and puts the bus in a quiescent condition.
5. Device issues COMINIT – When the device detects the release of COMRESET, it responds with a COMINIT. This is also the entry point if the device is late starting. The device may initiate communications at any time by issuing a COMINIT.
6. Host calibrates and issues a COMWAKE.
7. Device responds – The device detects the COMWAKE sequence on its RX pair and calibrates its transmitter (optional). Following calibration the device sends a six burst COMWAKE sequence and then sends a continuous stream of the ALIGN sequence starting at the device's highest supported speed. After $ALIGN_P$ primitives have been sent for 54.6 us (2048 nominal Gen1 Dword times) without a response from the host as determined by detection of $ALIGN_P$ primitives received from the host, the device assumes that the host cannot communicate at that speed. If additional speeds are available the device tries the next lower supported speed by sending $ALIGN_P$ primitives at that speed for 54.6 us (2048 nominal Gen1 Dword times.) This step is repeated for as many slower speeds as are supported. Once the lowest speed has been reached without response from the host, the device shall enter an error state.
8. Host locks – after detecting the COMWAKE, the host starts transmitting D10.2 characters (see **Error! Reference source not found.**) at its lowest supported speed. Meanwhile, the host receiver locks to the ALIGN sequence and, when ready, returns the ALIGN sequence to the device at the same speed as received. A host shall be designed such that it acquires lock in 54.6 us (2048 nominal Gen1 Dword times) at any given speed. The host should allow for at least 873.8 us (32768 nominal Gen1 Dword times) after detecting the release of COMWAKE to receive the first $ALIGN_P$. This insures interoperability with multi-generational and synchronous designs. If no $ALIGN_P$ is received within 873.8 us (32768 nominal Gen1 Dword times) the host restarts the power-on sequence – repeating indefinitely until told to stop by the Application layer.

9. Device locks – the device locks to the ALIGN sequence and, when ready, sends the $SYNC_P$ primitive indicating it is ready to start normal operation.
10. Upon receipt of three back-to-back non-$ALIGN_P$ primitives, the communication link is established and normal operation may begin.

Editor's note: the following diagram is to be inserted at this place in the document. It is not possible to highlight this in red. This is in section 8.4.3.

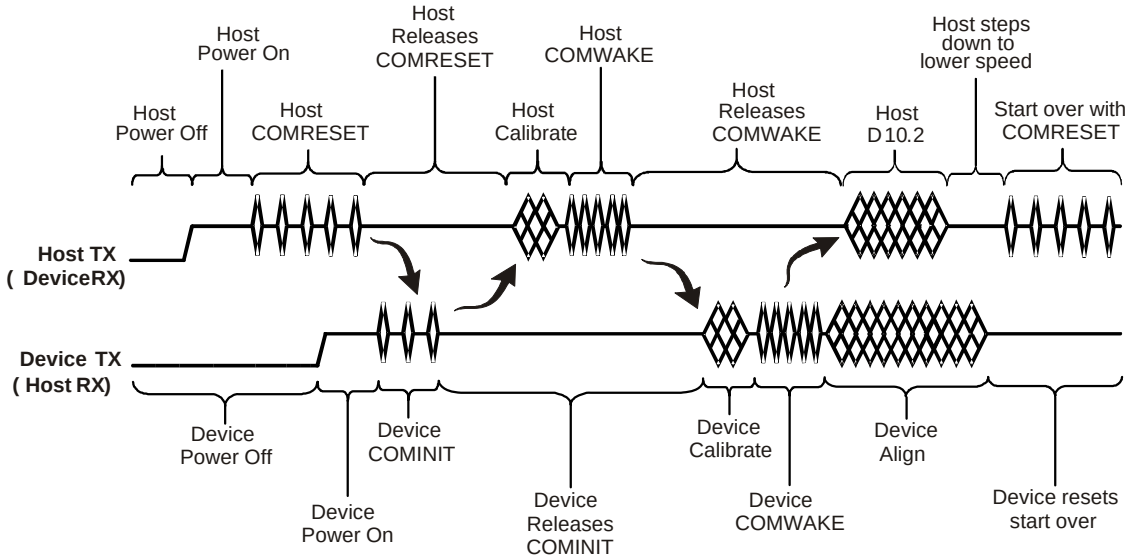

Figure 187B: Speed Negotiation using COMRESET (RSN)

Description:

1. Host/device are powered and operating normally with some form of active communication.
2. Some condition in the host causes the host to issue COMRESET
3. Host releases COMRESET. Once the condition causing the COMRESET is released, the host releases the COMRESET signal and puts the bus in a quiescent condition.
4. Device issues COMINIT – When the device detects the release of COMRESET, it responds with a COMINIT. This is also the entry point if the device is late starting. The device may initiate communications at any time by issuing a COMINIT.
5. Host calibrates and issues a COMWAKE.
6. Device responds – The device detects the COMWAKE sequence on its RX pair and calibrates its transmitter (optional). Following calibration the device sends a six burst COMWAKE sequence and then sends a continuous stream of the ALIGN sequence starting at the device's highest supported speed. After $ALIGN_P$ Dwords have been sent for 54.6us (2048 nominal Gen1 Dword times) without a response from the host as determined by detection of $ALIGN_P$ primitives received from the host, the device assumes that the host cannot communicate at that speed. If additional speeds are available the device tries the next lower supported speed by sending $ALIGN_P$ Dwords at that speed for 54.6 us (2048 nominal Gen1 Dword times.) This step is repeated for as many slower speeds as are supported. Once the lowest speed has been reached without response from the host, the device enters an error state.

7. Host locks – after detecting the COMWAKE, the host starts transmitting D10.2 characters (see **Error! Reference source not found.**) at its lowest supported speed. Meanwhile, the host receiver locks to the ALIGN sequence and, when ready, returns the ALIGN sequence to the device at the same speed as received. A host shall be designed such that it acquires lock in 54.6us (2048 nominal Gen1 Dword times) at any given speed. The host should allow for at least 873.8 us (32768 nominal Gen1 Dword times) after detecting the release of COMWAKE to receive the first $ALIGN_P$. This ensures interoperability with multi-generational and synchronous designs. If no $ALIGN_P$ is received within 873.8 us (32768 nominal Gen1 Dword times) the host restarts the power-on sequence – repeating indefinitely until told to stop by the Application layer. A Host, internally operating at a higher speed than the Device, that has the capability of detecting the incoming stream of lower speed $ALIGN_P$ Dwords may issue a COMRESET on detection and rate verification of the lower speed $ALIGN_P$ Dwords. This Host initiated COMRESET sequence would allow the Host to adjust its speed to match the lower speed Device and resume the speed negotiation sequence as described above at the matched rate, thereby reducing the time for the speed negotiation process. This is illustrated in Figure 187B.
8. Device locks – the device locks to the ALIGN sequence and, when ready, sends $SYNC_P$ indicating it is ready to start normal operation.
9. Upon receipt of three back-to-back non-$ALIGN_P$ primitives, the communication link is established and normal operation may begin.

8.4.3.2 Partial/Slumber to PHYRDY

(Editors note: Add to definitions section 4)
Reset Speed Negotiation (RSN) – The use of COMRESET to accomplish speed negotiation.