

**Proposed
Draft**

**Serial ATA
International Organization**

**Version 2
August 16, 2012**

**Serial ATA Revision 3.1 ECN # 061
Title : Dual Consecutive ALIGNp Sequence**

Permission is granted to the Promoters, Contributors and Adopters of the Serial ATA International Organization to reproduce this document for the purposes of evolving the technical content for internal use only without further permission provided this notice is included. All other rights are reserved and may be covered by one or more Non Disclosure Agreements including the Serial ATA International Organization participant agreements. Any commercial or for-profit replication or republication is prohibited. Copyright © 2000 to 2012 Serial ATA International Organization. All rights reserved.

The Serial ATA International Organization, Serial ATA Ratified ECN ("Final Specification") is available for download at <http://www.sata-io.org>.

SPECIFICATION DISCLAIMER

THIS SPECIFICATION IS PROVIDED TO YOU "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE. THE AUTHORS OF THIS SPECIFICATION DISCLAIM ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY PROPRIETARY RIGHTS, RELATING TO USE OR IMPLEMENTATION OF INFORMATION IN THIS SPECIFICATION. THE AUTHORS DO NOT WARRANT OR REPRESENT THAT SUCH USE WILL NOT INFRINGE SUCH RIGHTS. THE PROVISION OF THIS SPECIFICATION TO YOU DOES NOT PROVIDE YOU WITH ANY LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS.

Copyright 2002 to 2012, Serial ATA International Organization. All rights reserved.

For more information about Serial ATA, refer to the Serial ATA International Organization website at <http://www.sata-io.org>.

All product names are trademarks, registered trademarks, or servicemarks of their respective owners.

Serial ATA International Organization contact information:

SATA-IO
3855 SW 153rd Drive
Beaverton, Oregon 97006 USA
Tel: +1 503-619-0572
Fax: +1 503-644-6708
E-mail: admin@sata-io.org

Author Information

Author Name	Company	Email address
Harvey Newman	LSI Corporation	harvey.newman@lsi.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
0	6/11/12	Initial release.
1	6/12/2012	Changed section number and title for 7.2.4.1, other minor stuff
2	8/16/2012	Ratified

1 Introduction

1.1 Problem Statement

The phrase “align sequence” is used with two different meanings: a continuous stream of ALIGN_P primitives, or two consecutive ALIGN_P primitives. The phrase “dual consecutive ALIGN sequence” has been misinterpreted to mean two consecutive ALIGN_P primitives, when the correct interpretation is four consecutive ALIGN_P primitives.

1.2 Solution Summary

Use distinctive descriptions for each different use, define “Align sequence” for one, and use plain language for the other.

The first use of ALIGN_P primitives is in a continuous stream during the Phy initialization for setting Dword alignment and speed detection. This will be described as an “Align sequence” and defined.

The second use is of ALIGN_P primitives to manage the elasticity buffer. There are two situations: normal data transfer, and self test loopback. During normal data transfer, ALIGN_P primitives are always inserted by a transmitter two at a time which will be described as “two consecutive ALIGN_P primitives”. During self test loopback; transmitters insert ALIGN_P primitives four at a time and is currently described as “dual consecutive Align sequences” which will be changed to “four consecutive ALIGN_P primitives”.

The third use of ALIGN_P primitives is as a pattern during the OOB signals.

There is no change to any specification requirements, these changes are only to clarify the meaning of the existing requirements.

1.3 Background (optional)

1.3.1 Precedents

There are precedents for the use of the phrase “continuous primitives” or “consecutive primitives”:

- 1) section 10.3.9.1 Description “BIST Activate FIS should transmit continuous SYNC_P primitives after reception of R_OK_P ”.
- 2) section 15.3.2.1 Invalid State Transitions “one or more consecutive X_RDY_P at the receiver interface”.
- 3) section 16.3.3.7 Reducing Context Switching Complexity “Port Multiplier has received at least two consecutive SYNC_P primitives from the host”.

1.3.2 Consistent Descriptions (Sections not to be changed)

These sections have no changes because the “Align sequence” definition to be added makes the meaning clear: an Align sequence is a continuous stream of consecutive ALIGN_P primitives.

7.5.1.2 COMRESET

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 7.6) 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.
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.

7.5.1.3 COMINIT

1. Host/device are powered and operating normally with some form of active communication.
2. Some condition in the device causes the device to issues a COMINIT
3. Host calibrates and issues a COMWAKE.
4. 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 54.6 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 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.

5. Host locks – after detecting the COMWAKE, the host starts transmitting D10.2 characters (see section 7.6) 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 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.
6. Device locks – the device locks to the ALIGN sequence and, when ready, sends SYNC_P indicating it is ready to start normal operation.
7. Upon receipt of three back-to-back non-ALIGN_P primitives, the communication link is established and normal operation may begin.

8.4.3.1 Power-On Sequence Timing Diagram

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 7.6) 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.

9.6.5 Link Mode Power State Diagram

LPM8: L_WakeUp2 state: This state is entered when the Phy layer has acknowledged an initiated wakeup request by asserting its PHYRDY signal. In this state, the Link layer shall transmit the ALIGN sequence, and transition to the L1: L_IDLE state.

1.3.3 Technical Distinction

During self test loopback, described as BIST L, the aggregate data rate less ALIGN_p primitives must match on receive and transmit side of the device in loopback. The transmitter and receiver do not have the exact same clock speed, but rather one can be high by 350ppm and the other low by 5350ppm; the frequency difference is 5700ppm or 0.57%. Under normal transmission, the ALIGN_p primitives are transmitted as two every 254 Dwords which enables the link to tolerate a frequency difference of $(256 - 254)/256 = 0.78\%$ (for every 256 period, it transmits only 254). In order for the transmitter port of the device in loopback to provide ALIGN_p primitives to ensure Dword alignment, the specification currently requires four consecutive ALIGN_p primitives be at the loopback device receive port; it is currently described as “dual consecutive Align sequences” which is unclear.

2 Technical Specification Changes

2.1 Titles of Sections to be Changed

4.1.1.2 ALIGN Sequence

A continuous stream of consecutive ALIGN_P primitives.

7.2.4 Test Pattern Requirements

Compliant patterns are those specified patterns that contain the leading SOF_P primitive, the specified pattern as data content, and trailing CRC_P and EOF_P primitives. There is no suppression of the ~~dual-consecutive~~ two consecutive ALIGN_P primitives during stimulus with this class of pattern.

7.4.2.4 Bit Error Rate Testing (Informative)

In order to get a fair assessment of bit-error-rate performance, bit-errors, as well as burst errors, are considered separately. This is because a missing or an extra bit detected by the receiver translates into a series of errors that spans across multiple byte boundaries until re-alignment via an alignment sequence. This series of errors are defined as burst errors.

Another type of byte-wise error exists when an entire byte is not received. As viewed by the higher-level protocol it appears as a loss of word synchronization. It causes a burst error whose span may be limited by higher-layer protocol transmission conventions at the next alignment sequence.

7.5.1 Out Of Band Signaling

Previous versions of Serial ATA allow only for the ~~ALIGN-sequence~~ ALIGN_P primitives as legitimate OOB signal content. The alternate OOB sequence defined in this section has different characteristics than the ~~ALIGN-sequence~~ ALIGN_P primitives in both the time and frequency domains. The use of alternate OOB signal content may lead to backwards incompatibility with Gen1 Phys designed to previous Serial ATA specification versions. Interoperability issues with Gen1 Phys designed to the earlier SATA specification arising from the use of alternate OOB signal content are the sole responsibility of the Phy transmitting this alternate content.

7.6 Elasticity Buffer Management

The Link layer shall keep track of a resettable counter that rolls over at most every 1024 transmitted characters (256 Dwords). Prior to, or at the pre-roll-over point (all 1's), the Link layer shall trigger the issuance of ~~dual,~~ two consecutive ALIGN_P primitives which shall be included in the Dword count.

After communications have been established, the first and second words out of the Link layer shall be ~~the dual-ALIGN_P-primitive-sequence~~ two consecutive ALIGN_P primitives, followed by at most 254 non-ALIGN_P Dwords. The cycle repeats starting with ~~another dual-consecutive-ALIGN_P-primitive-sequence~~ two consecutive ALIGN_P primitives. The Link may issue more than ~~one dual-ALIGN_P-primitive-sequence~~ a single instance of two consecutive ALIGN_P primitives but shall not send an ~~unpaired~~ odd number of ALIGN_P primitives (i.e. ALIGN_P primitives are always sent in pairs) except as noted for retimed loopback.

8.4.3 Speed Negotiation

To reduce susceptibility to false ALIGN_P detection/handshake, receivers should fully qualify the entire received ~~ALIGN sequence~~ ALIGN_P primitives instead of relying on qualifying only a portion of ~~# each~~ (such as just the comma sequence). Additional means for ensuring that the transition from the HP6:HR_AwaitAlign is accurately traversed and not traversed in response to a spurious signal from the data recovery circuit is to ensure that a series of contiguous ALIGN_P primitives are successfully decoded. Other possible means for ensuring accuracy of the ALIGN_P detection are also possible.

9.4.5.2 Periodic Retransmission of Sustained Primitives (Informative)

In order to be able to determine the state that a bus is in, it is recommended that a sustained primitive periodically be retransmitted. The only requirement is that the interval at which the retransmit occurs is large enough that EMI is not substantially affected. Since ~~the ALIGN sequence is two consecutive~~ Align primitives are required to be sent at an interval of at most 256 Dwords, one solution to providing visibility to a suppressed primitive stream is to retransmit the suppressed primitive sequence immediately after the ALIGN_P primitives are inserted. For example, if the original sequence was PRIM / PRIM / CONT_P / junk . . . ALIGN_P / ALIGN_P / junk . . . the new sequence could look like: PRIM / PRIM / CONT_P / junk . . . ALIGN_P / ALIGN_P / PRIM / PRIM / CONT_P / junk.

10.3.9.1 Description [Editors note: Far End retimed loop]

L: The Far End Retimed LoopbackMode is defined as a mode where the receiver retimes the data, and retransmits the retimed data. The initiator of the retimed loopback mode shall account for the loopback device consuming up to two ALIGN_P primitives (~~one ALIGN sequence~~) every 256 Dwords transmitted and, if it requires any ALIGN_P primitives to be present in the returned data stream, it should insert additional ALIGN_P primitives in the transmitted stream. The initiator shall transmit additional ~~ALIGN sequences~~ two consecutive ALIGN_P primitives in a single burst at the normal interval of every 256 Dwords transmitted (as opposed to inserting ~~ALIGN sequences~~ two consecutive ALIGN_P primitives at half the interval).

The loopback device may remove zero, one, or two ALIGN_P primitives from the received data. It may insert one or more ALIGN_P primitives if they are directly preceded or followed by the initiator inserted ALIGN_P primitives (resulting in ~~ALIGN sequences consisting of~~ at least two consecutive ALIGN_P primitives) or it may insert two or more ALIGN_P primitives if not preceded or followed by the initiator's ALIGN_P primitives. One side effect of the loopback retiming is that the returned data stream may have instances of an odd number of ALIGN_P primitives, however, returned ALIGN_P primitives are always in bursts and if the initiator transmitted ~~dual ALIGN sequences~~ (four consecutive ALIGN_P primitives), then the returned data stream shall include ALIGN_P bursts that are no shorter than two ALIGN_P primitives long (although the length of the ALIGN_P burst may be odd). The initiator of the retimed loopback mode shall not assume any relationship between the relative position of the ALIGN_P primitives returned by the loopback device and the relative position of the ALIGN_P primitives sent by the initiator.

A: ALIGN_P ~~sequence~~ primitive bypass mode. When set to one, no ALIGN_P primitives are sent. When the A-bit is not asserted, ALIGN_P primitives are sent normally as defined in this document. The setting of this bit is applicable only when the T bit is set.