Clock Setup and Hold
Slack Analysis Explained

In synchronous systems composed of register transfer logic (RTL), one of the main aspects of timing analysis is measuring the slack or timing margin, on the clock setup and clock hold times from one register to another. This Technical Note describes how this slack is measured in typical systems and provides a few simple formulas to use in analyzing these circuits involving Altera® FPGAs.

QUESTION: *What is the "standard method" that ASIC designer's use to measure clock setup and clock hold margin, or slack?*

ANSWER: Observe the simple register-to-register circuit below in Figure 1.

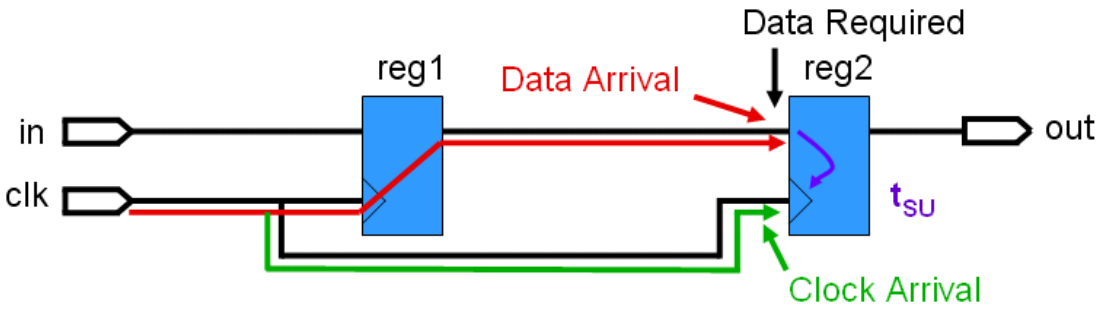

Figure 1

First let's analyze the *Clock Setup Slack*.

The *Data Arrival Time* is defined as the time from the clock input, through the launching register (reg1), to the latching register (reg2) data input,

$$\text{Data Arrival Time} = \text{Launch Edge} + \text{Longest } t_{CLK} + \mu t_{CO} + \text{Longest } t_D \quad [1]$$

where the Launch Edge (typically 0) is the relative time for the clock that "launches" the data from the

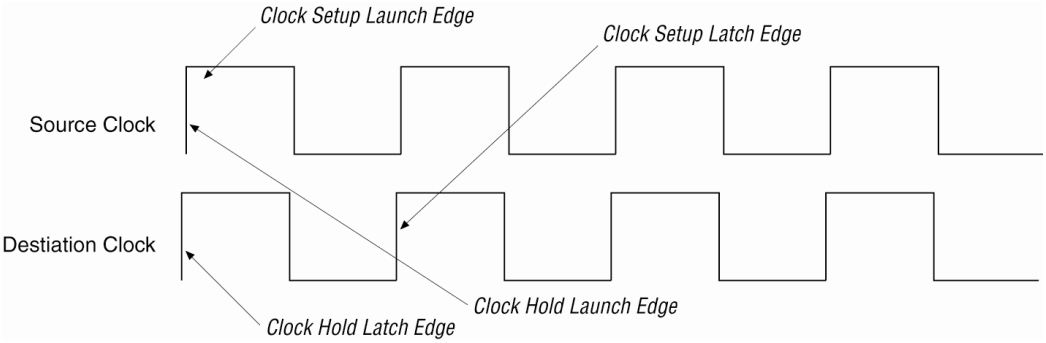

Figure 2

source register (reg1), t_{CLK} is the longest clock path from the clock input to the source register[1], μt_{CO} is the clock-to-output time of the source register[2], and t_D is the longest data delay from the source register to the destination register.

[1] This could include Late Clock Latency time.

[2] This is referred to as "micro" clock-to-output time when it is referring to the inherent clock-to-output delay of the internal register in the FPGA as opposed to the "real" clock-to-output delay which is always measured in reference to the input and output pins on the device.

The *Clock Arrival Time* is defined as the clock input to the latching register (reg2) clock input,

$$\text{Clock Arrival Time} = \text{Latch Edge} + \text{Shortest } t_{CLK}\text{'} \quad [2]$$

where the Latch Edge (typically the clock period)[3] is the relative time for the clock that "latches" the data at the destination register (reg2), and t_{CLK}' is the shortest clock path from the clock input to the destination register.

The *Data Required Time* is defined as the time the data can take to get to the destination register (reg2) just in time to meet the clock setup time at the destination register,

$$\text{Data Required Time} = \text{Clock Arrival Time} - \mu t_{SU} - \text{Clock Setup Uncertainty}^4 \quad [3]$$

where the μt_{SU} is the clock setup time of the destination register[5].

Now with these terms defined, we can define the *Clock Setup Slack* as the difference between the time available to reach the destination register and the actual time to get there.

$$\text{Clock Setup Slack} = \text{Data Required Time} - \text{Data Arrival Time} \quad [4]$$

Let's analyze the *Clock Hold Slack* next.

The *Data Arrival Time* is similar to the Clock Setup analysis case.

$$\text{Data Arrival Time} = \text{Launch Edge} + \text{Shortest } t_{CLK} + \min \mu t_{CO} + \text{Shortest } t_D \quad [5]$$

In this case we are using the shortest clock path and the shortest data delay.

The *Clock Arrival Time* is also similar to the Clock Setup analysis case.

$$\text{Clock Arrival Time} = \text{Latch Edge} + \text{Longest } t_{CLK}\text{'} \quad [6]$$

In this case we are using the longest clock path to the destination register[6].

The *Data Required Time* in this case is defined as the time the data must not be changed at the destination register (reg2) just in time to meet the clock hold time at the destination register.

$$\text{Data Required Time} = \text{Clock Arrival Time} + \mu t_H + \text{Clock Hold Uncertainty}^7 \quad [7]$$

where μt_H is the clock hold time of the destination register[8].

---

[3] By convention, clock *offset* (intentionally added clock delay), e.g. from a PLL in the clock path, will affect the Launch and Latch edges, and as such affect the Setup and Hold Relationships. Clock *uncertainty* will also have the same effect. Clock *skew* and clock *latency* will not affect the Launch and Latch edges, and therefore will not affect Setup and Hold Relationships. Clock skew (and latency) is handled separately in the slack equations.

[4] Clock Uncertainty can be used to model jitter, skew and/or create a guard band for the analysis.

[5] This is referred to as "micro" clock setup time when it is referring to the inherent setup requirement of the internal register in the FPGA as opposed to the "real" setup requirement which is always measured in reference to the input pins on the device.

[6] This can include the Early Clock Latency time.

[7] Clock Uncertainty can be used to model jitter, skew and/or create a guard band for the analysis.

[8] This is referred to as "micro" clock hold time when it is referring to the inherent hold requirement of the internal register in the FPGA as opposed to the "real" hold requirement which is always measured in reference to the input pins on the device.

Now with these terms defined, we can define the *Clock Hold Slack* as the difference between the actual time data is held at the destination register and the time it needs to be held the there.

$$\text{Clock Hold Slack} = \text{Data Arrival Time} - \text{Data Required Time} \quad [8]$$

Note that this is the reverse of the *Clock Setup Slack* calculation.

QUESTION: *What is the method that Quartus® II timing analysis (TAN) uses to measure clock setup and clock hold margin, or slack?*

ANSWER: Observe the simple register to register circuit below in Figure 3.

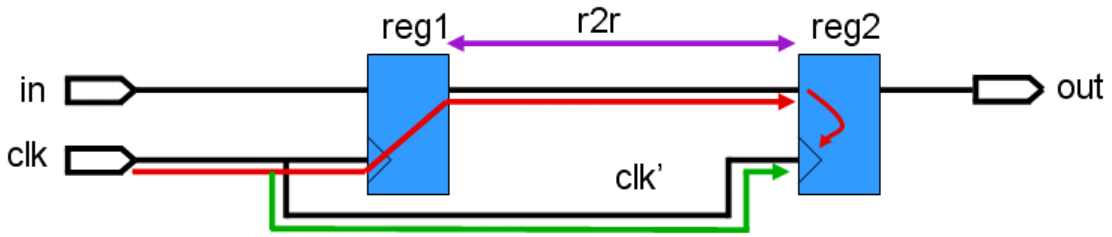

Figure 3

Quartus II TAN uses a different method to calculate *Clock Setup Slack* and *Clock Hold Slack*.

Let's first look at *Clock Setup Slack*.

The *Clock Setup Relationship (SR)* between the source and destination register is the time between the setup launching clock edge and the setup latching clock edge (see Figure 2).

$$\text{SR} = \text{Setup Latch Edge} - \text{Setup Launch Edge} - \text{Clock Setup Uncertainty} \quad [9]$$

The Setup Launch Edge (typically 0) and Setup Latch Edge (typically the clock period) were defined previously.

The largest *Register-to-Register (r2r) Requirement* is the time required for the data to get to the destination register to meet the clock setup time at the destination register,

$$\text{Largest r2r Required} = \text{SR} + \min t_{CS} - \mu t_{CO} - \mu t_{SU} \quad [10]$$

where the minimum t_{CS} is defined as clock skew, and μt_{CO} and μt_{SU} were previously defined. The minimum clock skew is defined as the shortest clock delay to the destination register minus the longest clock delay to the source register.

$$\min t_{CS} = \text{Shortest clk'} - \text{Longest clk} \quad [11]$$

The longest *Register-to-Register (r2r) Delay* is the time it takes for the data to get from the source register to the destination register taking the longest path.

$$\text{Longest r2r Delay} = t_D \quad [12]$$

Now with these terms defined, we can define the *Clock Setup Slack* as the difference between the time available to reach the destination register and the actual time to get there.

$$\text{Clock Setup Slack} = \text{Largest r2r Required} - \text{Longest r2r Delay} \quad [13]$$

Now, substituting equations 10 and 12 into 13, we get;

$$\boxed{\textbf{Clock Setup Slack} = \textbf{SR} + \textbf{min } t_{CS} - \mu t_{CO} - \mu t_{SU} - t_D} \quad [14]$$

Let's analyze the *Clock Hold Slack* next.

The *Clock Hold Relationship (HR)* between the source and destination register is the time between the hold launching clock edge and the hold latching clock edge (see Figure 2).

$$\text{HR} = \text{Hold Latch Edge} - \text{Hold Launch Edge} + \text{Clock Hold Uncertainty} \quad [15]$$

The Hold Launch Edge (typically 0) and Hold Latch Edge (typically 0) were defined previously.

The smallest *Register-to-Register (r2r) Requirement* is the time required that the data be held at the destination register to meet the clock hold time at the destination register,

$$\text{Smallest r2r Required} = \text{HR} + \max t_{CS} - \min \mu t_{CO} + \mu t_H \quad [16]$$

where the max t_{CS} is defined as clock skew, and min μt_{CO} is the minimum clock-to-output time of the source register[9], and μt_H was defined previously. The max clock skew is defined as the longest clock delay to the destination register minus the shortest clock delay to the source register.

$$\max t_{CS} = \text{Longest clk'} - \text{Shortest clk} \quad [17]$$

The shortest *Register-to-Register (r2r) Delay* is the time it takes for the data to get from the source register to the destination register taking the shortest path.

$$\text{Shortest r2r Delay} = t_D \quad [18]$$

Now with these terms defined, we can define the *Clock Hold Slack* as the difference between the quickest time the data can be taken away from the destination register and the time required to keep it there.

$$\text{Clock Hold Slack} = \text{Shortest r2r Delay} - \text{Smallest r2r Required} \quad [19]$$

Now, substituting equations 16 and 18 into 19, we get;

$$\boxed{\textbf{Clock Hold Slack} = t_D - \textbf{HR} - \textbf{max } t_{CS} + \textbf{min } \mu t_{CO} - \mu t_H} \quad [20]$$

QUESTION: *Do the two methods yield the same results, and how are they related to each other?*

ANSWER: Yes, the two methods yield the exact same result. Look at Figure 4 to see how the two methods are related.

[9] Quartus II TAN does not distinguish between min μt_{CO} and μt_{CO} for internal registers. The same very small value is used for both.

Standard View

$$\text{data arrival} = \text{launch} + \text{clk} + \mu t_{CO} + \text{data}$$

$$\text{clock arrival} = \text{latch} + \text{clk'}$$

$$\text{data required} = \text{clock arrival} - \mu t_{SU}$$

$$\text{slack} = \text{data required} - \text{data arrival}$$

Quartus II's View

$$\text{setup relationship} = \text{latch} - \text{launch}$$

$$\text{clock skew} = \text{clk'} - \text{clk}$$

$$\text{r2r required} = \text{setup relationship} + \text{clock skew} - \mu t_{CO} - \mu t_{SU}$$

$$\text{slack} = \text{r2r required} - \text{r2r delay}_{\text{data}}$$

Figure 4

Starting with the Standard View equations for slack above and substituting;

$$\text{slack} = \text{data required} - \text{data arrival} \quad [21]$$

$$\text{slack} = ((\text{latch} + \text{clk'}) - \mu t_{SU}) - (\text{launch} + \text{clk} + \mu t_{CO} + \text{data}) \quad [22]$$

Now, rearranging we get;

$$\text{slack} = (\text{latch} - \text{launch}) + (\text{clk'} - \text{clk}) - \mu t_{CO} - \mu t_{SU} - \text{data} \quad [23]$$

Starting with the Quartus II's View equations for slack above and substituting;

$$\text{slack} = \text{r2r required} - \text{r2r delay} \quad [24]$$

$$\text{slack} = (\text{latch} - \text{launch}) + (\text{clk'} - \text{clk}) - \mu t_{CO} - \mu t_{SU} - \text{data} \quad [25]$$

Note that equation 23 and equation 25 are identical.

QUESTION: *How are the equations for input and output delays derived and when they are used to analyze I/O timing, how are the equations for slack affected?*

ANSWER: The equations for *Clock Setup Slack* and *Clock Hold Slack* are used for analyzing register-to-register paths. In order to use these equations to analyze I/O timing, we have to take into consideration the registers external to the FPGA and the external paths connecting to them.

First let's look at the input side.

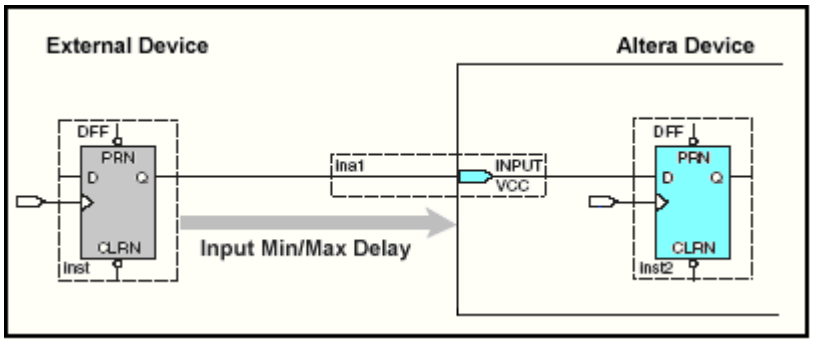

Figure 5

The slowest external path to the data inputs is called the *Input Maximum Delay (IMD)*. The fastest external path to the data inputs is called the *Input Minimum Delay (ImD)*.

Now let's look at the output side.

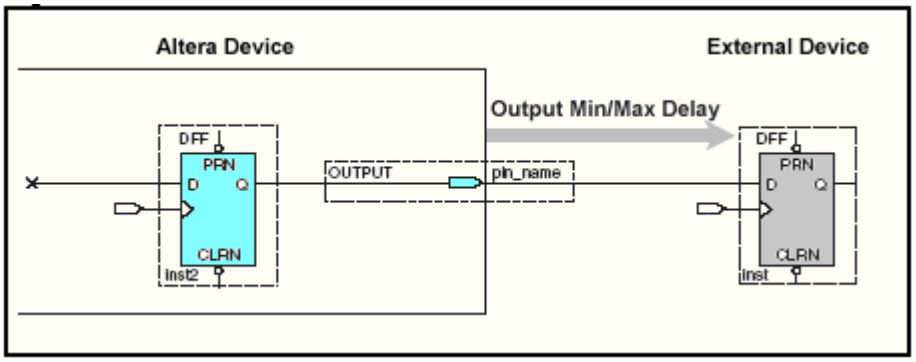

Figure 6

The slowest external path from the data outputs is called the *Output Maximum Delay (OMD)*. The fastest external path from the data outputs is called the *Output Minimum Delay (OmD)*.

Recall equation 14 for *Clock Setup Slack*.

$$\text{Clock Setup Slack} = SR + \min t_{CS} - \mu t_{CO} - \mu t_{SU} - t_D \qquad [26]$$

We need to modify this for the cases where the registers are external to the FPGA. The SR remains the same. On the input side, the min t_{CS} is the clock skew measured to the input pin for the clock instead of all the way to the internal register[10]. The μt_{CO} is replaced by the t_{CO} of the external register feeding the data input. The μt_{SU} is replaced by the t_{SU} of the data input relative to the clock input as reported by Quartus II TAN. The data delay is now the worst-case trace delay from the external register to the data input pin on the FPGA. The new *Clock Setup Slack* equation for the input case is now:

$$\text{Clock Setup Slack} = SR + \min t_{CS\,(EXT)} - t_{CO\,(EXT)} - t_{SU} - \max t_{D\,(EXT)} \qquad [27]$$

Regrouping the equation putting the external components together yields:

[10] The clock delay from the input pin to the input register and the data delay from the input pin to the input register are accounted for in the t_{SU} calculation.

$$\text{Clock Setup Slack} = SR - t_{SU} - (t_{CO\,(EXT)} + \max t_{D\,(EXT)} - \min t_{CS\,(EXT)}) \quad [28]$$

Now we can say that:

$$\mathbf{IMD = t_{CO\,(EXT)} + \max t_{D\,(EXT)} - \min t_{CS\,(EXT)}} \quad [29]$$

Substituting this back into equation 28 gives us the *Clock Setup Slack* for the input case.

$$\text{Clock Setup Slack} = SR - t_{SU} - IMD \quad [30]$$

Considering the output side, the largest t_{CS} is the clock skew measured again to the input pin for the clock instead of all the way to the internal register[11]. The μt_{CO} is replaced by the t_{CO} of the data output relative to the clock input as reported by Quartus II TAN. The μt_{SU} is replaced by the t_{SU} of the external register fed by the data output. The data delay is now the worst-case trace delay from the data output pin on the FPGA to the external register. The new *Clock Setup Slack* equation for the output case is now:

$$\text{Clock Setup Slack} = SR + \min t_{CS\,(EXT)} - t_{CO} - t_{SU\,(EXT)} - \max t_{D\,(EXT)} \quad [31]$$

Regrouping the equation putting the external components together yields:

$$\text{Clock Setup Slack} = SR - t_{CO} - (\max t_{D\,(EXT)} + t_{SU\,(EXT)} - \min t_{CS\,(EXT)}) \quad [32]$$

Now we can say that:

$$\mathbf{OMD = \max t_{D\,(EXT)} + t_{SU\,(EXT)} - \min t_{CS\,(EXT)}} \quad [33]$$

Substituting this back into equation 32 gives us the *Clock Setup Slack* for the input case.

$$\text{Clock Setup Slack} = SR - t_{CO} - OMD \quad [34]$$

Now, recall equation 20 for *Clock Hold Slack*.

$$\text{Clock Hold Slack} = t_D - HR - \max t_{CS} + \min \mu t_{CO} - \mu t_H \quad [35]$$

We need to modify this for the cases where the registers are external to the FPGA. The HR remains the same. On the input side, the smallest t_{CS} is the clock skew measured to the input pin for the clock instead of all the way to the internal register as with the case for the clock setup slack. The min μt_{CO} is replaced by the min t_{CO} of the external register feeding the data input. The μt_H is replaced by the t_H of the data input relative to the clock input as reported by Quartus II TAN. The data delay is now the best-case trace delay from the external register to the data input pin on the FPGA. The new *Clock Hold Slack* equation for the input case is now:

$$\text{Clock Hold Slack} = \min t_{D\,(EXT)} - HR - \max t_{CS\,(EXT)} + \min t_{CO\,(EXT)} - t_H \quad [36]$$

[11] The clock delay from the input pin to the output register and the data delay from the output register to the output pin are accounted for in the t_{CO} calculation.

Regrouping the equation putting the external components together yields:

$$\text{Clock Hold Slack} = (\min t_{CO\,(EXT)} + \min t_{D\,(EXT)} - \max t_{CS\,(EXT)}) - HR - t_H \quad [37]$$

Now we can say that:

$$\mathbf{ImD = \min t_{CO\,(EXT)} + \min t_{D\,(EXT)} - \max t_{CS\,(EXT)}} \quad [38]$$

Substituting this back into equation 33 gives us the *Clock Hold Slack* for the input case.

$$\text{Clock Hold Slack} = ImD - HR - t_H \quad [39]$$

Considering the output side, the max t_{CS} is the clock skew measured again to the input pin for the clock instead of all the way to the internal register. The min μt_{CO} is replaced by the min t_{CO} of the data output relative to the clock input as reported by Quartus II TAN. The μt_H is replaced by the t_H of the external register fed by the data output. The data delay is now the best-case trace delay from the data output pin on the FPGA to the external register. The new *Clock Hold Slack* equation for the output case is now:

$$\text{Clock Hold Slack} = \min t_{D\,(EXT)} - HR - \max t_{CS\,(EXT)} + \min t_{CO} - t_{H\,(EXT)} \quad [40]$$

Regrouping the equation putting the external components together yields:

$$\text{Clock Hold Slack} = (\min t_{D\,(EXT)} - t_{H\,(EXT)} - \max t_{CS\,(EXT)}) - HR + \min t_{CO} \quad [41]$$

Now we can say that:

$$\mathbf{OmD = \min t_{D\,(EXT)} - t_{H\,(EXT)} - \max t_{CS\,(EXT)}} \quad [42]$$

Substituting this back into equation 36 gives us the *Clock Hold Slack* for the output case.

$$\text{Clock Hold Slack} = OmD - HR + \min t_{CO} \quad [43]$$

QUESTION: *How are IMD, ImD, OMD, and OmD related to t_{SU}, t_H, t_{CO}, and min t_{CO}?*

ANSWER: In order to convert these two types of I/O constraints, let's look at the special case where *Clock Setup Slack* and *Clock Hold Slack* are equal to 0. When this occurs, the external delays are at their extremes to meet the internal requirements.

For *Input Maximum Delay*, starting with equation 30;

$$\text{Clock Setup Slack} \xrightarrow{0} = SR - t_{SU} - IMD \quad [44]$$

$$IMD = SR - t_{SU} \quad [45]$$

For the general case where SR = clock period (T)[12];

$$\boxed{\mathbf{IMD = T - t_{SU}}} \quad [46]$$

For *Input Minimum Delay*, starting with equation 39;

$$\text{Clock Hold Slack} \xrightarrow{0} = ImD - HR - t_H \quad [47]$$

$$ImD = HR + t_H \quad [48]$$

For the general case where HR = 0[13];

$$\boxed{\mathbf{ImD = t_H}} \quad [49]$$

For the *Output Maximum Delay*, starting with equation 34;

$$\text{Clock Setup Slack} \xrightarrow{0} = SR - t_{CO} - OMD \quad [50]$$

$$OMD = SR - t_{CO} \quad [51]$$

For the general case where SR = T;

$$\boxed{\mathbf{OMD = T - t_{CO}}} \quad [52]$$

For the *Output Minimum Delay*, starting with equation 43;

$$\text{Clock Hold Slack} \xrightarrow{0} = OmD - HR + \min t_{CO} \quad [53]$$

$$OmD = HR - \min t_{CO} \quad [54]$$

For the general case where HR = 0;

[12] Exceptions to this are things that affect the Latch and Launch edges, e.g. multicycle settings, clock offset, DDR, etc. Also, this assumes that the reference clock for the Input and Output Delay settings is the input clock that is used as the reference for t_{SU}, t_{CO}, etc.
[13] (See Footnote 10 above).

$$\boxed{\mathbf{OmD = -\min t_{CO}}} \qquad [55]$$

Here is a summary for converting between the two types of I/O constraints:

$\mathbf{IMD = T - t_{SU}}$	$\mathbf{t_{SU} = T - IMD}$
$\mathbf{ImD = t_H}$	$\mathbf{t_H = ImD}$
$\mathbf{OMD = T - t_{CO}}$	$\mathbf{t_{CO} = T - OMD}$
$\mathbf{OmD = -\min t_{CO}}$	$\mathbf{\min t_{CO} = -OmD}$

and a summary of the I/O Delay equations:

$$\mathbf{IMD = t_{CO\,(EXT)} + \max t_{D\,(EXT)} - \min t_{CS\,(EXT)}}$$

$$\mathbf{ImD = \min t_{CO\,(EXT)} + \min t_{D\,(EXT)} - \max t_{CS\,(EXT)}}$$

$$\mathbf{OMD = \max t_{D\,(EXT)} + t_{SU\,(EXT)} - \min t_{CS\,(EXT)}}$$

$$\mathbf{OmD = \min t_{D\,(EXT)} - t_{H\,(EXT)} - \max t_{CS\,(EXT)}}$$

where

$$\mathbf{\min t_{CS\,(EXT)} = \text{Shortest Destination } t_{CLK} - \text{Longest Source } t_{CLK}}$$
$$\mathbf{\max t_{CS\,(EXT)} = \text{Longest Destination } t_{CLK} - \text{Shortest Source } t_{CLK}}$$

Note here that Largest and Smallest are relative terms based on the definitions in register-to-register timing, and that the Largest t_{CS} is always a lower value than the Smallest t_{CS}.