SAS compliant jitter test pattern T10/02-390r0

Date: October 14, 2002
To: T10 Technical Committee
From: Alvin Cox (alvin.cox@seagate.com), Bernhard Laschinsky (blaschinsky@agere.com)
Subject: SAS compliant jitter test pattern

A jitter test pattern needs to be defined for SAS that takes into consideration running disparity and data scrambling to achieve the proper “on-the-wire” data sequence within a compliant protocol frame. The following proposed addition to the SAS specification defines the test pattern CJTPAT and includes an informative annex that gives guidance on how to achieve the desired “on-the-wire” pattern.

5.7.xx Jitter characteristics test pattern

The jitter test pattern, CJTPAT, shall be used for all jitter testing unless otherwise specified. CJTPAT consists of a long run of low-density pattern, followed by a long run of high transition density pattern, followed by another short run of low-density pattern. It is the transitions between the pattern segments that stress the receiver because it exposes the clock and data recovery circuitry to large phase shifts. The test pattern was designed to contain the phase shift in both polarities, from 0 to 1 and from 1 to 0. Due to scrambling and running disparity, special considerations must be made to achieve the correct data pattern “on-the-wire”. Refer to Annex xx for additional information regarding these considerations.

Annex xx
(Informative)

xx.1 Scrambling and disparity considerations for achieving a proper on-the-wire CJTPAT

The basic Jitter Tolerance Test Pattern is listed in Table 1 which shows both the 10b test pattern and, in Dxx.y and hexadecimal notation, the corresponding 8b pattern that should be input into the 8b / 10b encoder in the transmitter to result in the desired 10b pattern "on the wire".

The pattern consists of a long run of low-density pattern, followed by a long run of high transition density pattern, followed by another short run of low-density pattern. It is the transitions between the pattern segments that stress the receiver because it exposes the clock and data recovery circuitry to large phase shifts. The test pattern was designed to contain the phase shift in both polarities, from 0 to 1 and from 1 to 0. The critical pattern sections with the phase shifts are highlighted in the table.

Table 1 - Jitter Tolerance Test Pattern for RD+

+	D30.3(7Eh)			D30.3(7Eh)		D30.3(7Eh)			D30.3(7Eh)		+
	1000	0111	0001	1110	0011	1000	0111	0001	1110	0011	
	Above 4 byte (Dword) low density pattern is repeated 41 times										
+	D30.3(7Eh)			D30.3(7Eh)		D30.3(7Eh)			D20.3(74h)		-
	1000	0111	0001	1110	0011	1000	0111	**0000**	**10**11	1100	
	Phase shift 11100001011										
-	D30.3(7Eh)			D11.5(ABh)		D21.5(B5h)			D21.5(B5h)		+
	0111	1000	**1111**	**01**00	1010	1010	1010	1010	1010	1010	
	Phase shift 00011110100										
+	D21.5(B5h)			D21.5(B5h)		D21.5(B5h)			D21.5(B5h)		+
	1010	1010	1010	1010	1010	1010	1010	1010	1010	1010	
	Above 4 byte (Dword) high density pattern is repeated 12 times										
+	D21.5(B5h)			D30.2(5Eh)		D10.2(4Ah)			D30.3(7Eh)		+
	1010	1010	1**010**	**0001**	0101	0101	0101	0**101**	**1110**	0011	
	Phase shift 01010000 and 10101111										

Table 1 assumes a positive running disparity (RD+) at the beginning of the 8b pattern. If the 8b pattern shown in Table 1 is encoded with negative starting running disparity (RD-), the resulting 10b pattern will be different and does not provide the critical phase shifts. To achieve the same phase shift effects with RD-, a different 8b pattern is required, which is shown in TABLE 2 with the 10b pattern resulting from encoding with RD-. Note that the 8B pattern in TABLE 2 will not give a proper 10b pattern if it is encoded with RD+.

TABLE 2 - JITTER TOLERANCE TEST PATTERN FOR RD-

-	D30.3(7Eh)			D30.3(7Eh)		D30.3(7Eh)			D30.3(7Eh)		-
	0111	1000	1110	0001	1100	0111	1000	1110	0001	1100	
	Above 4 byte (dword) low density pattern is repeated 41 times										
-	D30.3(7Eh)			D30.3(7Eh)		D30.3(7Eh)			D11.3(6Bh)		+
	0111	1000	1110	0001	1100	0111	1000	**1111**	**01**00	0011	
	Phase shift 00011110100										
+	D30.3(7Eh)			D20.2(54h)		D10.2(4Ah)			D10.2(4Ah)		-
	1000	0111	**0000**	**10**11	0101	0101	0101	0101	0101	0101	
	Phase shift 11100001011										
-	D10.2(4Ah)			D10.2(4Ah)		D10.2(4Ah)			D10.2.(4Ah)		-
	0101	0101	0101	0101	0101	0101	0101	0101	0101	0101	
	Above 4 byte (dword) high density pattern is repeated 12 times										
-	D10.2(4Ah)			D30.5(BEh)		D21.5(B5h)			D30.3(7Eh)		-
	0101	0101	0**101**	**1110**	1010	1010	1010	1**010**	**0001**	1100	
	Phase shift 10101111 and 01010000										

To use the Jitter Tolerance Test Pattern as the payload in a protocol frame, the 8b patterns for both RD+ and RD- should be included, an example of which is shown in Table 3. The 10b pattern resulting from encoding the 8b pattern in Table 3 will contain the desired bit sequences for the phase shifts in both RD.

Table 3 - Jitter Tolerance Pattern for RD+ and RD-

D30.3(7Eh)	D30.3(7Eh)	D30.3(7Eh)	D30.3(7Eh)
Above dword is repeated 41 times			
D30.3(7Eh)	D30.3(7Eh)	D30.3(7Eh)	D20.3(74h)
D30.3(7Eh)	D11.5(ABh)	D21.5(B5h)	D21.5(B5h)
D21.5(B5h)	D21.5(B5h)	D21.5(B5h)	D21.5(B5h)
Above dword is repeated 12 times			
D21.5(B5h)	D30.2(5Eh)	D10.2(4Ah)	D30.3(7Eh)
D30.3(7Eh)	D30.3(7Eh)	D30.3(7Eh)	D30.3(7Eh)
Above dword is repeated 41 times			
D30.3(7Eh)	D30.3(7Eh)	D30.3(7Eh)	D11.3(6Bh)
D30.3(7Eh)	D20.2(54h)	D10.2(4Ah)	D10.2(4Ah)
D10.2(4Ah)	D10.2(4Ah)	D10.2(4Ah)	D10.2.(4Ah)
Above dword is repeated 12 times			
D10.2(4Ah)	D30.5(BEh)	D21.5(B5h)	D30.3(7Eh)

Before the pattern described in Table 3 can be encapsulated in a protocol frame, the effect of the scrambling of data in the transmitter before the 8b / 10b encoding should be compensated for. This is

done by scrambling the desired 8b pattern prior to submitting it to the transmitter scrambler. The scrambling in the transmitter scrambler will reverse the prior scrambling of the 8b pattern and the desired pattern will be presented to the 8b / 10b encoder.

The 8b data are scrambled by XOR-ing the pattern with the output of the scrambler Dword by Dword, taking into account the position of the 8b pattern within the protocol frame. Table 4 shows this principle for the pattern from Table 3 embedded in a SSP protocol frame with 24-byte address following the SOF primitive.

The columns titled "8b Data" lists the desired 8b pattern data that is to be 8b / 10b encoded.

The column titled "Scrambler Output" lists the output, in Dword format, of the transmit scrambler.

The column titled "Scrambled 8b Data" shows the result of XORing the 8b data with the scrambler output.

Note that the scrambler gets initialized (seeded) at the beginning of each frame (SOF) and the scrambler output is independent of the scrambled data.

Table 4 - Modified CJTPAT scrambled in SSP protocol frame

Frame Element	8B Data	Scrambler Output (SCR)	Scrambled 8B Data = 8B ⊕ SCR
SOF		n/a	n/a
Address	xxxxxxxx	C2D2768D	xxxxxxxx
	xxxxxxxx	1F26B368	xxxxxxxx
	xxxxxxxx	A508436C	xxxxxxxx
	xxxxxxxx	3452D354	xxxxxxxx
	xxxxxxxx	8A559502	xxxxxxxx
	xxxxxxxx	BB1ABE1B	xxxxxxxx
Pattern data	7E7E7E7E	FA56B73D	8428C943
	7E7E7E7E	53F60B1B	2D887565
	7E7E7E7E	F0809C41	8EFEE23F
	7E7E7E7E	747FC34A	0A01BD34
	7E7E7E7E	BE865291	C0F82CEF
	7E7E7E7E	7A6FA7B6	0411D9C8
	7E7E7E7E	3163E6D6	4F1D98A8
	7E7E7E7E	F036FE0C	8E488072
	7E7E7E7E	1EF3EA29	608D9457
	7E7E7E7E	EB342694	954A58EA
	7E7E7E7E	53853B17	2DFB4569
	7E7E7E7E	E94ADC4D	9734A233
	7E7E7E7E	5D200E88	235E70F6
	7E7E7E7E	6901EDD0	177F93AE
	7E7E7E7E	FA9E38DE	84E046A0
	7E7E7E7E	68DB4B07	16A53579
	7E7E7E7E	450A437B	3B743D05
	7E7E7E7E	960DD708	E873A976
	7E7E7E7E	3F35E698	414B98E6
	7E7E7E7E	FE7698A5	8008E6DB
	7E7E7E7E	C80EF715	B670896B
	7E7E7E7E	666090AF	181EEED1
	7E7E7E7E	FAF0D5CB	848EABB5
	7E7E7E7E	2B82009F	55FC7EE1
	7E7E7E7E	0E317491	704F0AEF
	7E7E7E7E	76F46A1E	088A1460
	7E7E7E7E	F46D6948	8A131736
	7E7E7E7E	7BCD8A93	05B3F4ED
	7E7E7E7E	1513AD7E	6B6DD300

	7E7E7E7E	1E72FEEE	600C8090
	7E7E7E7E	A014AA3B	DE6AD445
	7E7E7E7E	23AAD4E7	5DD4AA99
	7E7E7E7E	B0DC9E67	CEA2E019
	7E7E7E7E	E0A573FB	9EDB0D85
	7E7E7E7E	06CA944F	78B4EA31
	7E7E7E7E	63E29212	1D9CEC6C
	7E7E7E7E	4578626D	3B061C13
	7E7E7E7E	53260C93	2D5872ED
	7E7E7E7E	3E592202	40275C7C
	7E7E7E7E	2B6ECA63	5510B41D
	7E7E7E7E	636A1F1F	1D146161
	7E7E7E74	35B5A9ED	4BCBD799
	7EABB5B5	4AA2A0FD	34091548
	B5B5B5B5	71AFE196	C41A5423
	B5B5B5B5	E1D57B62	5460CED7
	B5B5B5B5	55A0568A	E015E33F
	B5B5B5B5	82D18968	37643CDD
	B5B5B5B5	234CB4FF	96F9014A
	B5B5B5B5	83481E7F	36FDABCA
	B5B5B5B5	B21AE87F	07AF5DCA
	B5B5B5B5	A9C5EACD	1C705F78
	B5B5B5B5	6201ACC3	D7B41976
	B5B5B5B5	F60939CE	43BC8C7B
	B5B5B5B5	395F767D	8CEAC3C8
	B5B5B5B5	2FA55841	9A10EDF4
	B55E4A7E	836D4A7A	36330004
	7E7E7E7E	388D587A	46F32604
	7E7E7E7E	773DFF5C	09438122
	7E7E7E7E	3C239CB3	425DE2CD
	7E7E7E7E	564D91A0	2833EFDE
	7E7E7E7E	43ED0BE1	3D93759F
	7E7E7E7E	987429A7	E60A57D9
	7E7E7E7E	E52DDBA2	9B53A5DC
	7E7E7E7E	E78DC87F	99F3B601
	7E7E7E7E	0AB8C669	74C6B817
	7E7E7E7E	64D083C9	1AAEFDB7
	7E7E7E7E	053DF93A	7B438744
	7E7E7E7E	EEE9D9EA	9097A794
	7E7E7E7E	44BD3B97	3AC345E9
	7E7E7E7E	0FE24B8C	719C35F2
	7E7E7E7E	F28D5694	8CF328EA
	7E7E7E7E	6310B6D9	1D6EC8A7
	7E7E7E7E	1792AECE	69ECD0B0
	7E7E7E7E	0A562EA1	742850DF
	7E7E7E7E	B048DF69	CE36A117
	7E7E7E7E	161A2878	68645606
	7E7E7E7E	5519CB51	2B67B52F
	7E7E7E7E	19F5BE56	678BC028
	7E7E7E7E	EFFFB4B6	9181CAC8
	7E7E7E7E	B3826E72	CDFC100C
	7E7E7E7E	E4722DDA	9A0C53A4
	7E7E7E7E	60BF5129	1EC12F57
	7E7E7E7E	248D90F5	5AF3EE8B
	7E7E7E7E	4D06D21C	3378AC62
	7E7E7E7E	7E96166C	00E86812
	7E7E7E7E	5FAFE3B4	21D19DCA
	7E7E7E7E	506CB855	2E12C62B
	7E7E7E7E	5BF03098	258E4EE6
	7E7E7E7E	46D4B6B3	38AAC8CD
	7E7E7E7E	051B9E11	7B65E06F
	7E7E7E7E	015CC556	7F22BB28

	7E7E7E7E	E21035EF	9C6E4B91
	7E7E7E7E	56604D75	281E330B
	7E7E7E7E	2E76675C	50081922
	7E7E7E7E	071476F0	796A088E
	7E7E7E7E	AFF087EB	D18EF995
	7E7E7E7E	1B62DB01	651CA57F
	7E7E7E6B	23661F6C	5D186107
	7E544A4A	F877B027	8623FA6D
	4A4A4A4A	F5E389A2	BFA9C3E8
	4A4A4A4A	EEC73611	A48D7C5B
	4A4A4A4A	4C04FB93	064EB1D9
	4A4A4A4A	E8D70F32	A29D4578
	4A4A4A4A	BFF03C54	F5BA761E
	4A4A4A4A	E3403C01	A90A764B
	4A4A4A4A	20FACA7E	6AB08034
	4A4A4A4A	9942458C	D3080FC6
	4A4A4A4A	37E2CB89	7DA881C3
	4A4A4A4A	5A1A9783	1050DDC9
	4A4A4A4A	CE48AA3F	8402E075
	4A4A4A4A	06C9A761	4C83ED2B
	4ABEB57E	06C03EAB	4C7E8BD5
CRC	xxxxxxxx	xxxxxxxx	xxxxxxxx
EOF		N/A	N/A