Asynchronous Research Center
Portland State University

Subject: **Distance Constraint Graph: A Graphical Representation for 6-4 GasP showing how Relative Timings constrain the Module Distances**
Date: **September 1, 2009**
From: **Swetha Mettala Gilla**
ARC#: **2009-smg01**

References:

[1] Prasad Joshi, Peter A; Beerel, Marly Roncken and Ivan Sutherland
Timing Verification of GasP Asynchronous Circuits: Predicted Delay Variations Observed by Experiment, *2008 Festschrift Series of Springer LNCS for the Festschrift of Willem-Paul de Roever*, Kiel, Germany, 4 July 2008.

[2] Ivan Sutherland, Bob Sproull and David Harris, Logical Effort: Designing Fast CMOS Circuits, *Morgan Kaufmann*, San Francisco, 1999

ABSTRACT

This paper follows up on the Festschrift publication in [1]. In [1], Joshi et al. analyze the relative timing constraints on which the correct operation of a 6-4 GasP circuit depends. They predict correct operation over a wide range of module distances provided the difference in the distances to predecessor and successor modules is limited. They predict failure if the distances differ by too much. Their analytical prediction is supported by experiments on a 90 nanometer test chip called “Infinity” which was built by Sun Microsystems and fabricated at TSMC.

In the present paper, I continue the analysis by Joshi et al. and calculate by how much the distances may differ and still maintain the two key relative timing constraints. I present this as a two-dimensional graph, which I call the *Distance Constraint Graph*.

The *Distance Constraint Graph* covers a range of wire distances from a given module to its predecessor and successor modules. It marks the minimum and maximum allowable difference from the equidistance line where the distances to the predecessor and successor modules are the same. It does this for the two key relative timing constraints and it overlays the results to show the pass-fail regions. Specifically, the graph indicates correct operation *for all plausible wire lengths* that the 90 nanometer Infinity chip would use without inserting 6-4 GasP repeater modules. For a quick preview of the Distance Constraint Graph, see **Figure 3** on page 9 and **Figure 4** on page 11.

1. Background

Before I present my analysis results, I will give some background on 6-4 GasP circuits, their basic operation, and the two key relative timing assumptions for correct operation.

1.1. 6-4 GasP Circuits

Figure 1 shows two control stages, each with a 6-4 GasP circuit. The stages are connected in series by a bidirectional state wire L2 [1] via connection ports SUCC and PRED, as shown in **Figure 1**. The basic operation of each 6-4 GasP circuit is as follows. When PRED is high and SUCC is low, signal FIRE rises. FIRE high does three things: (1) it clocks the latches to copy the present data (2) it raises SUCC to indicate that the data values are on their way, and (3) it lowers PRED to indicate that the data have been accepted and there is space again for new data.

Actions (2) and (3) have the additional side-effect of resetting the FIRE signal to low. Either (2) or (3) can do this: gate A synchronizes PRED high and SUCC low as PMOS-AND function, but either PRED low or SUCC high will trigger the complementary NMOS-OR function of gate A. Thus, FIRE high self-resets to FIRE low, and there are two self-resetting loops to choose from: one through gates DEABC related to action (2) and another through gates XFABC related to action (3). Each one takes 5 gate delays.

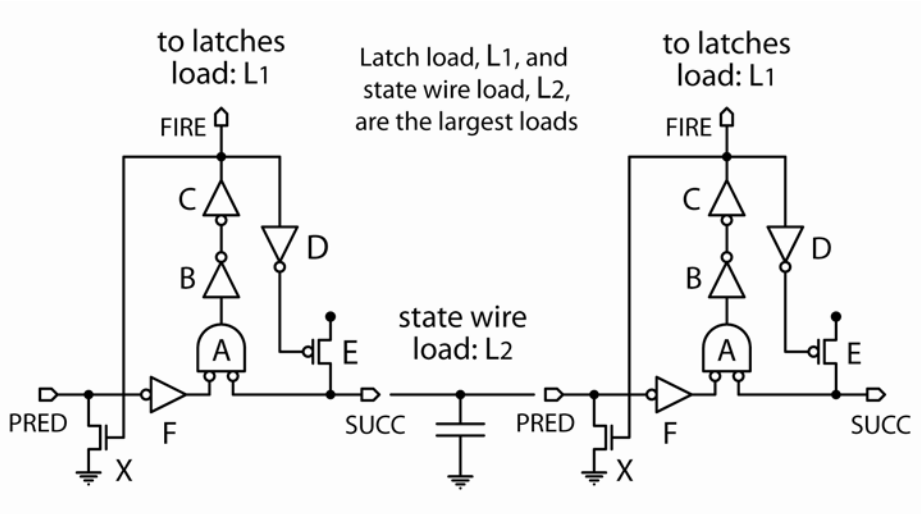

Figure 1 (Picture copied from Fig 1 in [1]) The picture shows two stages of 6-4 GasP circuit modules connected in series. The forward latency from gate A in the first stage to gate A in the second stage is 6 gate delays, and is covered by the path ABCDEF. The backward latency from gate A in the second stage to gate A in the first stage is 4 gate delays, and is covered by the path ABCX. This gives a cycle time of 10 gate delays. In addition to this global cycle time, each GasP module has two local self-resetting cycles of 5 gate delays each: in [1] these are called the successor loop, for ABCDE, and the predecessor loop, for ABCXF. It is these two self-resetting loops that incur the two key relative timing assumptions that cause the pass-fail circuit behavior that I analyze in this paper. The two relative timings are specified in **Section 1.2**, **Figure 2**.

[1] In the text of this paper, I loosely use L2 to denote either the state wire itself or its length or its load. I will be more precise in the delay calculations and presentation of the distance constraint graph in **Section 2**.

In 6-4 GasP, one can count delay in terms of logic gate delays, because the modules are custom-designed using Logical Effort [2]. This means that the transistors in each logic gate are sized such that all logic gates have the same delay. Gate sizing takes into account the transistor sizes of the gates that it drives and the connecting wire lengths.

This works as long as one can adequately predict the wire lengths. Gates A, B, D, F in **Figure 1** each drive a single local gate, and so one may assume that the connecting wire lengths are known in advance. The situation for gate C is more complicated: C drives two local gates, D and X, and a number of latches with a total latch load of L1, as well as a not so local wire between the gates and the latches. Fortunately, it is possible to place the latches near the 6-4 GasP controller, and design a macro module that contains both the data and the control and connects these with a fixed wire length.

The exceptions are gates E and X that drive state wire L2. The length of L2 is dependent on the distance between the modules, which is not known until the final system layout. If E and X are sized using the wire lengths for A, B, D and F, as was done in the Infinity chip, the gate delays for E and X will vary with different module distances and the delays of the self-resetting loops through DEABC and XFABC will vary correspondingly. If the delays vary by too much then the FIRE signal may self-reset before action (2) or (3) has completed, and cancel the action prematurely.

Premature cancellation of action (2) may result in loss of data because the successor stage never received the SUCC high indication that new data have arrived. Cancellation of action (3) may result in duplicate data because the predecessor stage never received the PRED low indicator to go ahead and replace the old data. To avoid this, Joshi et al. in [1] limit the delay variations due to module distance by enforcing so-called *relative timing constraints*. This is the topic of the next **Section 1.2**.

1.2. Relative Timing (RT) Constraints

Joshi et al. identified two key relative timing (RT) constraints to limit the delay variation between the two self-resetting loops of a GasP module. The two constraints state that one loop cannot shut off the other loop prematurely and cancel the associated action to the predecessor or successor module. The design and layout of 6-4 GasP circuits must maintain these constraints.

The two RT constraints, RT1 and RT2, are illustrated in **Figure 2**. Each constraint compares two path delays through a two-stage 6-4 GasP design, like **Figure 1**, and demands that one does not exceed the other. The lesser path delay is marked by a dotted line and corresponds to a forward or backward action through state wire L2. The larger path delay is marked by a solid line and corresponds to the self-resetting loop in the reverse direction. The paths are expressed as a sequence of up-going (+) and down-going (-) gate output transitions. **Figure 2** shows only the transitions for the key signals to the latches and neighboring GasP stages: FIRE, PRED and SUCC.

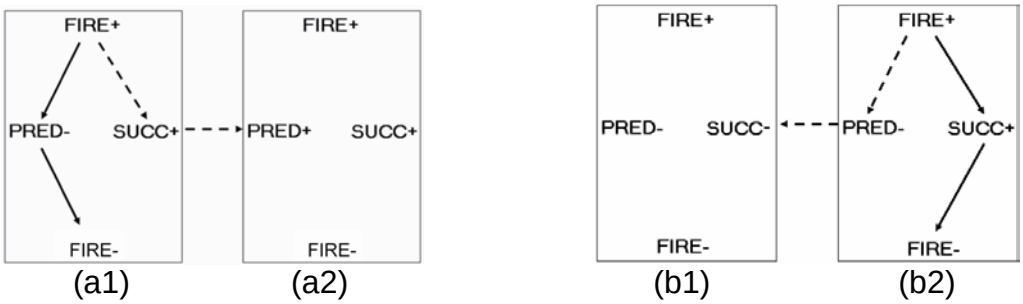

Figure 2 (Adapted from Fig 2 in [1]) The left and right sides show the two key relative timing constraints RT1 and RT2 for the two-stage 6-4 GasP design in **Figure 1**:

- RT1 is illustrated on the left for successive stages (a1)-(a2): the delay of the dotted forward path from FIRE+ to SUCC+ to PRED+ through gates DE in (a1) and up to (a2) must equal or be below the solid path delay for the self-resetting predecessor loop from FIRE+ to PRED- to FIRE- through gates XFABC in (a1).
- RT2 is illustrated on the right for successive stages (b1)-(b2): the delay of the dotted backward path from FIRE+ to PRED- to SUCC- through gate X in (b2) and up to (b1) must equal or be below the solid path delay for the self-resetting successor loop from FIRE+ to SUCC+ to FIRE- through gates DEABC in (b2).

Note that RT1 and RT2 use a different starting point for the self-resetting loops than the definitions in **Figure 1**: they starts from FIRE instead of from an input to gate A, but cover the same 5 gates — which is what matters.

The first timing constraint RT1 is illustrated for the left stage pair (a1)-(a2) of **Figure 2**. It states that PRED of the successor stage goes high before FIRE self-resets to low. In terms of the circuit diagram of **Figure 1** this *more-or-less* translates to: gates DE drive PRED of the successor stage high through state wire L2 before the self-resetting path through gates XFABCD shuts off gate E and stops it from driving PRED.

I emphasized that this translation "more-or-less" expresses what follows. This is because the RT1 expression in terms of FIRE, PRED and SUCC is slightly coarser: it stops at the FIRE input to gate D. The RT1 expression in **Figure 2** factually translates to: the path delay through DE does not exceed that of XFABC. This makes RT1 more conservative than is intended. I will come back to this in **Section 2.1**, when I present the pass-fail regions for both the conservative and the intended RT1 versions.

The second timing constraint RT2 is illustrated on the right, for stage pair (b1)-(b2) of **Figure 2**. It states that SUCC of the previous stage goes low before FIRE self-resets to low. In terms of the circuit diagram of **Figure 1**, this *exactly* translates to: gate X drives SUCC of the predecessor stage low through state wire L2 before the self-resetting path through gates DEABC shuts off gate X and stops it from driving SUCC.

Note that timing constraint RT2 can be expressed as intended in terms of transitions ending on FIRE, PRED, and SUCC.

In the following **Section 2**, I will compute the values for L2 for which RT1 and RT2 hold. More precisely, I will compute the range of values for $L2_{PRED}$ and $L2_{SUCC}$ so that the gate delays for X_{PRED} driving $L2_{PRED}$ and for E_{SUCC} driving $L2_{SUCC}$ satisfy the RT1 path delay equation $DE_{SUCC} \leq X_{PRED}FABC$. And likewise, I will compute the range of values for $L2_{PRED}$ and $L2_{SUCC}$ that satisfy RT2 path delay equation $X_{PRED} \leq DE_{SUCC}ABC$.

2. Computing Allowable Module Distances

I use the term *allowable module distances* to indicate the range of wire lengths between 6-4-GasP modules that satisfy relative timing constraints RT1 and RT2. RT1 and RT2 are path-delay constraints; they are computed from the delays of the gates and the wires on the corresponding paths. This paper adopts the wire model used by Joshi et al in [1]. This means that a wire is modeled as a lumped capacitance, and its delay is incorporated into the delay of each gate that drives this lumped capacitance. Delays due to wire resistance are ignored.[2]

Joshi et al. built a spreadsheet that calculates the gate delays in a 6-4 GasP module using wire loads that are typical for the 90 nanometer chip design of Infinity.[3] The delay calculations are based on the theory of Logical Effort [2]. Their spreadsheet calculations are repeated in **Table 1** below.

With the help of **Table 1** I compute the range of allowable module distances. I will first compute the path delays for RT1 and from these the allowable module distances for RT1. This is done **Section 2.1**. In **Section 2.2**, I do the same for RT2. I combine the two results in **Section 2.3**, where I give the full picture and relate my results back to the 90 nanometer results of the Infinity test chip presented by Joshi et al. in [1].

Table 1 (Adapted from Table 2 in [1]). Below follow the Logical Effort calculations of each gate delay in the 6-4 GasP module of Figure 1.The three columns with bold numbers for S, P and WL are fixed by design: gate size S and its self-delay P follow from the gate implementation and the gate complexity, and wire load WL follows from the wire dimensions. Short wires are indicated as `wire' (values taken from column WL); the long wires L1 and L2 are indicated explicitly. Note that the calculation of the self delay P for gates E and X counts the diffusion output capacitance of both E and X, because both their outputs connect to the state wire L2. Also note that the D2 values for E and X are different for different wire loads of L2. The Table uses the logical effort values of Table 1 in [1] (not included) to calculate the load per gate input: IL = (S * logical effort of the gate). The formulas for gate delay, wire load delay, and total delay are: D1 = GL/S, D2 = WL/S, Total = D1 + P + D2.

Gate Name	Size (S)	Load per input (IL)	Drives next gates and wire	Next gate load (GL)	Delay from GL (D1)	Self delay (P)	Next wire load (WL)	Delay from WL (D2)	Total delay [τ] [3]
A	**18**	30	B+wire	40	2.2	**2**	**9**	0.5	4.7
B	**40**	40	C+wire	100	2.5	**1**	**20**	0.5	4
C	**100**	100	D+X+ $L1_{latches}$ + $L1_{wire}$	20+20+200	2.4	**1**	**100**	1	4.4
D	**20**	20	E+wire	40	2	**1**	**20**	1	4
F	**10**	10	A+wire	30	3	**1**	**5**	0.5	4.5
E L2=15	**60**	40	A+F+L2	30+10	0.67	**with X = 1**	**15**	0.25	1.9
E L2=150	**60**	40	A+F+L2	30+10	0.67	**with X = 1**	**150**	2.5	4.2
X L2=15	**60**	20	A+F+L2	30+10	0.67	**with E = 1**	**15**	0.25	1.9
X L2=150	**60**	20	A+F+L2	30+10	0.67	**with E = 1**	**150**	2.5	4.2

[2] Delays due to wire resistance and other long wires effects will be included in a follow-up paper.
[3] The delay unit here and in [1] is `τ' (tau): the characteristic delay of an inverter in the given process.

2.1. Path Delays and Allowable Module Distances for RT1

From **Figure 2(a1)-(a2)** one can see that relative timing constraint RT1 involves both state wires at both ends of the 6-4 GasP module: the predecessor state wire participates in the solid backward path that self-resets FIRE from FIRE+ to FIRE-; the successor state wire participates in the dotted forward path from FIRE+ to PRED+.

To analyze RT1 it makes sense to distinguish these two state wires. I will use $L2_{PRED}$ to denote the state wire that connects the predecessor port PRED of a 6-4 GasP module to the previous stage. I will use $L2_{SUCC}$ to denote the state wire that connects the successor port SUCC of a 6-4 GasP module to the next stage. This matches with the terminology in **Figure 1**. For convenience, I wish to identify the gates that drive $L2_{PRED}$ respectively $L2_{SUCC}$, so that the name of the gate indicates that the gate delay depends on the load value of $L2_{PRED}$ respectively $L2_{SUCC}$. To this end, I rename gate X as X_{PRED} and gate E as E_{SUCC}.

In **Section 1.2**, I distinguished two RT1 constraints: a conservative version and the intended version. The conservative version, which I will call $RT1_{conservative}$, states that the forward path delay from FIRE+ to SUCC+ to PRED+ through D and E_{SUCC} must be less than or equal to the backward self-resetting path from FIRE+ to FIRE- through the series of gates X_{PRED}, F, A, B and C. The intended RT1 constraint version, which I will call $RT1_{intended}$, is similar, but has one more gate in the backward path: X_{PRED}FABCD.

I use the gate delay calculation procedure in **Table 1** to compute the backward self-resetting path delays for X_{PRED}FABC. For each fixed value of $L2_{PRED}$, this gives me the maximum forward path delay for DE_{SUCC} for which $RT1_{conservative}$ holds. Given this, I reverse-engineer the calculation procedure in **Table 1** to obtain the corresponding maximum load value of $L2_{SUCC}$ that satisfies $RT1_{conservative}$.[4]

The resulting difference in load values $L2_{SUCC}$-$L2_{PRED}$ between the successor and predecessor state wires represents the maximum allowable module distance in terms of load capacity. I call this the *margin value*. $L2_{SUCC}$-$L2_{PRED}$ values below the margin value pass the relative timing constraint, those above fail the constraint.

The computations for the $RT1_{conservative}$ margin values are outlined in **Table 2**. It turns out that the margin value is approximately constant, 816 units of load, over the given range of $L2_{PRED}$ state wire loads.[5]

[4] The load capacitance unit here and in [1] is `X': the gate load, or capacitance, of the smallest inverter one can build in the given process.

[5] Reminder: I have used a lumped wire load. I ignored long wire delay effects, which are outside the scope of this study and which will be included in a follow-up paper.

Table 2 ($RT1_{conservative}$ margin) Calculations of the maximum allowable difference between the wire load $L2_{SUCC}$ of a given module to its successor module and the wire load $L2_{PRED}$ to its predecessor module. The calculations show that differences up to and including 816 units of load satisfy relative timing constraint RT1 of **Figure 2(a1)-(a2)**, because $DE_{SUCC} \leq X_{PRED}FABC$. The calculation procedure is as follows. Given the range of $L2_{PRED}$ values, I compute the path delays for $X_{PRED}FABC$. The computations are based on spreadsheet calculations using gate delays from **Table 1** and its extension to a wide range of L2 wire loads. The path delay of $X_{PRED}FABC$ determines the maximum path delay for DE_{SUCC}. Given the maximum delay for DE_{SUCC}, I use a logarithmic search and additional spreadsheet calculations to get the corresponding maximum load for $L2_{SUCC}$. This gives a maximum allowable difference of 816 units of load from $L2_{SUCC}$ to $L2_{PRED}$.

$L2_{PRED}$ [X] [4] (fixed)	$X_{PRED}FABC$ [τ] [3] (computed from Table 1)	DE_{SUCC} [τ] (set to $X_{PRED}FABC$ value)	$L2_{SUCC}$ [X] (computed from DE_{SUCC})	$L2_{SUCC} - L2_{PRED}$ [X]
10	19.4	19.4	826	816
200	22.6	22.6	1016	816
500	27.6	27.6	1316	816
1000	35.9	35.9	1816	816
2000	52.6	52.6	2816	816
10000	185.9	185.9	10816	816
12000	219.3	219.3	12816	816
15000	269.3	269.3	15816	816

The computations for the margin values of $RT1_{intended}$ are similar. The only difference is that the backward self-resetting path delay for $X_{PRED}FABCD$ is larger than the previous self-resetting path delay for $X_{PRED}FABC$ in **Table 2**. Specifically, it is larger by a fixed amount, namely by the gate delay value of D, which is independent of $L2_{PRED}$.

From the previous calculations I know that the margin value for $RT1_{conservative}$ is constant over the given range of $L2_{PRED}$ load values. This implies that the margin value for $RT1_{intended}$ is also constant, and moreover the constant value is larger than 816. **Table 3** gives the computations for the margin value of $RT1_{intended}$. It suffices to do these computations for one fixed value of $L2_{PRED}$ because I know by now that the margin values for $L2_{SUCC}$-$L2_{PRED}$ are the same for the entire $L2_{PRED}$ range.[5]

Table 3 ($RT1_{intended}$ margin) Calculations of the maximum allowable difference between the wire load $L2_{SUCC}$ of a given module to its successor module and the wire load $L2_{PRED}$ to its predecessor module. The calculations show that differences up to and including 1056 units of load satisfy the intended relative timing constraint $RT1_{intended}$ for **Figure 2(a1)-(a2)**, discussed in **Section 1.2**, because $DE_{SUCC} \leq X_{PRED}FABCD$. The calculation procedure is as in **Table 2**, but adds a fixed gate delay for gate D. The result is a larger maximum allowable difference of 1056 units of load from $L2_{SUCC}$ to $L2_{PRED}$.

$L2_{PRED}$ [X] [4] (fixed)	$X_{PRED}FABCD$ [τ] [3] (computed from Table 1)	DE_{SUCC} [τ] (set to $X_{PRED}FABCD$ value)	$L2_{SUCC}$ [X] (computed from DE_{SUCC})	$L2_{SUCC} - L2_{PRED}$ [X]
10	23.4	23.4	1066	1056

2.2. Path Delays and Allowable Module Distances for RT2

The calculations for the margin values for relative timing constraint RT2 follow the same procedure as those for $RT1_{conservative}$ and $RT1_{intended}$ in **Section 2.1.**

I use the gate delay calculation procedure in **Table 1** to compute the forward self-resetting path delays for $DE_{SUCC}ABC$. For each fixed value of $L2_{SUCC}$, this gives me the maximum backward path delay for X_{PRED} for which RT2 holds. Given this, I reverse-engineer the calculation procedure in **Table 1** to obtain the corresponding maximum load value of $L2_{PRED}$. The resulting difference in load values $L2_{SUCC}$-$L2_{PRED}$ between the successor and predecessor state wires represents the maximum allowable module distance in terms of load capacity for RT2. This is the *margin value* for RT2, and it is negative: $L2_{SUCC}$-$L2_{PRED}$ values above the margin value pass the relative timing constraint, those below fail the constraint.

The computations for the RT2 margin values are outlined in **Table 4**. Also here, the margin value is approximately constant, -1026 load units, over the given range of $L2_{SUCC}$ state wire loads.[6]

Table 4 (RT2 margin) Calculations of the maximum allowable difference between the wire load $L2_{SUCC}$ of a given module to its successor module and the wire load $L2_{PRED}$ to its predecessor module. The calculations show that $L2_{SUCC}$ – $L2_{PRED}$ differences down to and including -1026 units of load satisfy relative timing constraint RT2 of **Figure 2(b1)-(b2)**, because $X_{PRED} \leq DE_{SUCC}ABC$. The calculation procedure is similar to that of **Table 2**. Given the range of $L2_{SUCC}$ values, I compute the path delays for $DE_{SUCC}ABC$. This determines the maximum path delay for X_{PRED}. From here, a logarithmic search gives the corresponding maximum load value for $L2_{PRED}$, and a maximum allowable difference down to -1026 units of load from $L2_{SUCC}$ to $L2_{PRED}$.

$L2_{SUCC}$ [X] [4] (fixed)	$DE_{SUCC}ABC$ [τ] [3] (computed from Table 1)	X_{PRED} [τ] (set to $E_{SUCC}DABC$ value)	$L2_{PRED}$ [X] (computed from X_{PRED})	$L2_{SUCC}$ - $L2_{PRED}$ [X]
10	18.9	18.9	1036	-1026
200	22.1	22.1	1226	-1026
500	27.1	27.1	1526	-1026
1000	35.4	35.4	2026	-1026
1500	43.8	43.8	2526	-1026
10000	185.4	185.4	11026	-1026
12000	218.8	218.8	13026	-1026
15000	268.8	268.8	16026	-1026

[6] Reminder: also in the delay calculations for RT2 I have used a lumped wire load. I ignored long wire delay effects, which fall outside the scope of this study and will be included in a follow-up paper.

2.3. The Full Picture: Distance Constraint Graph

The graph in **Figure 3** below helps visualize the result of the previous two Sections. This graph — which I call the *Distance Constraint Graph* — shows the allowable distances from a 6-4 GasP module to its predecessor and successor modules. I calculated the distance in units of length [7] as well as in units of load. The solid and dashed blue lines are the margin lines for $RT1_{conservative}$ and $RT1_{intended}$. The green line is the margin line for RT2. Predecessor-successor pairs ($L2_{PRED}$, $L2_{SUCC}$) in the top-left blue region fail $RT1_{conservative}$. Those in the bottom-right green region fail RT2. Those in the white region in-between satisfy $RT1_{conservative}$ (and hence also $RT1_{intended}$) and RT2.

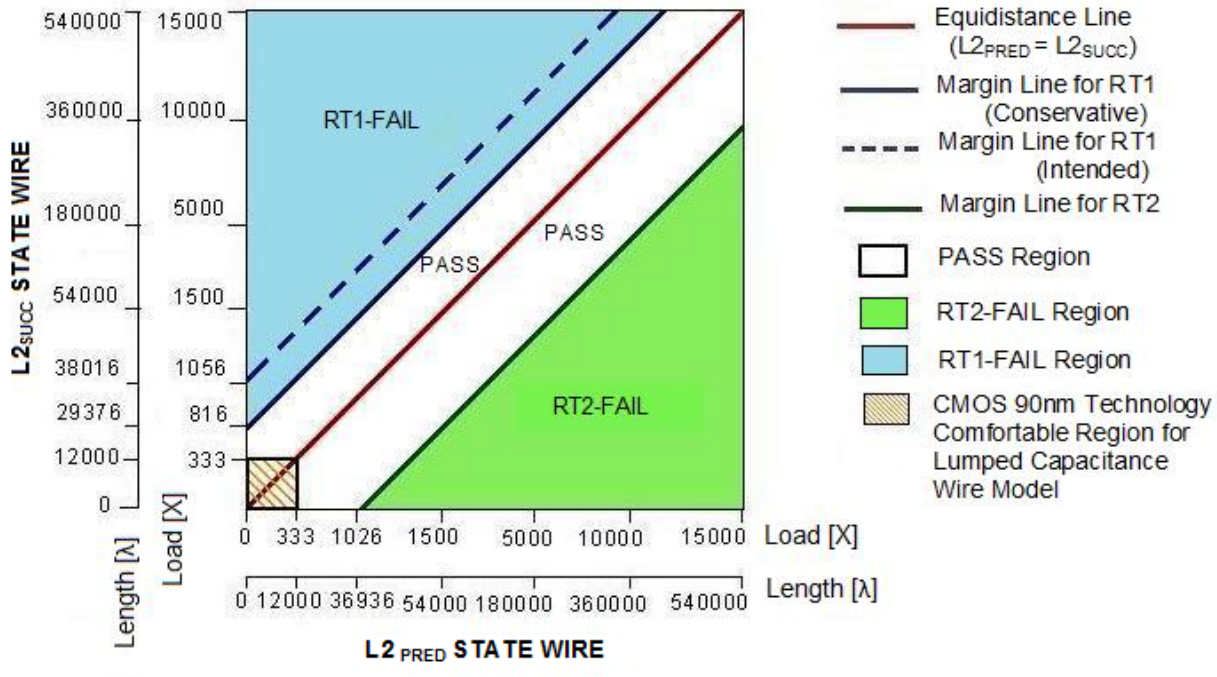

Figure 3 (Distance Constraint Graph) The graph shows the allowable distances from a 6-4 GasP module to its predecessor and successor modules. The X-axis shows the distance to the predecessor module, $L2_{PRED}$. The Y-axis shows the distance to the successor module, $L2_{SUCC}$. Both $L2_{PRED}$ and $L2_{SUCC}$ are expressed in units of length as well as in units of load. The conversion is based on the 90 nanometer CMOS technology used for Infinity, in which 36 units of length equals 1 unit of load — see also footnotes 4 and 7. The solid and dashed blue lines are the margin lines for $RT1_{conservative}$ and $RT1_{intended}$ from **Table 2** and **Table 3**. The Predecessor-successor pairs ($L2_{PRED}$, $L2_{SUCC}$) in the blue region above the solid line fail relative timing constraint $RT1_{conservative}$ because $L2_{SUCC}$-$L2_{PRED}$ exceeds 816 units of load (29376 units of length). Those above the interrupted blue line also fail $RT1_{intended}$. The solid green line is the margin line for RT2, as computed in **Table 4**. The predecessor-successor pairs ($L2_{PRED}$, $L2_{SUCC}$) in the green region below it fail RT2 because $L2_{SUCC}$-$L2_{PRED}$ is below -1026 units of load (-36936 units of length). The ($L2_{PRED}$, $L2_{SUCC}$)-pairs in the white region between the blue and green lines satisfy both relative timing constraints $RT1_{conservative}$ and RT2.

[7] The length unit here and in [1] is `λ' (lambda): 1/20-th of a micrometer in the TSMC 90 nanometer process used for Infinity.

In addition to the allowable module distances and their relation to the key relative timing constraints, **Figure 3** shows two other interesting facts.

1. The red equidistance line where $L2_{PRED}$ and $L2_{SUCC}$ have the same load and length falls in the PASS region, approximately midway between the margin lines for $RT1_{intended}$ and RT2. This is good news: it makes designing GasP systems easier because the designer can assume equidistant module distances and still leave adequate margins to place and route the modules.

2. The orange-striped square box of about 333 x 333 units of load in the bottom-left corner of the graph represents the short-to-medium length wire region for the TSMC 90 nanometer CMOS manufacturing process that was used for Infinity. In this process, 333 units of wire load correspond to approximately 12000 units of wire length. This is the comfort region for the lumped capacitance wire model.

   The anticipation is that for wires longer than 12000 units, the wire resistance is no longer negligible, and neither are the rise and fall times of the wire transitions. To mitigate these long wire effects, one can insert extra 6-4 GasP modules, called *repeater modules*, at state wire intervals of 12000 units of length.

   Note that the orange-striped box is completely within the white PASS region of **Figure 3**. So, apparently, the repeater constraints to avoid long wires are much stricter than the 6-4 GasP relative timing constraints for functional correctness.[8]

3. Conclusion

The 6-4 GasP modules discussed in this paper are used to route data. The operation of each module is limited by relative timing constraints. In [1] Joshi et al. identified two key constraints, called RT1 and RT2, that limit the module distance between successive GasP modules in the router network. They also set up a framework to validate whether the distances from a 6-4 GasP module to its predecessor and successor are allowed.

In this paper, I introduce a new representation, the *Distance Constraint Graph*, to visualize the relation between the allowable module distances and the relative timing constraints RT1 and RT2. By looking at the graph one can immediately locate the pass and fail regions for each constraint for up to 15000 load units, i.e. 540000 length units.

There is one caveat: my current calculations ignore long wire effects — these will be taken into account in a follow-up paper.[9]

[8] I surely will investigate this further in my long-wire analysis.
[9] See also footnotes 5, 6 and 8.

Using the Distance Constraint Graph, it is immediately clear that the short to medium range module distances used in the 90 nanometer Infinity test chip are safe. Infinity uses up to 150 units of load, or 5400 units of length. Distances of length 5400 are safe regardless of the difference in distances to predecessor and successor modules.

Figure 4 offers an enlarged view of the orange-striped 6-4 GasP operating region from **Figure 3** with short-to-medium range module distances.

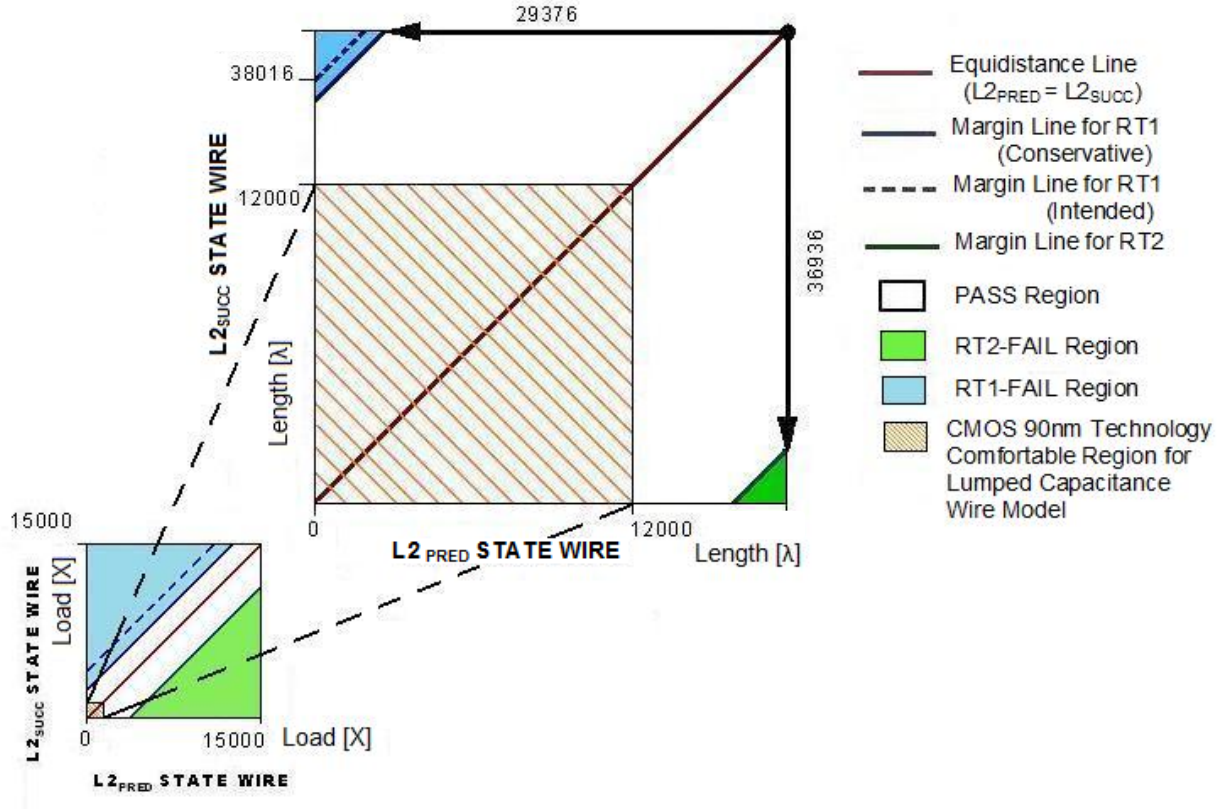

Figure 4 (Detail of Figure 3) Detailed segment of the Distance Constraint Graph for short-to-medium length module distances for 6-4 GasP in TSMC's 90 nanometer CMOS manufacturing process. The short-to-medium length distances are marked by the orange –striped region of 12000x12000 units of wire length. Note that the orange-striped region falls completely inside the white (PASS) region.

I anticipate that the long wire effect will kick in somewhere near the top and right boundaries of the orange-striped region. From there on, the margin lines for RT1 and RT2 will likely no longer be constant. My next research topic is to investigate how exactly the margin lines will change in relation to the individual module distances and in relation to the relative distance between predecessor and successor modules.