Setup and Hold Tests
Using Ultra2 Data/REQ Signals

Dan Smith
Seagate Technology
5/2/98

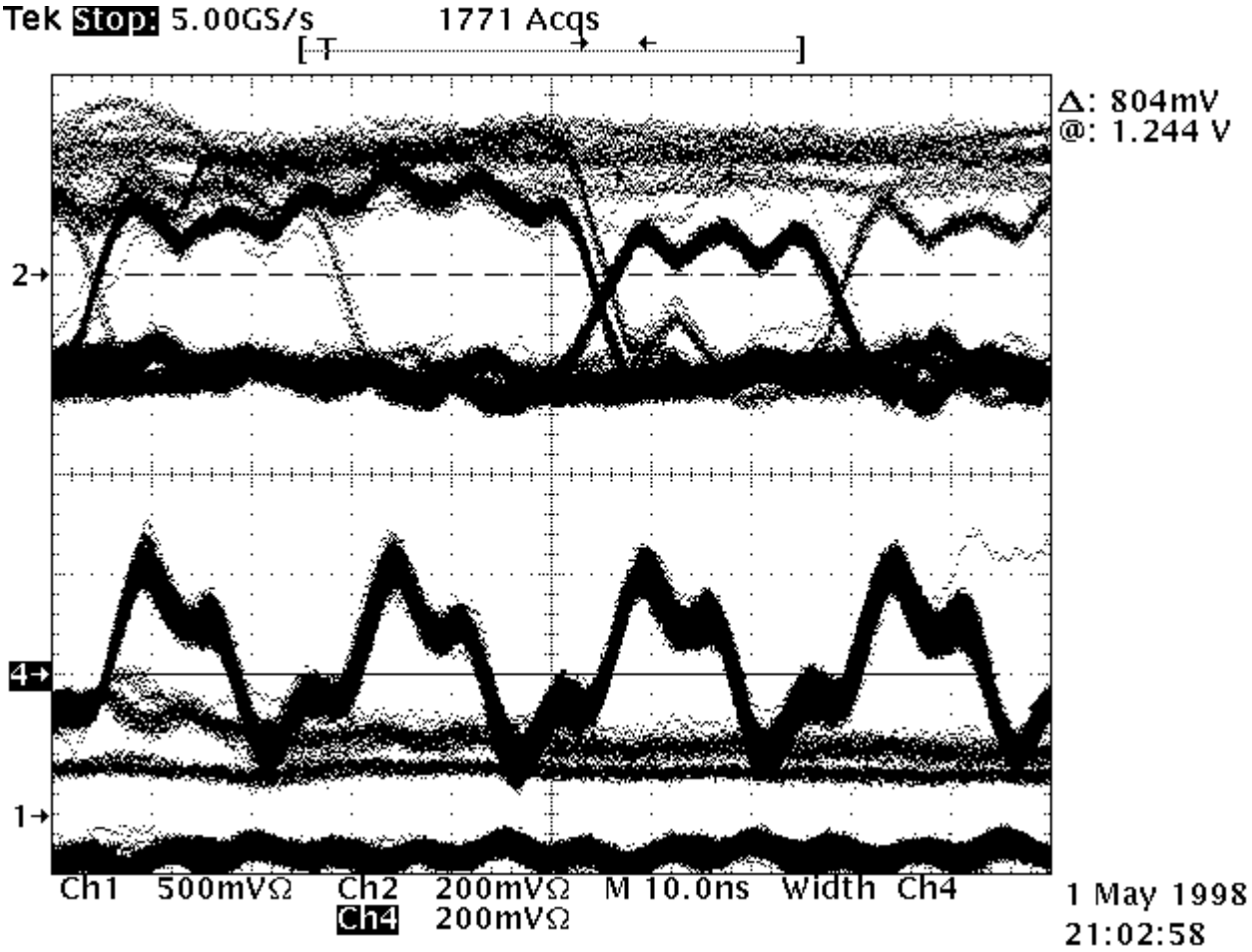
Premise:

Ultra3/Fast80 is assumed to use a dual-edge clocking scheme to double the transfer rate of the data. Because Ultra3/Fast80 data would be changing between each edge of the REQ/ACK pulses, it is important to analyze the setup and hold requirements to determine whether there is enough margin for the new technique to succeed. Observation of current Fast40 setup times can supply some information to help analyze future requirements. The data below are setup time measurements between data and REQ for a system having worst-case loading and cable length.

Setup:

- LVD, Fast40 data transfers, pseudo-random data pattern.
- 12 meter cable; active termination at both ends of the bus.
- 16 total devices on the cable: host at one end, 15 devices at the other end in a cluster with a spacing of 0.3 meters apart (cable capacitance: 45 pF/m).
- 14 devices in the cluster were dummy loads having the maximum LVD capacitance specified in SPI-2.
- The driving device was placed at the worst-case position for waveform distortion and attenuation as seen at the receiving end of the cable. This position was the mid-point of the cluster with 7 dummy loads on either side of the driver.
- Measurements were made with a Tektronix TDS684B scope using FET probes with < 2pF of capacitance. The probes were summed using the math function of the scope. (Differential probes were unavailable at the time of this test.)
- All waveforms were recorded with the scope probes attached to the receiving end of the cable.
- The scope's "infinite persistence" mode was used to create a type of "eye" pattern to allow many hundreds of samples to be overlaid to help find the worst-case edges. Like their analog scope counter-part, the extreme edges of the "fuzz" represent the worst-case boundaries.

1. Data vs. REQ, Vdiff Zero Crossing

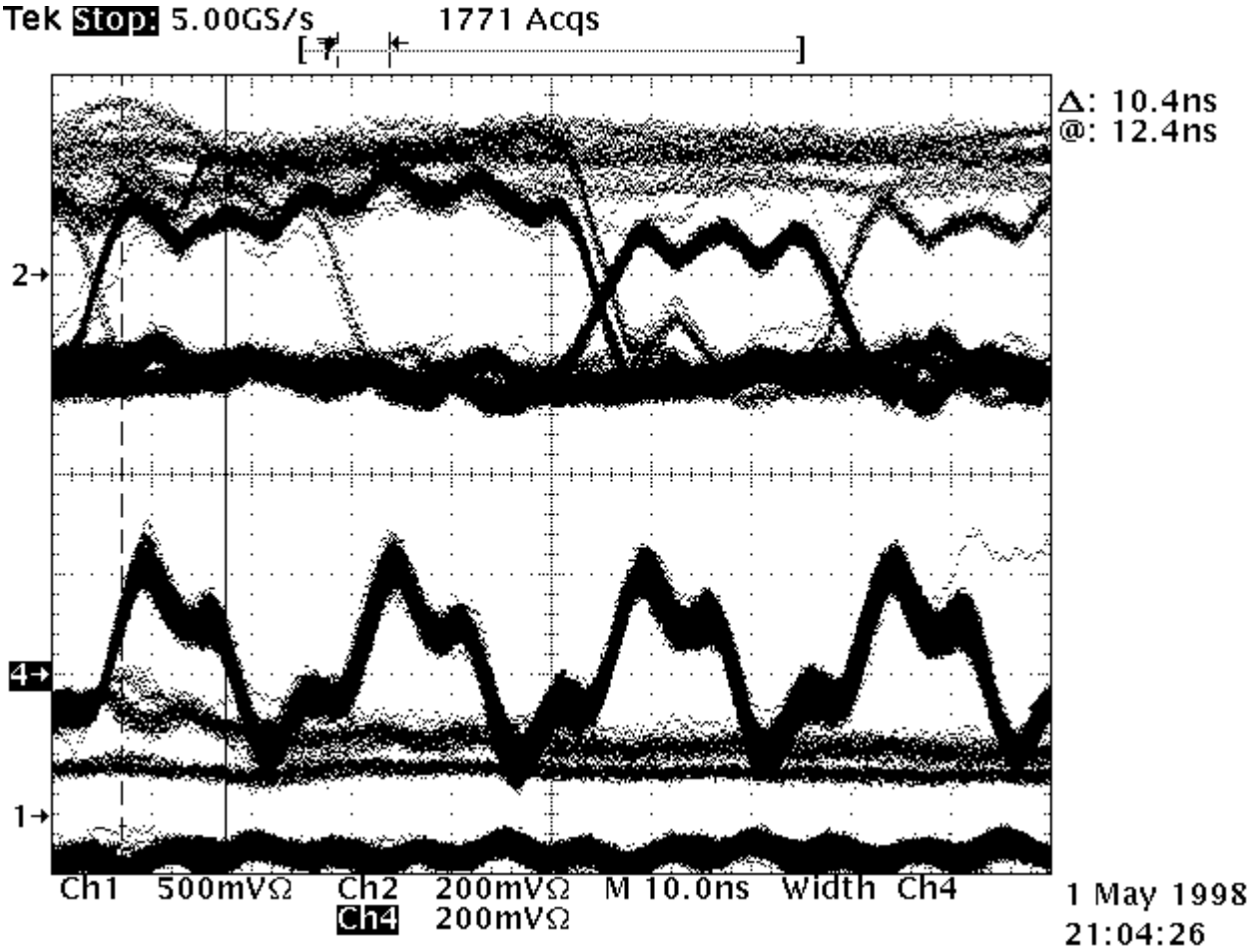


top trace: differential view of the worst-case data channel (using scope math function)
bottom trace: differential view of REQ (using scope math function)

REQ rate: 40MHz
Data: pseudo-random pattern during a Data-In phase

Horizontal cursors show the Vdiff zero lines of the waveforms.
(See follow-on slides for timing detail)

2. Data vs. REQ, Setup Data-to-REQ at Vdiff Zero Crossing, Sample #1

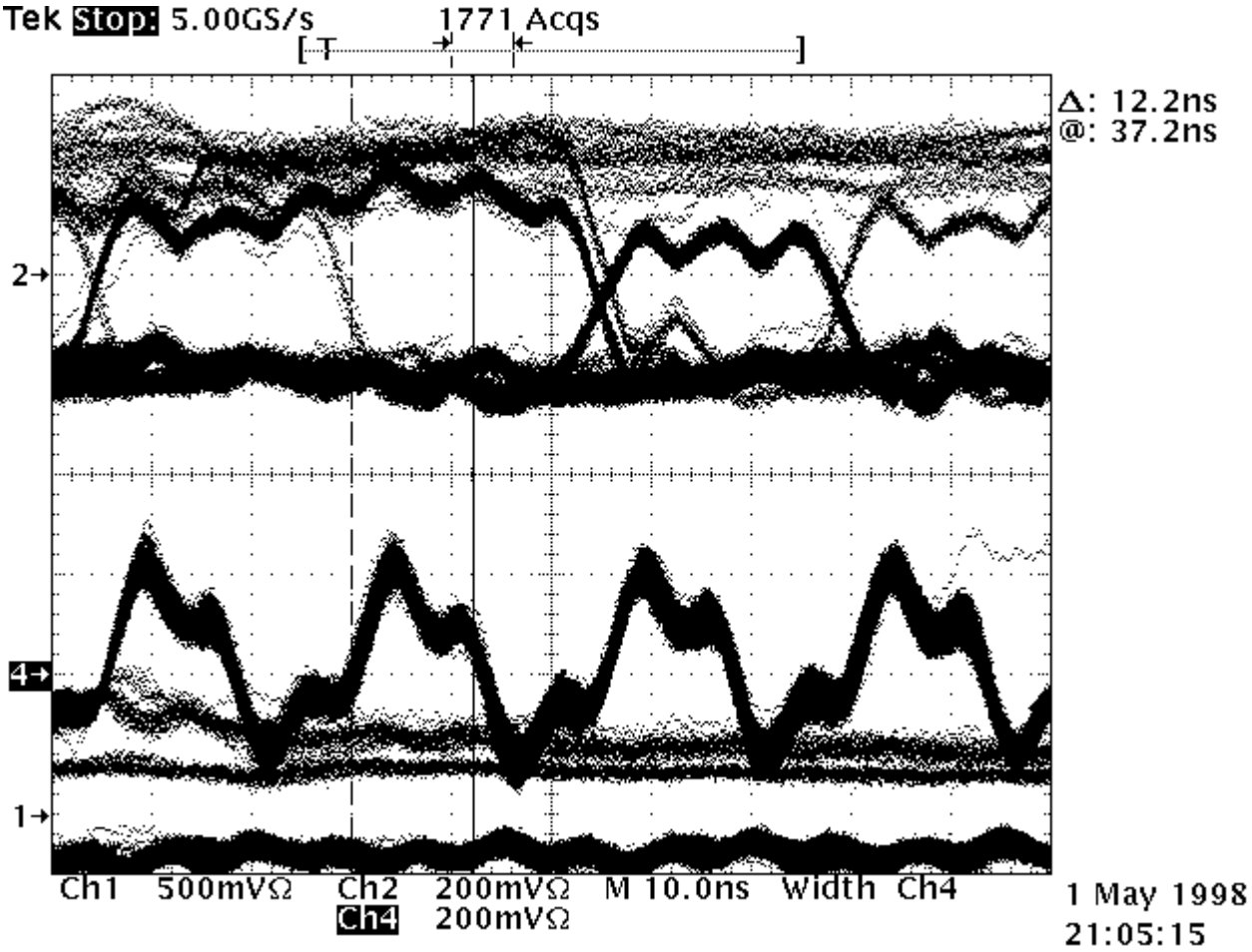


top trace: differential view of the worst-case data channel (using scope math function)
bottom trace: differential view of REQ (using scope math function)

REQ rate: 40MHz
Data: pseudo-random pattern during a Data-In phase

Vertical cursors show a minimum setup time between data and REQ.
Setup Time: 10.4 nsec.

3. Data vs. REQ, Setup Data-to-REQ at Vdiff Zero Crossing, Sample #2

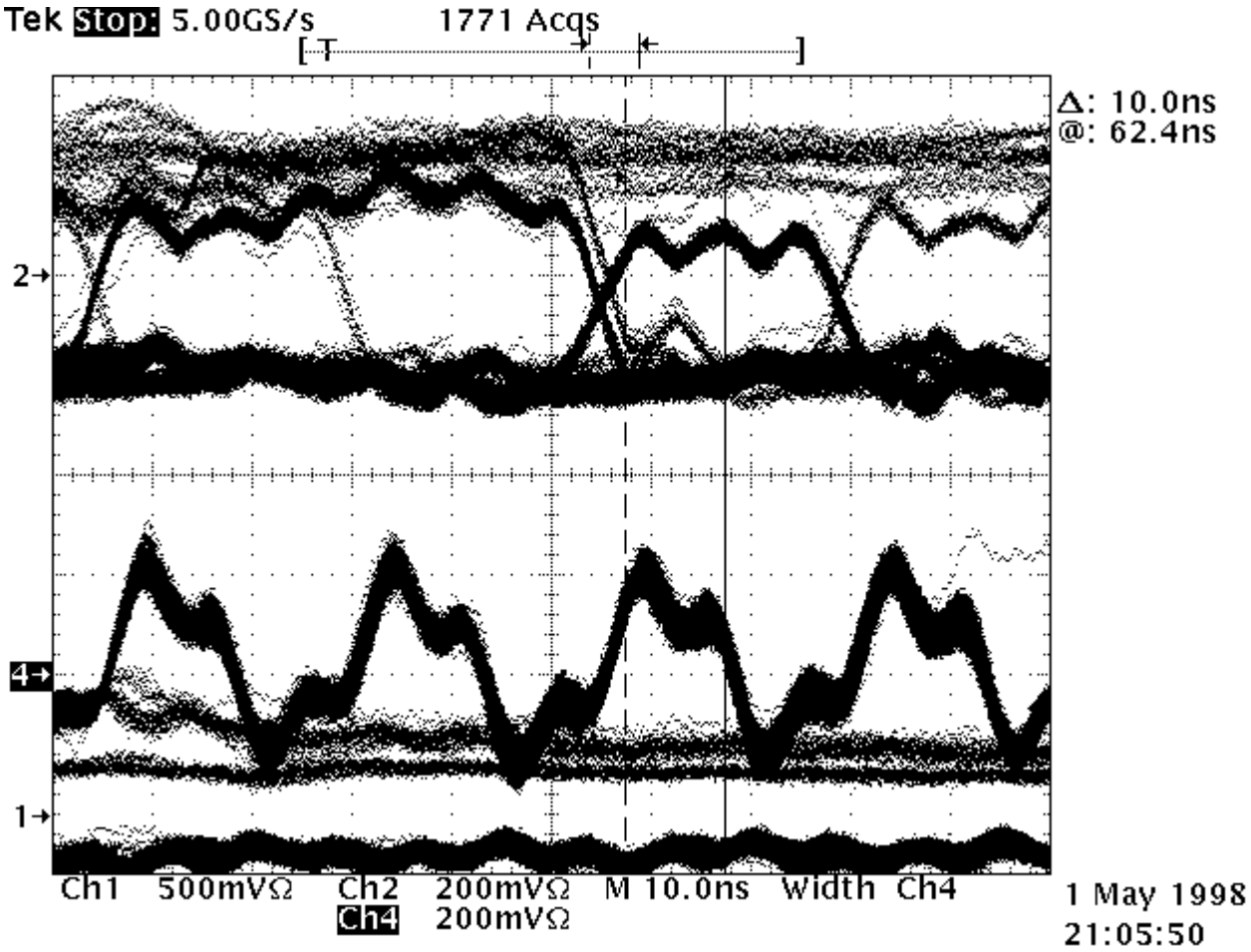


top trace: differential view of the worst-case data channel (using scope math function)
bottom trace: differential view of REQ (using scope math function)

REQ rate: 40MHz
Data: pseudo-random pattern during a Data-In phase

Vertical cursors show a minimum setup time between data and REQ.
Setup Time: 12.2 nsec.

4. Data vs. REQ, Setup Data-to-REQ at Vdiff Zero Crossing, Sample #3

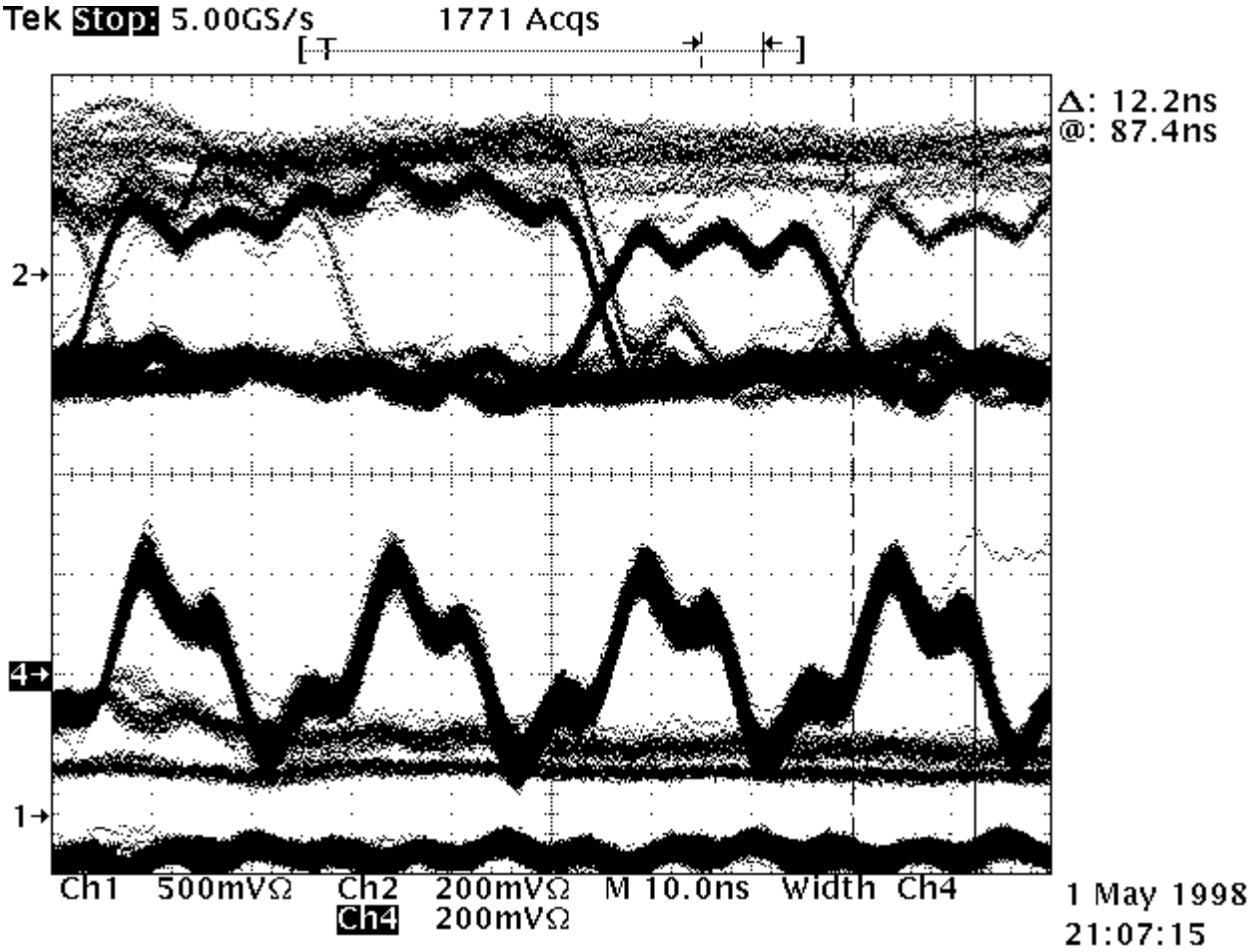


top trace: differential view of the worst-case data channel (using scope math function)
bottom trace: differential view of REQ (using scope math function)

REQ rate: 40MHz
Data: pseudo-random pattern during a Data-In phase

Vertical cursors show a minimum setup time between data and REQ.
Setup Time: 10.0 nsec.

5. Data vs. REQ, Setup Data-to-REQ at Vdiff Zero Crossing, Sample #4

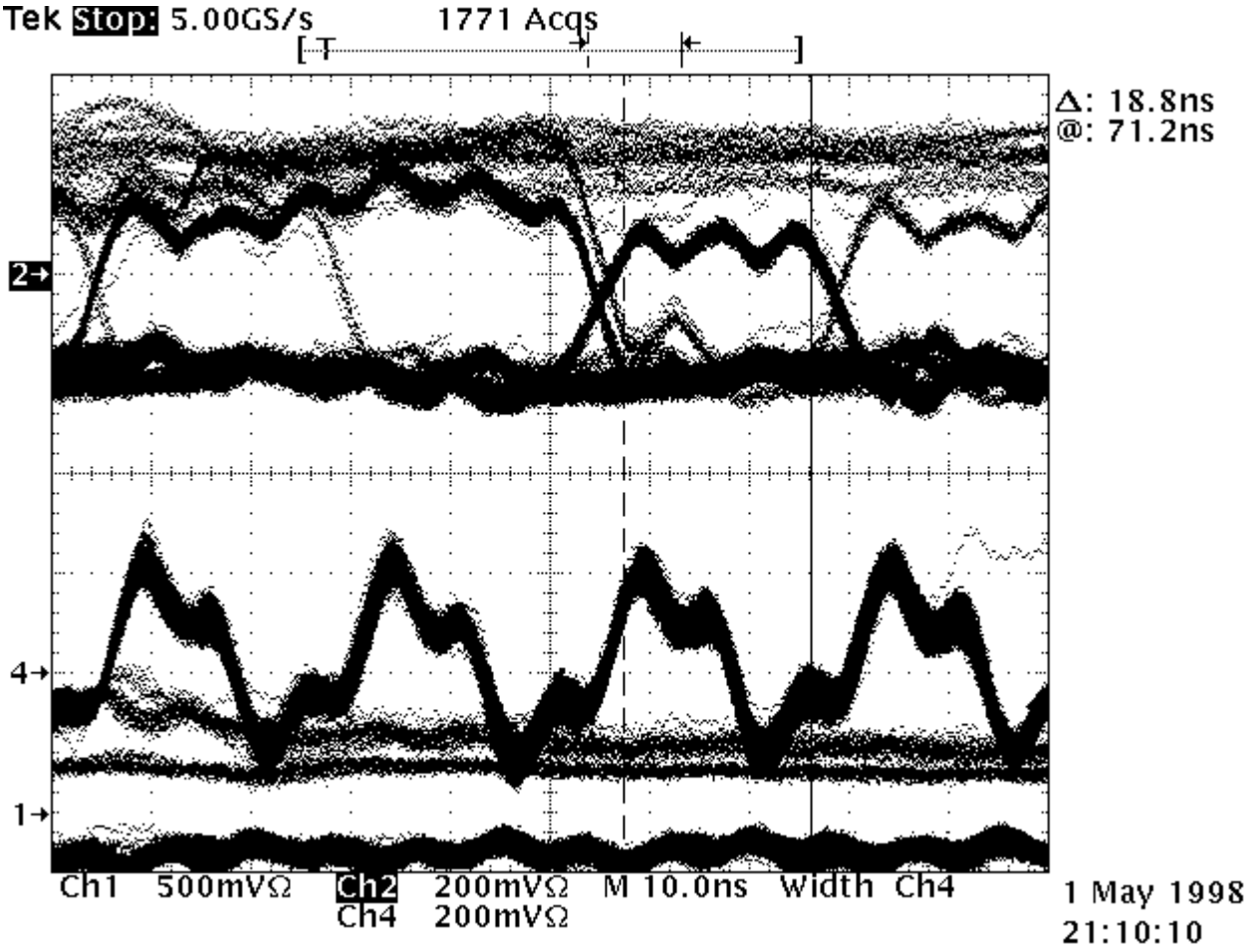


top trace: differential view of the worst-case data channel (using scope math function)
bottom trace: differential view of REQ (using scope math function)

REQ rate: 40MHz
Data: pseudo-random pattern during a Data-In phase

Vertical cursors show a minimum setup time between data and REQ.
Setup Time: 12.2 nsec.

6. Data Window – Minimum Width, at Vdiff Zero Crossing, Sample #1

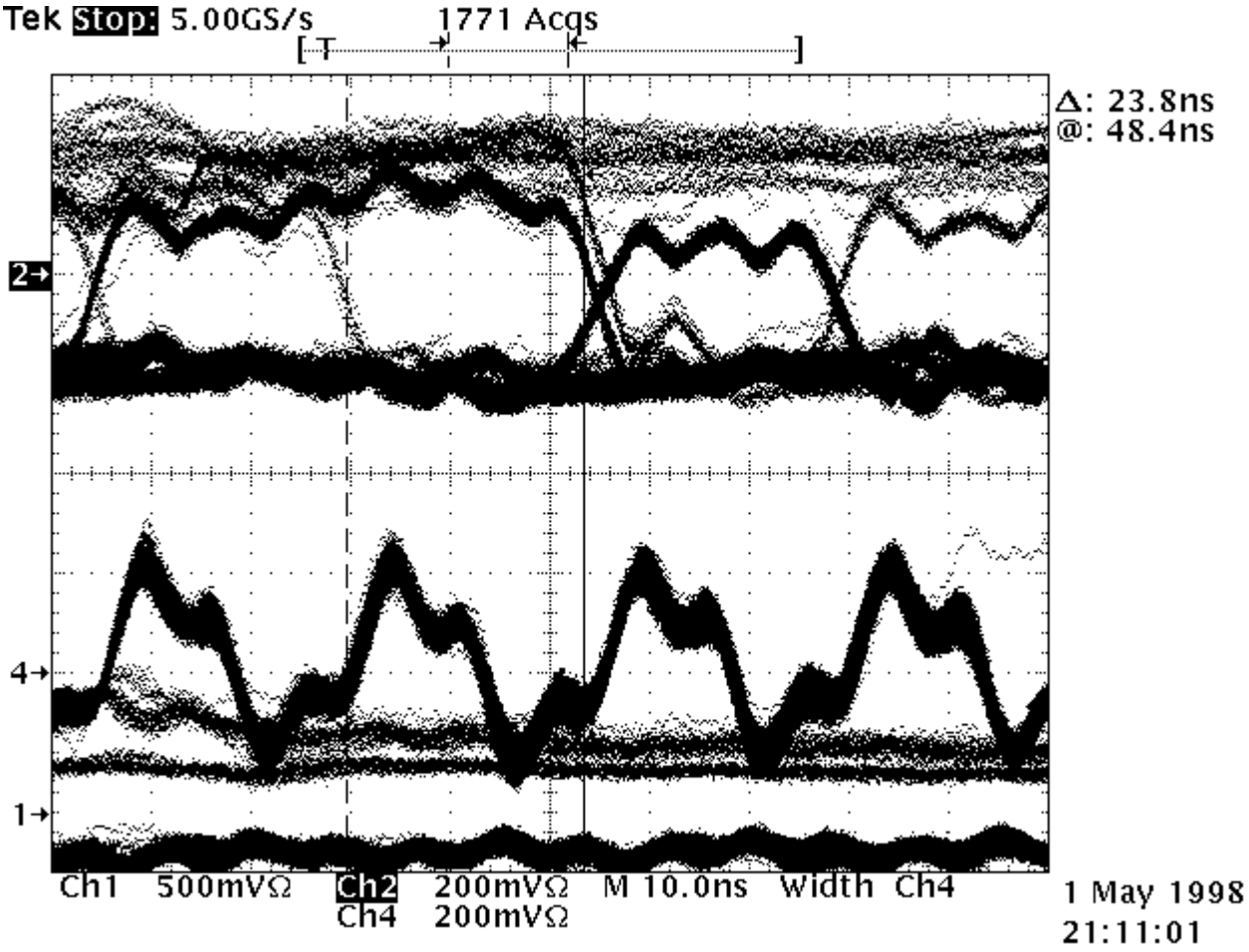


top trace: differential view of the worst-case data channel (using scope math function)
bottom trace: differential view of REQ (using scope math function)

REQ rate: 40MHz
Data: pseudo-random pattern during a Data-In phase

Vertical cursors show the width of a data valid window, minimum width.
Width of data opening = 18.8 nsec.

7. Data Window – Minimum Width, at Vdiff Zero Crossing, Sample #2

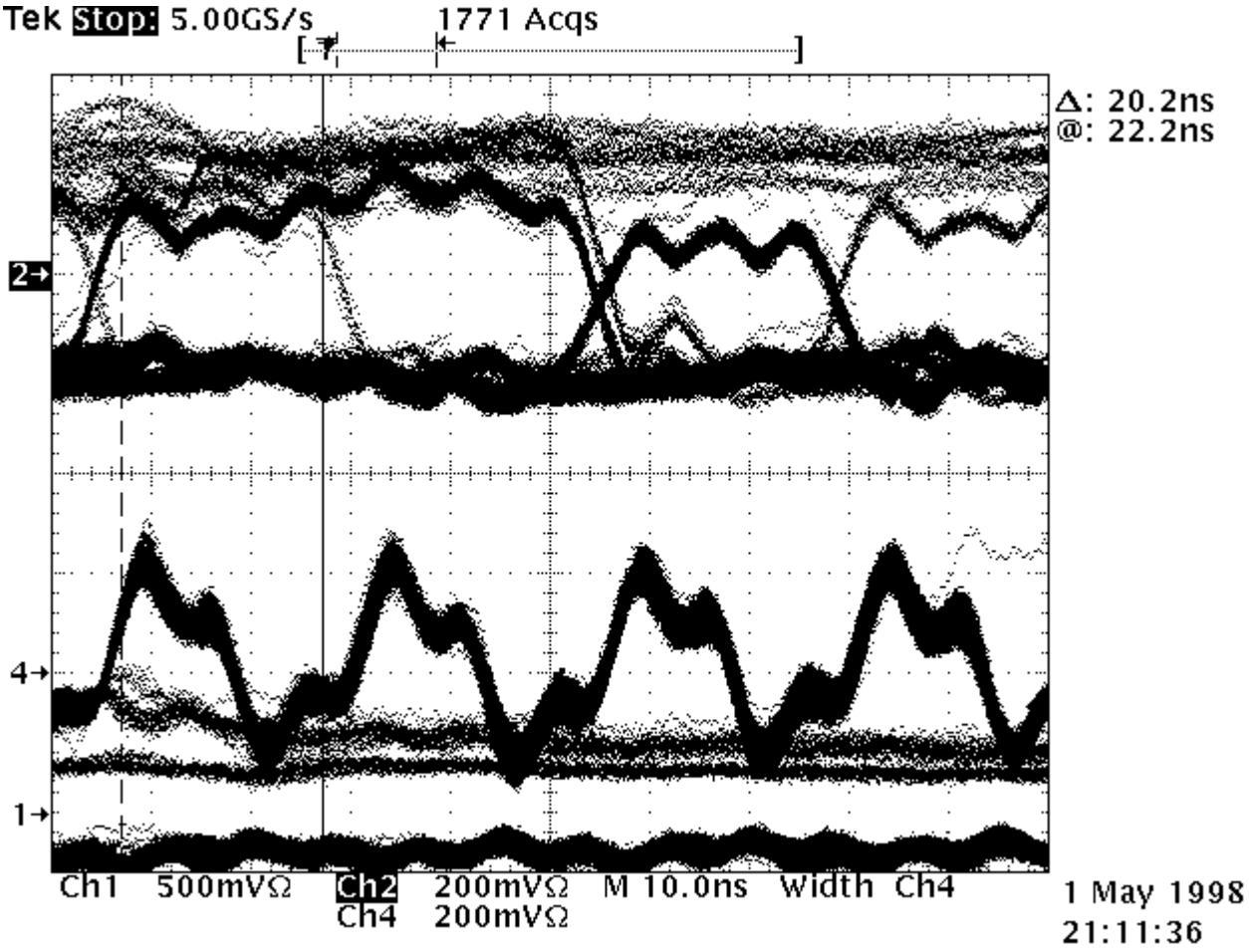


top trace: differential view of the worst-case data channel (using scope math function)
bottom trace: differential view of REQ (using scope math function)

REQ rate: 40MHz
Data: pseudo-random pattern during a Data-In phase

Vertical cursors show the width of a data valid window, minimum width.
Width of data opening = 23.8 nsec.

8. Data Window – Minimum Width, at Vdiff Zero Crossing, Sample #3



top trace: differential view of the worst-case data channel (using scope math function)
bottom trace: differential view of REQ (using scope math function)

REQ rate: 40MHz
Data: pseudo-random pattern during a Data-In phase

Vertical cursors show the width of a data valid window, minimum width.
Width of data opening = 20.2 nsec.

Conclusions:

Minimum Setup Time = 10.0 nsec
Minimum "data valid" window = 18.8 nsec

All variations in pulse width of data or REQ/ACK reduce the margins for setup and hold times. The nominal setup and hold time that can be achieved using the Ultra3, dual-edge clocking technique is 12.5 nsec.

The results of this test can only be used to analyze setup time because data is latched on only one edge of REQ or ACK. Assuming hold times will be similar to setup times in Ultra3 designs (disregarding any circuit sampling differences of either edge), these measurements can help define the requirements for system margins.

The minimum setup time measured above was 10.0 nsec. Dividing this by 2 equals 5.0 nsec, presumably the setup time at the Fast80 rate.

The minimum data window shown above is 18.8 nsec for a Fast40 data transfer. If this value is divided by two, the resulting 9.4 nsec could be the minimum opening for valid data at the Fast80 rate. Dividing 9.4 nsec by 2 equals 4.7 nsec. This could presumably be the resulting setup and hold time for Fast80, assuming the REQ/ACK pulse was centered in the data window.

These two values are assumed overlapping and therefore not additive. The smallest value above represent the least amount of margin. This test case used a worst-case cable length and loading, so shorter lengths and lighter loads would presumably increase the margins.

If 4.7 nsec is assumed to be a possibility for minimum Fast80 setup and hold time, then this leaves a margin for the rest of the system of 3.1 nsec (12.5 ns – 9.4 ns). Presently, Fast40 allows 5.0 nsec for the system portions of setup and hold margin. (4.5 ns for cable and 0.5 ns for board skew)

If the above data proves true, the system portion of the setup and hold margin will be reduced by 1.9 nsec. This means the existing systems may not work at the Ultra3 rate if an effort is not made to reduce the cable and board portion of this margin.