

**BR065R02 (submitted for p1394.1 committee vote):
Clock synchronization through bridges**

Dr. David V. James, Sony
3300 Zanker Road, MS SJ3C1
San Jose, CA 95134-4591
Phone: 408 955-6295
FAX: 408 955-4591
Email: dvj@alum.mit.edu

December 30, 1999

This contribution is one of several, presented for review and incorporation. An overall contribution, which provides an overall context for this and other contributions, is presented in BR047R10.

Clock synchronization throughout a bridged net is compounded by the need to constrain cycle-start time across bridges, while maintaining acceptable clock-reference precisions, and sometimes supporting more accurate clock references.

This proposal presents a simple technique for synchronization of bus bridges, by regenerating the clock at each bus bridge, compensating that regenerated output clock by the detected drift between input and output clocks.

Clock master information is collected and distributed during net cleanup to allow clock slaves to maintain more precise time synchronization for actual (as opposed to worst case) clock-master accuracies.

1.12 Clock synchronization

1.12.1 Clock distribution

The clock master is responsible for generating the net clock for all buses in the net. Each bus introduces phase-jitter errors while forwarding the clock to the net bus bridge, as illustrated in figure 1. To contain the cumulative jitter, bridges regenerate the cycle clock by passing their clock reference through DLC (drift locked clock) circuits.

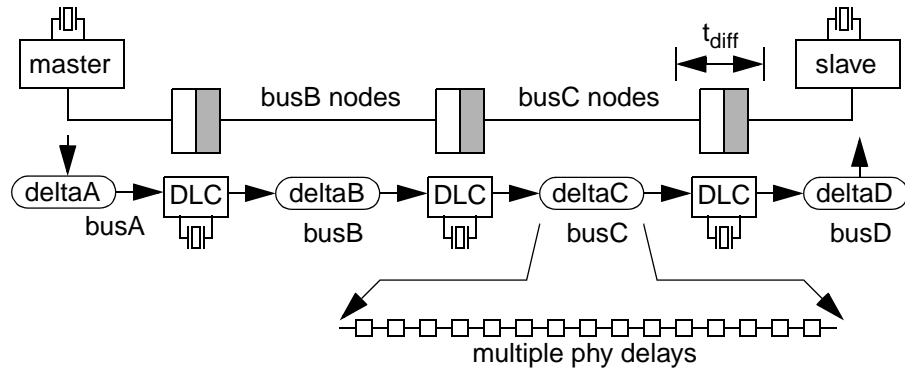


Figure 1—Clock distribution model

The deviation between adjacent portals on the most remote bus bridge depends on the cumulative jitter introduced by intermediate buses and bus bridges. On 1394a and 1394b buses, the maximum number of phy delays on a single bus is limited by the worst-case 3.5 μ s round-trip delay requirement used to detect acknowledge timeouts.

A minimal clock-master accuracy is specified and the clock-master accuracy is communicated to other portals during net cleanup operations. This determines the easily-achieved accuracy of clock-slave nodes. Goals of the clock-synchronization protocols include the following:

- 1) Constrained. The value of t_{diff} remains within the range supported by the bus bridge. (t_{diff} is the difference between cycle-time values observed & asserted by adjacent bus bridge portals)
- 2) Precise. The clock accuracy is maintained when passing through multiple bridges.
- 3) Responsive. The bus-bridge quickly adapts to clock-value changes caused by reassignment of the cycle-master node or writes to the cycle master's CYCLE_TIME register.

1.12.2 Bridge DLCs

A bridge’s drift locked clock (DLC) is responsible for reducing the cumulative bus-jitter errors. The clocking circuits in a bridge include a CYCLE_TIME_IN register and a CYCLE_TIME_OUT register, both of which are driven off of a common bus-bridge oscillator, as illustrated in figure 2. Other circuits are used to phase-lock the CYCLE_TIME_OUT register to the observed CYCLE_TIME_IN register, as discussed in the remainder of this subclause.

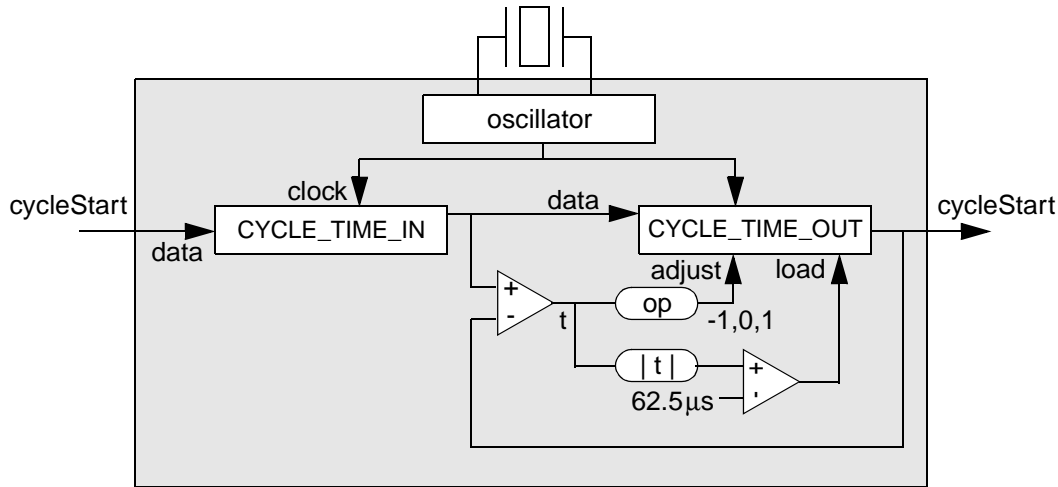


Figure 2—Higher level delay lock clock (DLC) circuitry

The CYCLE_TIME_OUT clock, like the CYCLE_TIME_IN clock, is normally incremented once each clock-tick cycle. An exception occurs at the end of each 8 kHz isochronous cycle, where the cycle timer is compensated by -1 , 0 , or $+1$ ticks. The intent is to synchronize the asserted CYCLE_TIME_OUT values to the observed CYCLE_TIME_IN values, while limiting the CYCLE_TIME_OUT jitter from one isochronous cycle to another.

NOTE—The period of the regenerated isochronous cycle can deviate from the nominal value by more than the specified 100PPM error rate for the isochronous clock (the worst case is more than a 400 PPM deviation). This is not viewed to be a concern, since a clock-slave can still accurately phase-lock to a clock-master (see E.2.2).

To maintain synchronization with the clock master, the CYCLE_TIME_IN clock is synchronized to the observed bus time, by transferring the contents of each cycle-start packet into the CYCLE_TIME_IN register.

A data path is provided between the CYCLE_TIME_IN out CYCLE_TIME_OUT registers. When the values in these two registers differs by more than $62.5 \mu\text{s}$ (half of an isochronous cycle), the load value is one and this data transfer occurs at the start of the next isochronous cycle.

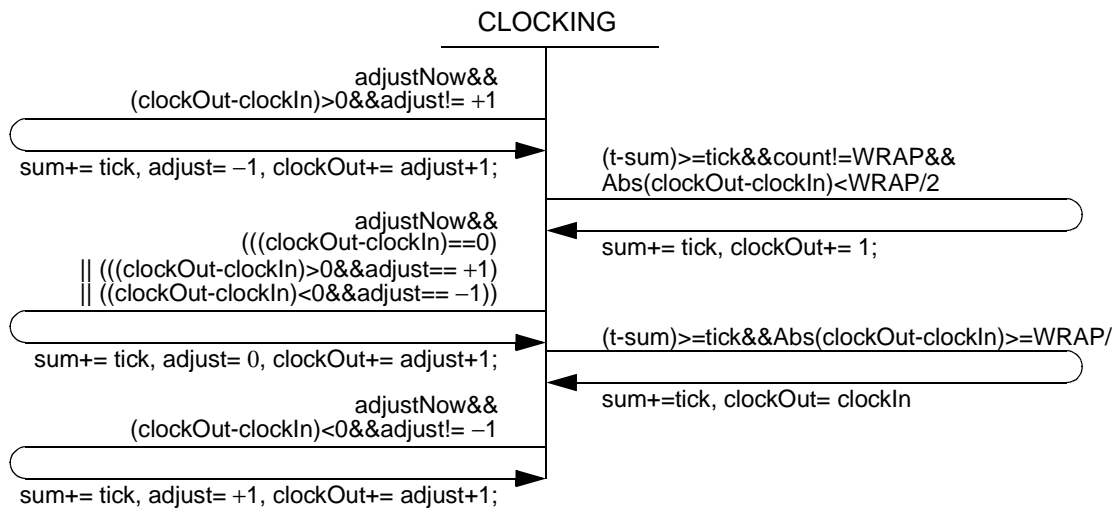
If the two registers differ by less than this 1/2 cycle threshold, then the difference between their values affects whether less, same, or more ticks are added at the start of the next isochronous cycle, as specified by table 1. The intent is to effect the desired tick-adjustment direction, which reducing the jitter by avoiding back-to-back adjustments of $\{-1, +1\}$ or $\{+1, -1\}$.

Table 1—Specified behavior of “op” circuit

past adjust value	current t	next adjust value
-1	negative	-1
	positive	0
0	negative	-1
	positive	+1
+1	negative	0
	positive	+1

1.12.3 Clock synchronization

An alternative way representing bus-bridge clock-synchronization behavior is a state machine that specifies the behavior of the CYCLE_TIME_OUT register. This *CLOCKING* state machines is specified in figure 3. In this state machine, *t* represents a monotonically increasing time value that is tested (but not affected by) the state machine.



Note:
`#define Abs(x) ((x)>0 ? (x):- (x))`
`#define adjustNow (t-sum)>=tick && count==WRAP && Abs(clockOut-clockIn)<WRAP/2)`

Figure 3—Clock distribution model

E.2.2 Clock-related topology constraints

NOTE—Although approximately correct, the equations in this subclause should be more accurate and correlated with a design model that illustrates the worst-case assumptions.

In extreme topologies, the cumulative jitter between the clock-master and clock-slave nodes may exceed the value supported by a bridge. As a result, false reloads may occur at lower levels of the bridge hierarchy, as illustrated in figure E.4. Reliable isochronous operations (i.e. avoiding false reloads) therefore limits the number of phy's and bridges between the clock master and clock slave nodes.

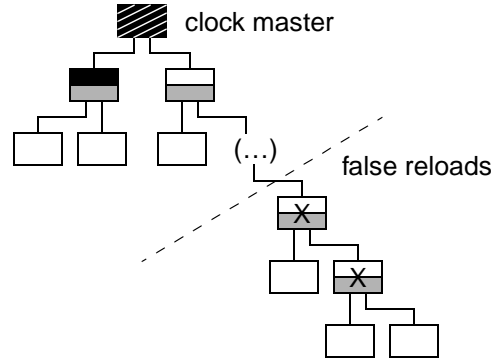


Figure E.4—Clock distribution model

The maximum number of “hops” between the clock master and the clock slave is specified by equation E.1. The cumulative jitter shall be less than 125 μs, where jitter on each “hop” is defined as the difference between the best and worst case delay values through that hop.

$$125 \leq \sum_{i=0}^N (Max(delta[i]) - Min(delta[i])) \tag{E.1}$$

NOTE—The cumulative jitter can be twice as large as the reload threshold, due to the approximate 1/2 ratio between the maximum and minimum slew-rate differences between nodes.

Assuming the worst case 1394 phy jitter is 320 ns and bridges stay within this value, this effectively limits the total number of hops to 390 hops. This is viewed to be beyond the expected diameter of any system, so no provisions are provided for detecting and/or reporting violations of this configuration-limit constraint.

Tick size = 1 / 24.576 MHz

E.2.3 Assumed design parameters

The 320 ns value was derived from the following information, extracted from p1394a, Draft 4.0, page #70:

$$jitterValue = .2 * (Jitter + 1) / BASE_RATE \mu s, \text{ where } Jitter \text{ is a 4-bit (maximum is 15) value} \tag{2}$$

The value of a nominal clock tick is specified as:

$$tick = 40.69 \text{ ns} = 1 / 24.576 \text{ MHz} \tag{3}$$

