Description

Technical Field

The present invention concerns an accurate timing model for logic simulation of integrated circuits. The timing model reflects the timing delays caused by input ramp and output load of circuitry.

Background Art

When designing an integrated circuit, it is generally desirable to simulate the functioning of logic circuitry within the integrated circuit. In order to accurately access the performance of the circuitry, it is additionally desirable that the logic simulation of logic include an accurate assessment of timing delays through the circuitry, at least in the critical paths.

In general, timing delays through circuitry are caused by propagation delays through and between logic cells which comprise the circuitry. The actual amount of propagation delay through and between logic cells is generally dependent on various capacitances within and between the logic cells, as well as the current available to charge or discharge the capacitances.

When developing a timing model, it is desirable to take into account the time delay introduced by charging the input capacitance of logic cells. This time delay is directly affected by the input current available to charge the input capacitance to the logic cell. This input current, in turn, is directly affected by the fan out of the output of the logic cell providing the input current.

In the prior art, timing delay caused by charging input capacitance was approximated as being directly proportional to the output fan out of the logic cell providing current for the input. However, this has proved to be an inaccurate modelling technique because propagation delay through the logic cell is additionally influenced by the shape of the voltage input ramp of the input signal to the logic cell. The effect of the shape of the voltage input ramp on the timing delay may be approximated by using general piecewise linear functions of the voltage input ramp and the output load. See for example, An-Chang Deng, Piecewise-Linear Timing Delay Modeling for Digital CMOS Circuits, *IEEE* Transactions on Circuits and Systems, vol. 35, no. 10, October 1988, pp. 1330-1334. However, this approach does not easily lead to a practical characterization system. In order to obtain a fairly accurate timing model, a circuit simulation needs to be performed for every grid point of the function. A sufficient number of grid points need to be selected to obtain the desired accuracy of the timing model. This can require numerous circuit simulations.

Disclosure of the Invention

In accordance with the preferred embodiment, the present invention provides a method running on a computer for approximating propagation delay through a logic device, the method comprising the steps of: (a) dividing operation of the logic device into a fast input ramp region, wherein an input ramp to the logic device is completely switched before the output signal wave form changes significantly, and a slow input ramp region, wherein the input ramp switches so slowly that the output signal switches while the input ramp is still switching, a boundary between the fast input ramp region and the slow input ramp region being based on duration of input ramp to the logic device and amount of capacitive load driven by the logic device; (b) when the logic device operates in the fast input ramp region, using a first formula to obtain a first value representing delay through the logic device, the first formula varying the first value based on the duration of the input ramp to the logic device and the capacitive load driven by the logic device; and (c) when the logic device operates in the slow input ramp region, using a second formula to obtain the first value, the second formula varying the first value based on the duration of the input ramp to the logic device and the capacitive load driven by the logic device.

For example, the first formula may have a form as follows:

$$DT_0 = A_0 + B * (\text{INPUT RAMP}) + D_0 * (\text{LOAD})$$

where DT_0 represents the first value, INPUT RAMP represents duration of the input ramp, LOAD represents capacitive load driven by the logic device, and A_0, B and D_0 are coefficients determined using circuit simulations.

Similarly, the second formula may have a form as follows:

$$DT_0 = A_0 + B * (\text{CRITINRAMP}) + D_0 * (\text{LOAD}) + Z * (\text{INPUT RAMP} - \text{CRITINRAMP})$$

where CRITINRAMP represents INPUT RAMP at the border between the first region and the second region for the LOAD, and Z is determined using circuit simulations.

The above examples for the first formula and the second formula use first order linear approximations for the input ramp of the logic device. Alternately a second order approximation of the input ramp may include "tails"

of the input ramp. Using the second order model, the first formula may have a form as follows:

$$DT_0 = A_0 + FAT * B * (INPUT\ RAMP) + D_0 * (LOAD)$$

wherein DT_0 represents the first value, INPUT RAMP represents duration of the input ramp, LOAD represents capacitive load driven by the logic device, and A_0, B, D_0 and FAT are coefficients determined using circuit simulations.

Similarly, using the second order model, the second order model may have a form as follows:

$$DT_0 = DT_{0(CRITINRAMP)} + Z * (INPUT\ RAMP - CRITINRAMP)$$

wherein CRITINRAMP represents INPUT RAMP at the border between the first region and the second region for the LOAD, Z is a coefficient determined using circuit simulations, and $DT_{(CRITINRAMP)}$ is the first value when the logic ramp has an input ramp equal to CRITINRAMP and a load equal to LOAD.

In the preferred embodiment of the present invention, the method also includes calculating a next input ramp for a next logic device coupled to an output of the logic device. When the logic device operates in the first region, a third formula is used to generate a second value representing duration from beginning of the input ramp of the logic device to an end of an output ramp of the logic device. The second value is based on the length of the input ramp to the logic device and the amount of the load driven by the logic device. When the logic device operates in the second region, a fourth formula is used to generate the second value. The first value is then subtracted from the second value to obtain the next input ramp.

For example, the third formula may have a form as follows:

$$DT_1 = A_1 + B * (INPUT\ RAMP) + D_1 * (LOAD)$$

where DT_1 represents the second value, and A_1 and D_1 are coefficients determined using circuit simulations. Similarly, the fourth formula may have a form as follows:

$$DT_1 = A_1 + B * (INPUT\ RAMP) + D_1 * (LOAD)..$$

Alternately, using the second order model which takes into account "tails" of the input ramp, the third formula may have a form as follows:

$$DT_1 = A_1 + FAT * B * (INPUT\ RAMP) + D_1 * (LOAD)$$

wherein DT_1 represents the second value, and A_1 and D_1 are coefficients determined using circuit simulations. Similarly, using the second order model, fourth formula may have a form as follows:

$$DT_1 = DT_{1(CRITINRAMP)} + B * (INPUT\ RAMP - CRITINRAMP).$$

The present invention is advantageous over the prior art in that it allows for a fairly accurate timing model of a logic device with the need to perform only a limited number of circuit simulations.

The closest prior art is given by D 1 (IEEE Trans. on Circuits and Systems, vol. 35, no. 10, Oct. 1988, N.Y.; pages 1330-1334; An-Chang Deng: "Piecewise-Linear Timing Delay Modeling for Digital CMOS Circuits"), disclosing a timing model that divides the operation of the logic device into a first and second region, based on the direction of the input ramp. However D1 does not distinguish between two different input ramps, whereby the distinction is made due to the behaviour of these input ramps with respect to the behaviour of the corresponding output signals.

Brief Description of the Drawings

Figure 1 shows an example of circuitry through which propagation delay may be calculated.

Figure 2 shows a graph on which is superimposed a ramp waveform.

Figure 3 shows a graph on which is superimposed a ramp waveform.

Figure 4 shows two operation regions for a logic device in accordance with the preferred embodiment of the present invention.

Figure 5 shows a graph on which is superimposed an input ramp waveform and a resulting output ramp waveform in a first operation region shown in Figure 4 in accordance with the preferred embodiment of the present invention.

Figure 6 shows a graph on which is superimposed an input ramp waveform and a resulting output ramp waveform in a second operation region shown in Figure 4 in accordance with the preferred embodiment of the present invention.

Figure 7 shows a circuit used for a second order model simulation in accordance with the preferred embodiment of the present invention.

Description of the Preferred Embodiment

There are many causes of propagation delay through logic circuitry. For example, Figure 1 shows circuitry which includes a logic device 11, a logic device 12, a logic device 13, a logic device 14, a logic device 15, a logic device 16, a logic device 17, a logic device 18, a logic device 19, a logic device 20 and a logic device 21

connected as shown. For simplicity in the discussion of the present invention, inverters as used for the logic devices. However, the present invention is applicable to all logic devices used in the manufacture of integrated circuits.

Causes for propagation delay through the circuitry shown in Figure 1 include, for example, the time required to charge or discharge internal capacitance within each of logic devices 11-21, the time required to charge or discharge input capacitance to each of logic devices 11-21, and the time required to overcome inductance in the lines between logic devices 11-21. The present invention, however, focuses on the effects that input transition time and output load has on propagation delay. The input transition time of a logic device is the time required to charge or discharge input capacitance of the logic device. The output load is the total amount of capacitance which the logic device has to charge or discharge. The output load is directly affected by, for example, the fan out of the output of the logic device.

For example, as shown in Figure 1, logic device 11 has a fan out of two. Logic device 12 has a fan out of three. Logic devices 13-17 have a fan out of one. Because of the relatively large fan out of logic device 12, it is expected that the output transition time of(and thus the propagation delay through) logic device 12 will take longer than, for example, the output transition time of logic device 13, which has a fan-out of only one.

Figure 2 shows a graph 129 on which is superimposed a ramp waveform 153 which represents a voltage ramp during a transition period. A vertical axis 128 of graph 129 represents voltage at an input or an output of a logic device. A horizontal axis 127 represents passage of time. A zero point 130 of graph 129 indicates zero voltage and the passage of zero nanoseconds. A point 141 on horizontal axis 127 represents the passage of 2.5 nanoseconds. A point 142 on horizontal axis 127 represents the passage of 5.0 nanoseconds. A point 143 on horizontal axis 127 represents the passage of 7.5 nanoseconds. A point 144 on horizontal axis 127 represents the passage of 10.0 nanoseconds. A point 131 on vertical axis 128 represents 0.5 volts. A point 132 on vertical axis 128 represents 1.0 volts. A point 133 on vertical axis 128 represents 1.5 volts. A point 134 on vertical axis 128 represents 2.0 volts. A point 135 on vertical axis 128 represents 2.5 volts. A point 136 on vertical axis 128 represents 3.0 volts. A point 137 on vertical axis 128 represents 3.5 volts. A point 138 on vertical axis 128 represents 4.0 volts. A point 139 on vertical axis 128 represents 4.5 volts. A point 140 on vertical axis 128 represents 5.0 volts.

Ramp waveform 153 shows voltage verses time for a transition from low to high. Ramp waveform 153 may be approximated by a waveform 154. Waveform 154 approximates as a straight line the transition of ramp waveform 153 from a low voltage to a high voltage. The total transition time 155, or ramp, of ramp waveform 153 is defined as the duration of time between a time 151 (T_0), which is the point at which waveform 154 begins to increase in voltage, and a time 152 (T_1), which is the point at which waveform 154 ceases to increase in voltage.

Figure 3 shows a ramp waveform 163 superimposed on graph 129. Ramp waveform 163 shows voltage verses time for a transition from high to low. Ramp waveform 163 may be approximated by a waveform 164. Waveform 164 approximates as a straight line the transition of ramp waveform 163 from a high voltage to a low voltage. The total transition time 165, or ramp, of ramp waveform 163 is defined as the duration of time between a time 161 (T_0), which is the point at which waveform 164 begins to decrease in voltage, and a time 162 (T_1), which is the point at which waveform 164 ceases to decrease in voltage.

The terminology introduced in the discussion of Figures 2 and 3 may be used in the general description of input and output transition timing for a logic cell. For an input pin i (not shown) of a logic cell, the beginning of a ramp is defined as $T_{0,I}$ and the end of the ramp is defined as $T_{1,I}$. For an output pin j (not shown) of a logic cell, the beginning of a ramp is defined as $T_{0,J}$ and the end of the ramp is defined as $T_{1,J}$. Using this terminology, 50%-50% propagation delay (Tau) between input pin i and output pin j, is given by the following Formula 1:

$$\text{Formula 1} \qquad Tau,ij = 1/2(T_{1,J} + T_{0,J}) - 1/2(T_{1,I} + T_{0,I})$$

While it is often the case that propagation delay is calculated using Tau, for the present invention, other calculated values may also be used. For example, the value DT_0 is defined as the difference in time between the beginning of the ramp for the input and the beginning of the ramp for the output. More generally, the value DT_0 between input pin i and output pin j is described by the following Formula 2:

$$\text{Formula 2} \qquad DT_{0,IJ} = T_{0,J} - T_{0,I}$$

The value DT_1 is defined as the difference in time between the beginning of the ramp for the input and the end of the ramp for the output. More generally, the value DT_1 between input pin i and output pin j is described by the following Formula 3:

$$\text{Formula 3} \qquad DT_{1,IJ} = T_{1,J} - T_{0,I}$$

For the present invention, two regions of operation of a logic device are considered. These regions are shown in Figure 4. In Figure 4, a vertical axis 171 represents increases in output load for the logic device. An axis 172 represents increase in time duration for the input ramp of the input ramp. A first region 174 is referred to as the fast input ramp region because in this region the input ramp rises relatively fast in comparison to the

output ramp. A second region 175 is referred to as the slow input ramp region because in this region the input ramp rises relatively slowly in comparison to the output ramp. A dividing line 173 between fast input ramp region 174 and slow input ramp region 175 may be determined empirically as further discussed below.

Figure 5 shows an example of an input ramp 251 and a resulting output signal waveform 252 in fast input ramp regions 174. As may be seen from Figure 5, input ramp 251 falls at a relatively fast rate as compared to the rate of the rise of output signal waveform 252.

Input ramp 251 and output signal waveform 252 are superimposed on a graph 229. A vertical axis 228 of graph 229 represents voltage. A horizontal axis 227 represents passage of time. A zero point 230 of graph 229 indicates zero voltage and the passage of zero nanoseconds. A point 241 on horizontal axis 227 represents the passage of 1.67 nanoseconds. A point 242 on horizontal axis 227 represents the passage of 3.33 nanoseconds. A point 243 on horizontal axis 227 represents the passage of 5.00 nanoseconds. A point 244 on horizontal axis 227 represents the passage of 6.67 nanoseconds. A point 245 on horizontal axis 227 represents the passage of 8.33 nanoseconds. A point 246 on horizontal axis 227 represents the passage of 10.0 nanoseconds. A point 231 on vertical axis 228 represents 0.5 volts. A point 232 on vertical axis 228 represents 1.0 volts. A point 233 on vertical axis 228 represents 1.5 volts. A point 234 on vertical axis 228 represents 2.0 volts. A point 235 on vertical axis 228 represents 2.5 volts. A point 236 on vertical axis 228 represents 3.0 volts. A point 237 on vertical axis 228 represents 3.5 volts. A point 238 on vertical axis 228 represents 4.0 volts. A point 239 on vertical axis 228 represents 4.5 volts. A point 240 on vertical axis 228 represents 5.0 volts.

Figure 6 shows an example of an input ramp 281 and a resulting output signal waveform 282 in slow input ramp regions 175. As may be seen from Figure 6, input ramp 281 falls at a relatively slow rate as compared to the rate of the rise of signal waveform 282.

Input ramp 281 and output signal waveform 282 are superimposed on a graph 259. A vertical axis 258 of graph 259 represents voltage. A horizontal axis 257 represents passage of time. A zero point 260 of graph 259 indicates zero voltage and the passage of zero nanoseconds. A point 271 on horizontal axis 257 represents the passage of 1.25 nanoseconds. A point 272 on horizontal axis 257 represents the passage of 2.5 nanoseconds. A point 273 on horizontal axis 257 represents the passage of 3.75 nanoseconds. A point 274 on horizontal axis 257 represents the passage of 5.0 nanoseconds. A point 275 on horizontal axis 257 represents the passage of 6.25 nanoseconds. A point 276 on horizontal axis 257 represents the passage of 7.5 nanoseconds. A point 261 on vertical axis 258 represents 0.5 volts. A point 262 on vertical axis 258 represents 1.0 volts. A point 263 on vertical axis 258 represents 1.5 volts. A point 264 on vertical axis 258 represents 2.0 volts. A point 265 on vertical axis 258 represents 2.5 volts. A point 266 on vertical axis 258 represents 3.0 volts. A point 267 on vertical axis 258 represents 3.5 volts. A point 268 on vertical axis 258 represents 4.0 volts. A point 269 on vertical axis 258 represents 4.5 volts. A point 270 on vertical axis 258 represents 5.0 volts.

In fast input ramp region 174, input ramp 251 is completed switched before output signal waveform 252 significantly changes. For a CMOS technology implementation of an inverter, one transistor of the output buffer of the inverter would be on in saturation and the other transistor of the output buffer of the inverter would be off during substantially all the time output signal waveform 252 switches. This is essentially equivalent to a constant current source charging or discharging the load capacitance. In the preferred embodiment of the present invention, for fast input region 174, value DT_0 and value DT_1 are approximated using the following Formula 4 and Formula 5, respectively:

$$\underline{\text{Formula 4}} \qquad DT_0 = A_0 + B * (\text{INPUT RAMP}) + D_0 * (\text{LOAD})$$

$$\underline{\text{Formula 5}} \qquad DT_1 = A_1 + B * (\text{INPUT RAMP}) + D_1 * (\text{LOAD})$$

In Formulas 4 and 5, values A_0, A_1, B, D_0 and D_1 are coefficients which may be determined empirically.

In slow input ramp region 175, input ramp 281 switches so slowly that output signal waveform 282 switches while input ramp 282 is still switching. For a CMOS technology implementation of an inverter, both transistors of the output buffer of the inverter are on when output signal waveform 282 switches. In the preferred embodiment of the present invention, for slow input region 175, value DT_0 and value DT_1 are approximated using the following Formula 6 and Formula 7, respectively:

$$\underline{\text{Formula 6}} \qquad DT_0 = A_0 + B * (\text{CRITINRAMP}) + D_0 * (\text{LOAD}) + Z * (\text{INPUT RAMP} - \text{CRITINRAMP})$$

$$\underline{\text{Formula 7}} \qquad DT_1 = A_1 + B * (\text{INPUT RAMP}) + D_1 * (\text{LOAD})$$

In formula 6, the additional co-efficient Z is a proportionality factor calculated from the DC transfer function between the input voltage and the output voltage. In the slow ramp region, a plot of DT_0 versus the input ramp is a straight line for a given output load. Z is the slope of that line and can be calculated from the DC transfer curve. CRITINRAMP is the INPUT RAMP for each load which is along dividing line 173 between fast input ramp region 174 and slow input ramp region 175. In the preferred embodiment of the present invention, the CRITINRAMP (and thus dividing line 173) is defined to be at the point for each load where the end of the input ramp occurs at 50% of the output ramp. Since time 0 is considered to be at the beginning of the input ramp, then

the end of the input ramp occurs at INPUT RAMP. Therefore, CRITINRAMP occurs when INPUT RAMP equals $1/2 * (DT_0 + DT_1)$. CRITINRAMP is therefore defined by the following Formula 8:

Formula 8 $\quad CRITINRAMP = 1/2 * (DT_0 + DT_1)$

The coefficients for the formulas may be calculated using circuit simulations. First, three circuit simulations are performed to determine values for the coefficients (A_0, A_1, B, D_0 and D_1) used in formula 4 and formula 5. In the circuit simulations values for load and input ramp are chosen so that the three circuit simulations will all fall within fast input ramp region 174. Actually, initially the location of dividing line 173 is not known so that the actual boundaries of fast input ramp region 174 are not known. Therefore, later in the model it is necessary to perform a check to see if the initial three circuit simulations were in fast input ramp region 174. If not, the circuit simulations are performed again using a faster input ramps and/or higher loads.

For each simulation, using formulas 2 and 3, DT_0 and DT_1 are found. Then using formulas 4 and 5, A_0, A_1, B, D_0 and D_1 may be calculated. A first consistency check may be done by solving formulas 4 and 5 separately for the coefficient B. If the coefficient B does not have the same value, then at least one of the circuit simulations was not in fast input ramp region 174.

From the calculated values for coefficients A_0, A_1, B, D_0 and D_1, using Formula 8, CRITINRAMP is calculated. Once CRITINRAMP is found, the chosen circuit simulations may again be checked to be sure they fall within fast input ramp region 174.

An additional circuit simulation is performed to find additional points on the DC transfer curve. From these points, the DC transfer function may be approximated. A linear approximation of the DC transfer function yields the proportionality constant Z.

In a manner similar to that described above, coefficients are calculated for each logic device type in a logic device library. Delay through a circuit built from logic devices in the logic device library may then be calculated. Total propagation delay through a circuit is the total amount of time for a signal to propagate through the circuit. The path of logic cells in a circuit through which a signal takes longest to propagate is called the critical path of the circuit. Total propagation for a circuit may be calculated by summing up the total delays through all logic cells in the critical path of the circuit.

Formula 1, for example, may be used to calculate delay through each logic cell. Additionally, starting with the input of the circuit and continuing through to the output of the circuit, the input ramp and the resulting output ramp (DT_1 - DT_0) may be calculated for each logic device in the logic path. The output ramp for one logic device is the input ramp for the next logic device. The output ramps are separately calculated for both rising inputs and falling inputs.

To more fully illustrate how the present invention is practiced, an example is given below for the case of an inverter. In the example, three simulations are done with the input ramp and output load (LOAD) listed in Table 1 below:

Table 1

INPUT RAMP	LOAD
.25 nS	1 pF
.25 nS	2 pF
1 nS	2 pF

For the first two simulations input voltage (VINPUT) is given by the following Formula 9:

Formula 9

$VINPUT = 18.2 * TIME \quad$ For TIME < .25 nS

$VINPUT = 4.55 \quad$ For TIME > .25 nS

Similarly, for the third simulation, input voltage (VINPUT) is given by the following Formula 10:

Formula 10

$VINPUT = 4.55 * TIME \quad$ For TIME < 1 nS

$VINPUT = 4.55 \quad$ For TIME > 1 nS

In the above Formula 9 and Formula 10, TIME is measured from the beginning of the INPUT RAMP and VINPUT is measured in volts.

For each of the simulations, a linear approximation is made of the output of the simulated circuit. The linearization is done by a two point fit through times at which the output voltage is at 3.15 volts and at 1.4 volts. In the simulation, the results of the linearization are given by Table 2 below:

Table 2

INPUT RAMP	LOAD	Time when OUTPUT = 3.15 V	Time when OUTPUT = 1.4 V
.25 nS	1 pF	1.027 nS	2.107 nS
.25 nS	2 pF	1.806 nS	3.911 nS
1 nS	2 pF	2.289 nS	4.394 nS

Using linear approximation for the first simulation gives an output voltage (VOUTPUT) for the first simulation as given by Formula 11 below:

Formula 11

$$\begin{aligned} &VOUTPUT = 4.55 && \text{For TIME} < .163 \text{ nS} \\ &VOUTPUT = 4.55 - 4.55 * (TIME - .163)/(2.971 - .163) && \\ & && \text{For } .163 \text{ nS} < TIME < 2.971 \text{ nS} \\ &VOUTPUT = 0 && \text{For TIME} > 2.971 \text{ nS} \end{aligned}$$

Using linear approximation for the second simulation gives an output voltage (VOUTPUT) for the second simulation as given by Formula 12 below:

Formula 12

$$\begin{aligned} &VOUTPUT = 4.55 && \text{For TIME} < .122 \text{ nS} \\ &VOUTPUT = 4.55 - 4.55 * (TIME - .122)/(5.595 - .122) && \\ & && \text{For } .122 \text{ nS} < TIME < 5.595 \text{ nS} \\ &VOUTPUT = 0 && \text{For TIME} > 5.595 \text{ nS} \end{aligned}$$

Using linear approximation for the third simulation gives an output voltage (VOUTPUT) for the third simulation as given by Formula 13 below:

Formula 13

$$\begin{aligned} &VOUTPUT = 4.55 && \text{For TIME} < 1.251 \text{ nS} \\ &VOUTPUT = 4.55 - 4.55 * (TIME - 1.251)/(6.724 - 1.251) && \\ & && \text{For } 1.251 \text{ nS} < TIME < 6.724 \text{ nS} \\ &VOUTPUT = 0 && \text{For TIME} > 6.724 \text{ nS} \end{aligned}$$

Values for DT_0 and DT_1 for the three simulations are extracted from the linear approximations as listed in Table 3 below:

Table 3

INPUT RAMP	LOAD	DT_0	DT_1
.25 nS	1 pF	.163 nS	2.971 nS
.25 nS	2 pF	.122 nS	5.595 nS
1 nS	2 pF	1.251 nS	6.724 nS

Now the values for DT_0 and DT_1 may be substituted into Formula 4 and Formula 5 to solve for coefficients

A_0, A_1, B, D_0 and D_1. For example, substituting the values for INPUT RAMP, LOAD and DT_0 for the three simulations into Formula 4 gives the following equations:

$$.163\ nS = A_0 + B * (.25\ nS) + D_0 * (1\ pF)$$
$$.122\ nS = A_0 + B * (.25\ nS) + D_0 * (2\ pF)$$
$$1.251\ nS = A_0 + B * (1.0\ nS) + D_0 * (2\ pF)$$

These three equations are solved for A_0, B, and D_0 to produce the following values:

$A_0 = .043$
$B = .644$
$D_0 = -.041$

Similarly, substituting the values for INPUT RAMP, LOAD and DT_1 for the three simulations into Formula 5 gives the following equations:

$$2.971\ nS = A_1 + B * (.25\ nS) + D_1 * (1\ pF)$$
$$5.595\ nS = A_1 + B * (.25\ nS) + D_1 * (2\ pF)$$
$$6.724\ nS = A_1 + B * (1.0\ nS) + D_1 * (2\ pF)$$

These three equations are solved for A_1, B, and D_1 to produce the following values:

$A_1 = .186$
$B = .644$
$D_1 = 2.624$

The first consistency check is done by noting that the coefficient B has the same value for the solutions of both Formula 4 and Formula 5. If two different values of B had been obtained, it would imply that at least one of the original simulations was not in the fast ramp region. The simulations would have to be redone with faster input ramps and/or larger output loads.

Now that the coefficients have been determined CRITINRAMP may be calculated from Formula 8. For example, Formula 8 is simplified as below (remember that at CRITINRAMP: INPUT RAMP = CRITINRAMP):

$$CRITINRAMP = 1/2 * (DT_0 + DT_1)$$
$$CRITINRAMP = 1/2 * DT_0 + 1/2 * DT_1$$
$$CRITINRAMP = 1/2 * [A_0 + B * (INPUT\ RAMP) + D_0 * (LOAD)] + 1/2 * [A_1 + B * (INPUT\ RAMP) + D_1 * (LOAD)]$$
$$CRITINRAMP = 1/2 * (A_0 + A_1) + 1/2 * (D_0 + D_1) * (LOAD) + B * (CRITINRAMP)$$
$$CRITINRAMP = 1/2 * [A_0 + A_1 + (D_0 + D_1) * (LOAD)]/(1 - B)$$

Using the values for the coefficients A_0, A_1, B, D_0 and D_1 determined above, CRITINRAMP has the value given in Formula 14 below:

Formula 14 $$CRITINRAMP = .322 + 3.628 * (LOAD)$$

The second consistency check is done by checking to see whether the three simulations fall within fast input ramp region 174. For the two values for loads used in the three simulations, CRITINRAMP is given by Table 4 below:

Table 4

LOAD	CRITINRAMP
1 pF	3.950 nS
2 pF	7.578 nS

The first simulation was done with a 1 pF load and an input ramp of .25 nS. Table 4 indicated this is within fast input ramp region 174. The second and third simulations were done with a 2 pF load and input rams of .25 nS and 1 nS, respectively. Table 4 indicates these are also within fast input ramp region 174.

An additional circuit simulation of DC transfer curves and linear approximations is made to obtain the proportionality constant Z. In the circuit simulation, DC input voltages are found which cause output voltage to be 3.15 volts and 1.4 volts. The circuit simulation yields the values given in Table 5 below:

Table 5

Output Voltage	Input Voltage
3.15 V	2.176 V
1.40 V	2.298 V

Using linear approximations, the DC transfer curve is calculated as given in Formula 15 below:

Formula 15

$$VOUTPUT = 4.55 \qquad \text{For } VINPUT < 2.078$$

$$VOUTPUT = 4.55 - 4.55*(VINPUT - 2.078)/(2.396\ 2.078) \qquad \text{For } 2.078 < VINPUT < 2.396$$

$$VOUTPUT = 0 \qquad \text{For } VINPUT > 2.396$$

In the slow ramp region, a plot of DT_0 versus the input ramp is a straight line for a given ouput load. Z is the slope of that line and can be calculated from the DC transfer curve. In determining proportionality constant Z it is helpful to define the following breakpoints. Breakpoint V + is the value for VINPUT at which VOUTPUT reaches 0. For the present example, from Formula 15, it is seen that breakpoint V+ is 2.078 volts and breakpoint V- is 2.396. Thus, proportionality constant Z may be calculated from Table 6 below:

Table 6

VINPUT	VOUTPUT	Z
RISING	RISING	$(V^- - 0) / (4.55 - 0)$
RISING	FALLING	$(V^+ - 0) / (4.55 - 0)$
FALLING	RISING	$(4.55 - V^-) / (4.55 - 0)$
FALLING	FALLING	$(4.55 - V^+) / (4.55 - 0)$

In the given example, the logic device is an inverter with input rising. Therefore, VINPUT is rising and VOUTPUT is falling. From Table 6, the value of proportionality constant is therefore:

$$Z = 2.078/4.55 = .457$$

In the logic device library for the logic device, the values for coefficients A_0, A_1, B, D_0 and D_1 are stored. Additionally, the value for CRITINRAMP from Formula 14 is stored and four values calculate from Table 6 are stored for proportionality constant Z. When calculating propagation delay through a first logic device, DT_0 of the first logic device is equal to the beginning of an input ramp for a second logic device following the first logic device. Similarly the value $(DT_1 - DT_0)$ for the first logic device is equal to the input ramp for the second logic device. By determining DT_0 and DT_1 for each logic device and by using the value $(DT_1 - DT_0)$ as an input ramp for a next logic device, the total propagation delay through circuitry is accomplished

In the above example, the linear approximations assumed perfectly linear inputs. However, tails in the input waveform were not accounted for. The tail in the input waveform has a noticeable effect in fast input region 174; however, the tail in the input waveform has a negligible effect in slow input ramp region 174. To take account of this second order effect, two factors to adjust for the tails (FAT) may be included, one for rising inputs and one for falling inputs. To take these into account, Formula 4 and Formula 5 are modified resulting, respectively, in Formula 16 and Formula 17 given below:

Formula 16 $\quad DT_0 = A_0 + FAT * B * (INPUT\ RAMP) + D_0 * (LOAD)$

Formula 17 $\quad DT_1 = A_1 + FAT * B * (INPUT\ RAMP) + D_1 * (LOAD)$

As in the above example, the value FAT may be determined by using circuit simulations. Figure 7 shows a model used for circuit simulations for the second order model. Timing delay is modeled for an inverter 30. A capacitor 36 connected between an ouput 35 of inverter 30 and a ground 37 represents output load of inverter 30. A capacitor 34 connected between an input 33 of inverter 30 and ground 37 represents input load of inverter 30. An inverter 32 is used in order to place on input 33 of inverter 30 a realistic voltage, that is an input voltage with a tail.

Three simulations are performed using a falling (linear) input on an input 31 of inverter 32 with a ramp of .1 nS. The simulations are performed as described in the prior examples, with the exception each of the three simulations includes a simulation of both inverter 32 and inverter 30. In a first simulation, capacitance 34 (Cint) has a value of 0 pF and capacitance 35 (Cout) has a value of 1 pF. The first simulation of inverter 32 and linear approximation yields the following voltage (Vint) at input 33 of inverter 30, shown by Formula 18 below:

Formula 18

$$Vint = 0 \qquad \text{For TIME} < .064 \text{ nS}$$

$$Vint = 4.55*(TIME - .064)/(.396 - .064) \qquad \text{For } .064 \text{ nS} < \text{TIME} < .395 \text{ nS}$$

$$Vint = 4.55 \qquad \text{For TIME} > .395 \text{ nS}$$

The first simulation of inverter 30 and linear approximation yields the following voltage (Vout) at output 35 of inverter 30, shown by Formula 19 below:

Formula 19

$$Vout = 4.55 \qquad \text{For TIME} < .311 \text{ nS}$$

$$Vout = 4.55 - 4.55 * (TIME - .311)/(3.126 - .311) \qquad \text{For } .311 \text{ nS} < \text{TIME} < 3.126 \text{ nS}$$

$$Vout = 0 \qquad \text{For TIME} > 3.126 \text{ nS}$$

In Formula 18 and Formula 19, TIME is measured from the beginning of the ramp on input 31 of inverter 32. Inverter 30, therefore, on input 33 sees an input transition that starts at .064 nS and ends at .395 nS. The output transition on output 35 of inverter 30 starts at .311 nS and ends at 3.126 nS. Thus the input ramp on input 33 of inverter 30 is .395 minus .064, that is .331 nS. Using the Formula 2, DT_0 is seen to be .311 minus .064, that is, .247 nS. Using Formula 3, DT_1 is seen to be 3.126 minus .064, that is, 3.062 nS.

Table 7 below summarizes the results of all three simulations:

Table 7

Cint	Cout	INPUT RAMP	DT_0	DT_1
0 pF	1 pF	.331 nS	.247 nS	3.062 nS
0 pF	2 pF	.331 nS	.206 nS	5.686 nS
.2 pF	2 pF	.826 nS	.826 nS	6.040 nS

The above values are used to solve for the coefficients a_0, a_1, b, d_0 and d_1 in the following following Formula 20 and Formula 21:

Formula 20 $\quad DT_0 = a_0 + b * (\text{INPUT RAMP}) + d_0 * (\text{Cout})$

Formula 21 $\quad DT_1 = d_1 + b * (\text{INPUT RAMP}) + d_1 * (\text{Cout})$

Formula 20 and Formula 21 uses the coefficients a_0, a_1, b, d_0 and d_1 because the values of the coefficients sometimes vary from the values of the coefficients A_0, A_1, B, D_0 and D_1 used in the first order model. As above, the values in Table 7 provide for three sets of equations for Formula 20 and Formula 21. Solving the three equations yields the following values:

$a_0 = .051$
$a_1 = .201$
$b = .715$
$d_0 = -.041$
$d_1 = 2.624$

When these are compared with the values for A_0, A_1, B, D_0 and D_1 calculated above, it is seen that the values for d_0 and d_1 are equivalent to the values for D_0 and D_1, respectively. The values for a_0 and a_1 vary slightly from the values for A_0 and A_1, respectively. The most significant difference is between the value for b and the value for B. The value FAT is obtained by the ratio of b to B, as shown by Formula 22 below:

Formula 22 $\quad FAT = b/B$

$= .715/.644$

$= 1.11$

Once calculated, the value for FAT is stored in the logic device library along with the values for the coefficients A_0, A_1, B, D_0 and D_1. The value for FAT is dependent on the process technology and does not depend on the

logic design of a logic cell. Therefore, a single entry for the value of FAT may be obtained from characterization of a single logic cell and stored for use for all logic cells in a cell library.

The value for CRITINRAMP for the second order model may be recalculated from Formula 8, to be as set out in Formula 23 below:

Formula 23 $$CRITINRAMP = 1/2 * [A_0 + A_1 + (D_0 + D_1) * (LOAD)]/(1 - FAT * B)$$

Using the values for the coefficients A_0, A_1, B, D_0, D_1 and FAT determined above, CRITINRAMP has the value given in Formula 24 below:

Formula 24 $$CRITINRAMP = .402 + 4.532 * (LOAD)$$

Using the second order model, the following Formula 25 and Formula 26 are used to calculate DT_0 and DT_1 in slow input ramp region 175:

Formula 25 $$DT_0 = DT_{0(CRITINRAMP)} + Z * (INPUT\ RAMP - CRITINRAMP)$$

Formula 26 $$DT_1 = DT_{1(CRITINRAMP)} + B * (INPUT\ RAMP - CRITINRAMP)$$

In Formula 25, $DT_{0(CRITINRAMP)}$ is the value of DT_0 at CRITINRAMP. In Formula 26, $DT_{1(CRITINRAMP)}$ is the value of DT_1 at CRITINRAMP.

Once the logic device has been characterized, as above, it is possible to determine a value for DT_0 and DT_1 for any input ramp and load. For example, suppose the load (LOAD) is equal to .5 pF and the input ramp is equal to .25 nS. Then, using Formula 24, CRITINRAMP has the following value:

$$CRITINRAMP = .402 + 4.532 * (LOAD)$$
$$CRITINRAMP = .402 + 4.532 * (.5)$$
$$CRITINRAMP = 2.668\ nS$$

Since the input ramp (.25 nS) is smaller than the CRITINRAMP (2.668 nS), the example falls in fast input ramp region 174. Therefore, Formula 16 is used to calculate DT_0 and Formula 17 is used to calculate DT_1, as shown below:

$$DT_0 = A_0 + FAT * B * (INPUT\ RAMP) + D_0 * (LOAD)$$
$$DT_0 = .043 + 1.11 * .644 * (.25) - .041 * (.5)$$
$$DT_0 = .201\ nS$$
$$DT_1 = A_1 + FAT * B * (INPUT\ RAMP) + D_1 * (LOAD)$$
$$DT_1 = .186 + 1.11 * .644 * (.25) + 2.624 * (.5)$$
$$DT_1 = 1.677\ nS$$

As an alternate example, suppose the load (LOAD) is equal to .5 pF and the input ramp is equal to 3 nS. Then CRITINRAMP still has the following value:

$$CRITINRAMP = .402 + 4.532 * (LOAD)$$
$$CRITINRAMP = .402 + 4.532 * (.5)$$
$$CRITINRAMP = 2.668\ nS$$

Since the input ramp (3 nS) is now larger than the CRITINRAMP (2.668 nS), the example falls in slow input ramp region 175. Therefore, Formula 25 is used to calculate DT_0 and Formula 26 is used to calculate DT_1. First $DT_{0(CRITINRAMP)}$ and $DT_{1(CRITINRAMP)}$ are calculated as shown below:

$$DT_{0(CRITINRAMP)} = A_0 + FAT * B * (CRITINRAMP) + D_0 * (LOAD)$$
$$DT_{0(CRITINRAMP)} = .043 + 1.11 * .644 * (2.668) - .041 * (.5)$$
$$DT_{0(CRITINRAMP)} = 1.930\ nS$$
$$DT_{1(CRITINRAMP)} = A_1 + FAT * B * (CRITINRAMP) + D_1 * (LOAD)$$
$$DT_{1(CRITINRAMP)} = .186 + 1.11 * .644 * (2.668) + 2.624 * (.5)$$
$$DT_{1(CRITINRAMP)} = 3.406\ nS$$

Once $DT_{0(CRITINRAMP)}$ and $DT_{1(CRITINRAMP)}$ have been calculated, Formula 25 is used to calculate DT_0 and Formula 26 is used to calculate DT_1, as shown below:

$$DT_0 = DT_{0(CRITINRAMP)} + Z * (INPUT\ RAMP - CRITINRAMP)$$
$$DT_0 = 1.930 + .457 * (3 - 2.668)$$
$$DT_0 = 2.082\ nS$$
$$DT_1 = DT_{1(CRITINRAMP)} + B * (INPUT\ RAMP - CRITINRAMP)$$
$$DT_1 = 3.406 + .644 * (3 - 2.668)$$
$$DT_1 = 3.620\ nS$$

As discussed above the input ramp for the following logic device is equal to $DT_1 - DT_0$.

The foregoing discussion discloses and describes merely exemplary methods and embodiments of the present invention. As will be understood by those familiar with the art, the invention may be embodied in other specific forms. For example, the present invention may be implemented by a computer.

* C L M S *

Claims

1. A method running on a computer for approximating propagation delay through a logic device (30), the method comprising the steps of:
   (a) dividing operation of the logic device (30) into a fast input ramp region (174), wherein an input ramp (251) to the logic device (30) is completely switched before the output signal wave form (252) changes significantly, and a slow input ramp region (175), wherein the input ramp (281) switches so slowly that the output signal (282) switches while the input ramp (281) is still switching, a boundary (173) between the fast input ramp region (174) and the slow input ramp region (175) being based on duration of input ramp to the logic device (30) and amount of capacitive load (36) driven by the logic device (30);
   (b) when the logic device (30) operates in the fast input ramp region (174), using a first formula to obtain a first value representing delay through the logic device (30), the first formula varying the first value based on the duration of the input ramp to the logic device (30) and the capacitive load (36) driven by the logic device (30); and;
   (c) when the logic device (30) operates in the slow input ramp region (175), using a second formula to obtain the first value, the second formula varying the first value based on the duration of the input ramp to the logic device (30) and the capacitive load (36) driven by the logic device (30).

2. A method as in claim 1 wherein the first formula has a form as follows:
$$DT = A + B * (\text{INPUT RAMP}) + D * (\text{LOAD})$$
where DT represents the first value, INPUT RAMP represents duration of the input ramp, LOAD represents capacitive load (36) driven by the logic device (30), and A, B and D are coefficients determined using circuit simulations.

3. A method as in claim 2 wherein the second formula has a form as follows:
$$DT = A + B * (\text{CRITINRAMP}) + D * (\text{LOAD}) + Z * (\text{INPUT RAMP} - \text{CRITINRAMP})$$
wherein CRITINRAMP represents INPUT RAMP at the border between the fast input ramp region (174) and the slow input ramp region (175) for the LOAD, and Z is a determined using circuit simulations

4. A method as in claim 1 wherein the first formula has a form as follows:
$$DT = A + FAT * B * (\text{INPUT RAMP}) + D * (\text{LOAD})$$
wherein DT represents the first value, INPUT RAMP represents duration of the input ramp, LOAD represents capacitive load (36) driven by the logic device (30), and A, B, D and FAT are coefficients determined using circuit simulations.

5. A method as in claim 4 wherein the second formula has a form as follows:
$$DT = DT_{(\text{CRITINRAMP})} + Z * (\text{INPUT RAMP} - \text{CRITINRAMP})$$
wherein CRITINRAMP represents INPUT RAMP at the border between the fast input ramp region (174) and the slow input ramp region (175) for the LOAD, Z is a coefficient determined using circuit simulations, and $DT_{(\text{CRITINRAMP})}$ is the first value when the logic ramp has an input ramp equal to CRITINRAMP and a load equal to LOAD.

6. A method as in claim 1 wherein the method includes calculating a next input ramp for a next logic device (30) coupled to an output of the logic device (30), the method additionally comprising the steps of:
   (d) when the logic device (30) operates in the fast input ramp region (174), using a third formula to generate a second value representing duration from beginning of the input ramp of the logic device (30) to an end of an output ramp of the logic device (30), the second value being based on the length of the input ramp to the logic device (30) and the amount of the load driven by the logic device (30); and;
   (e) when the logic device (30) operates in the slow input ramp region (175), using a fourth formula to generate the second value; and,
   (f) subtracting the first value from the second value to obtain the next input ramp.

7. A method as in claim 6 wherein the first formula has a form as follows:
$$DT_0 = A_0 + B * (\text{INPUT RAMP}) + D_0 * (\text{LOAD})$$
where DT_0 represents the first value, INPUT RAMP represents duration of the input ramp, LOAD represents capacitive load (36) driven by the logic device (30), and A_0, B and D_0 are coefficients determined using circuit simulations.

* C L M S *

8. A method as in claim 7 wherein the third formula has a form as follows :

$$DT_1 = A_1 + B * (\text{INPUT RAMP}) + D_1 * (\text{LOAD})$$

where DT_1 represents the second value, and A_1 and D_1 are coefficients determined using circuit simulations.

9. A method as in claim 8 wherein the second formula has a form as follows:

$$DT_0 = A_0 + B * (\text{CRITINRAMP}) + D_0 * (\text{LOAD}) + Z * (\text{INPUT RAMP} - \text{CRITINRAMP})$$

wherein CRITINRAMP represents INPUT RAMP at the border between the fast input ramp region (174) and the slow input ramp region (175) for the LOAD, and Z is determined using circuit simulations.

10. A method as in claim 9 wherein the fourth formula has a form as follows:

$$DT_1 = A_1 + B * (\text{INPUT RAMP}) + D_1 * (\text{LOAD}).$$

11. A method as in claim 6 wherein the first formula has a form as follows:

$$DT_0 = A_0 + FAT * B * (\text{INPUT RAMP}) + D_0 * (\text{LOAD})$$

wherein DT_0 represents the first value, INPUT RAMP represents duration of the input ramp, LOAD represents capacitive load (36) driven by the logic device (30), and A_0, B, D_0 and FAT are coefficients determined using circuit simulations.

12. A method as in claim 11 wherein the third formula has a form as follows:

$$DT_1 = A_1 + FAT * B * (\text{INPUT RAMP}) + D_1 * (\text{LOAD})$$

wherein DT_1 represents the second value, and A_1 and D_1 are coefficients determined using circuit simulations.

13. A method as in claim 12 wherein the second formula has a form as follows:

$$DT_0 = DT_{0(\text{CRITINRAMP})} + Z * (\text{INPUT RAMP} - \text{CRITINRAMP})$$

wherein CRITINRAMP represents INPUT RAMP at the border between the fast input.ramp region (174) and the slow input ramp region (175) for the LOAD, Z is a coefficient determined using circuit simulations, and $DT_{(\text{CRITINRAMP})}$ is the first value when the logic ramp has an input ramp equal to CRITINRAMP and a load equal to LOAD.

14. A method as in claim 13 wherein the fourth formula has a form as follows:

$$DT_1 = DT_{1(\text{CRITINRAMP})} + B * (\text{INPUT RAMP} - \text{CRITINRAMP}).$$

15. A method as in claim 6 wherein in step (a) the boundary (173) between the fast input ramp region (174) and the slow input ramp region (175) occurs where for each value of the capacitive load (36), the output ramp is one half complete when the input ramp is complete.

16. A method as in claim 1 wherein in step (a) the boundary (173) between the fast input ramp region (174) and the slow input ramp region (175) occurs where for each value of the capacitive load (36), an output ramp for the logic device (30) is one half complete when the input ramp is complete.

Patentansprüche

1. Auf einem Rechner ablaufendes Verfahren zur näherungsweisen Berechnung der Ausbreitungsverzögerung durch einen Logikbaustein (30), wobei das Verfahren folgende Schritte umfaßt:
   (a) Aufteilen des Betriebs des Logikbausteins (30) in einen schnellen Eingangsrampenbereich (174), innerhalb dem eine an den Logikbaustein (30) angelegte Eingangsrampe (251) vollständig umgeschaltet wird, bevor sich die Ausgangssignal-Kurvenform wesentlich ändert, und einen langsamen Eingangsrampenbereich (175), innerhalb dem die Eingangsrampe (281) so langsam umschaltet, daß das Ausgangssignal (282) umschaltet, während die Eingangsrampe (281) immer noch umschaltet, wobei ein Grenzbereich (173) zwischen dem schnellen Eingangsrampenbereich (174) und dem langsamen Eingangsrampenbereich (175) auf der Dauer der an den Logikbaustein (30) angelegten Eingangsrampe und der Größe einer durch den Logikbaustein (30) angesteuerten kapazitiven Last (36) basiert;
   (b) Verwenden einer ersten Formel, wenn der Logikbaustein (30) in dem schnellen Eingangsrampenbereich (174) arbeitet, zum Berechnen eines ersten Werts, der eine Ausbreitungsverzögerung durch den Logikbaustein (30) darstellt, wobei die erste Formel den ersten Wert entsprechend der Dauer der

an den Logikbaustein (30) angelegten Eingangsrampe und der kapazitiven Last (36), die von dem Logikbaustein (30) angesteuert wird, ändert; und

(c) Verwenden einer zweiten Formel, wenn der Logikbaustein (30) in dem langsamen Eingangsrampenbereich (175) arbeitet, zur Berechnung des ersten Werts, wobei die zweite Formel den ersten Wert entsprechend der Dauer der an den Logikbaustein (30) angelegten Eingangsrampe und der kapazitiven Last (36), die von dem Logikbaustein (30) angesteuert wird, ändert.

2. Verfahren nach Anspruch 1, wobei die erste Formel folgende Form aufweist:

$$DT = A + B * (EINGANGSRAMPE) + D * (LAST)$$

wobei DT für den ersten Wert steht, EINGANGSRAMPE für die Dauer der Eingangsrampe, LAST für die kapazitive Last (36), die von dem Logikbaustein (30) angesteuert wird, und A, B und D Koeffizienten sind, die durch Schaltungssimulationen bestimmt werden.

3. Verfahren nach Anspruch 2, wobei die zweite Formel die folgende Form aufweist:

$$DT = A + B * (KRITEINRAMPE) + D * (LAST) + Z * (EINGANGSRAMPE - KRITEINRAMPE)$$

wobei KRITEINRAMPE für EINGANGSRAMPE an der Grenze zwischen dem schnellen Eingangsrampenbereich (174) und dem langsamen Eingangsrampenbereich (175) für die LAST steht, und Z durch Schaltungssimulationen bestimmt wird.

4. Verfahren nach Anspruch 1, wobei die erste Formel folgende Form aufweist:

$$DT = A + FAT * B * (EINGANGSRAMPE) + D * (LAST)$$

wobei DT für den ersten Wert steht, EINGANGSRAMPE für die Dauer der Eingangsrampe, LAST für die kapazitive Last (36), die von dem Logikbaustein (30) angesteuert wird, und A, B, D und FAT Koeffizienten sind, die durch Schaltungssimulationen bestimmt werden.

5. Verfahren nach Anspruch 4, wobei die zweite Formel folgende Form aufweist:

$$DT = DT_{(KRITEINRAMPE)} + Z * (EINGANGSRAMPE - KRITEINRAMPE)$$

wobei KRITEINRAMPE für die Eingangsrampe an der Grenze zwischen dem schnellen Eingangsrampenbereich (174) und dem langsamen Eingangsrampenbereich (175) für die LAST steht, Z ein Koeffizient ist, der durch Schaltungssimulationen bestimmt wird, und $DT_{(KRITEINRAMPE)}$ der erste Wert ist, falls der Logikbaustein eine Eingangsrampe aufweist. die gleich KRITEINRAMPE ist, und eine Last aufweist, die gleich LAST ist.

6. Vefahren nach Anspruch 1, wobei das Verfahren die Berechnung einer nächsten Eingangsrampe für einen nächsten Logikbaustein (30) einschließt, der an den Ausgang des Logikbaustein (30) angeschlossen ist, wobei das Verfahren zusätzlich folgende Schritte umfaßt:

(d) Verwenden einer dritten Formel, wenn der Logikbaustein (30) in dem schnellen Eingangsrampenbereich (174) arbeitet, zur Erzeugung eines zweiten Werts, der die Zeitdauer vom Beginn der Eingangsrampe des Logikbausteins (30) bis zum Ende einer Ausgangsrampe des Logikbausteins (30) darstellt, wobei der zweite Wert auf der Länge der an den Logikbaustein (30) angelegten Eingangsrampe und der Größe der kapazitiven Last, die vom Logikbaustein (30) angesteuert wird, basiert; und

(e) Verwenden einer vierten Formel zum Erzeugen des zweiten Werts, wenn der Logikbaustein (30) in dem langsamen Eingangsrampenbereich (175) arbeitet; und

(f) Subtrahieren des ersten Werts von dem zweiten Wert, um die nächste Eingangsrampe zu erhalten.

7. Verfahren nach Anspruch 6, wobei die erste Formel folgende Form aufweist:

$$DT_0 = A_0 + B * (EINGANGSRAMPE) + D_0 * (LAST)$$

wobei DT_0 für den ersten Wert steht, EINGANGSRAMPE für die Dauer der Eingangsrampe, LAST für die kapazitive Last (36), die von dem Logikbaustein (30) angesteuert wird, und A_0, B und D_0 Koeffizienten sind, die durch Schaltungssimulationen bestimmt werden.

8. Verfahren nach Anspruch 7, wobei die dritte Formel folgende Form aufweist:

$$DT_1 = A_1 + B * (EINGANGSRAMPE) + D_1 * (LAST)$$

wobei DT_1 für den zweiten Wert steht und A_1 und D_1 Koeffizienten sind, die durch Schaltungssimulationen bestimmt werden.

9. Verfahren nach Anspruch 8, wobei die zweite Formel folgende Form aufweist:

$$DT_0 = A_0 + B * (KRITEINRAMPE) + D_0 * (LAST) + Z * (EINGANGSRAMPE - KRITEINRAMPE)$$

* C L M S *

wobei KRITEINRAMPE für EINGANGSRAMPE im Grenzbereich zwischen dem schnellen Eingangsrampenbereich (174) und dem langsamen Eingangsrampenbereich (175) für die LAST steht, und Z durch Schaltungssimulationen bestimmt wird.

10. Verfahren nach Anspruch 9, wobei die vierte Formel folgende Form aufweist:

$$DT_1 = A_1 + B * (EINGANGSRAMPE) + D_1 * (LAST)$$

11. Verfahren nach Anspruch 6, wobei die erste Formel folgende Form aufweist:

$$DT_0 = A_0 + FAT * B * (EINGANGSRAMPE) + D_0 * (LAST)$$

wobei DT_0 für den ersten Wert steht, EINGANGSRAMPE für die Dauer der Eingangsrampe, LAST für die kapazitive Last (36), die von dem Logikbaustein (30) angesteuert wird, und A_0 B, D_0 und FAT Koeffizienten sind, die durch Schaltungssimulationen bestimmt werden.

12. Verfahren nach Anspruch 11, wobei die dritte Formel folgende Form aufweist:

$$DT_1 = A_1 + FAT * B * (EINGANGSRAMPE) + D_1 * (LAST)$$

wobei DT_1 für den zweiten Wert steht, und A_1 und D_1 Koeffizienten sind, die durch Schaltungssimulationen bestimmt werden.

13. Verfahren nach Anspruch 12, wobei die zweite Formel folgende Form aufweist:

$$DT_0 = DT_{0(KRITEINRAMPE)} + Z * (EINGANGSRAMPE - KRITEINRAMPE)$$

wobei KRITEINRAMPE für EINGANGSRAMPE im Grenzbereich zwischen dem schnellen Eingangsrampenbereich (174) und dem langsamen Eingangsrampenbereich (175) für die LAST steht, Z ein Koeffizient ist, der durch Schaltungssimulationen bestimmt wird, und $DT_{(KRITEINRAMPE)}$ gleich dem ersten Wert ist, falls der Logikbaustein eine Eingangsrampe aufweist, die gleich KRITEINRAMPE ist, und eine Last aufweist, die gleich LAST ist.

14. Verfahren nach Anspruch 13, wobei die vierte Formel folgende Form aufweist:

$$DT_1 = DT_{1(KRITEINRAMPE)} + B * (EINGANGSRAMPE - KRITEINRAMPE)$$

15. Verfahren nach Anspruch 6, wobei in Schritt (a) der Grenzbereich (173) zwischen dem schnellen Eingangsrampenbereich (174) und dem langsamen Eingangsrampenbereich (175) auftritt, wenn für jeden Wert der kapazitiven Last (36) die Ausgangsrampe zur Hälfte vollständig ist, wenn die Eingangsrampe vollständig ist.

16. Verfahren nach Anspruch 1, wobei in Schritt (a) der Grenzbereich (173) zwischen dem schnellen Eingangsrampenbereich (174) und dem langsamen Eingangsrampenbereich (175) auftritt, wenn für jeden Wert der kapazitiven Last (36) eine Ausgangsrampe für den Logikbaustein (30) zur Häfte vollständig ist und wenn die Eingangsrampe vollständig ist.

Revendications

1. Procédé fonctionnant sur un ordinateur permettant d'approcher le temps de propagation par un dispositif logique (30), caractérisé en ce qu'il comprend les étapes suivantes :
   (a) la division du fonctionnement du dispositif logique (30) en un domaine de rampe d'entrée rapide (174), caractérisé par le fait qu'une rampe d'entrée (251) du dispositif logique (30) est entièrement commutée avant que la forme de l'onde du signal de sortie (252) change de façon importante, et un domaine de rampe d'entrée lente (175), caractérisé par le fait que la rampe d'entrée (281) commute si lentement que le signal de sortie (282) bascule lors de la commutation de la rampe d'entrée (281), la délimitation (178) entre le domaine de rampe d'entrée rapide (174) et le domaine de rampe d'entrée lente (175) étant basée sur la durée de rampe d'entrée du dispositif logique (30) et la charge capacitive (36) commandée par le dispositif logique (30) ;
   (b) lorsque le dispositif logique (30) fonctionne dans le domaine de rampe d'entrée rapide (174), l'utilisation d'une première formule permet d'obtenir une première valeur représentant le retard par le dispositif logique (30), la première formule faisant varier la première valeur basée sur la durée de la rampe d'entrée du dispositif logique (30) et sur la charge capacitive (36) commandée par le dispositif logique (30) ; et
   (c) lorsque le dispositif logique (30) fonctionne dans le domaine de rampe d'entrée lente (175), l'utili-

* C L M S *

sation d'une deuxième formule permet d'obtenir la première valeur, la deuxième formule faisant varier la première valeur basée sur la durée de la rampe d'entrée du dispositif logique (30) et la charge capacitive (36) commandée par le dispositif logique (30).

2. Un procédé selon la revendication 1, caractérisé en ce que la première formule a la forme suivante :

$$DT = A + B * (\text{RAMPE D'ENTRÉE}) + D * (\text{CHARGE})$$

où DT représente la première valeur, RAMPE D'ENTRÉE représente la durée de la rampe d'entrée, CHARGE représente la charge capacitive (36) commandée par le dispositif logique (30), A, B et D sont des coefficients déterminés au moyen de simulations de circuit.

3. Un procédé selon la revendication 2, caractérisé en ce que la deuxième formule a la forme suivante :

$$DT = A + B * (\text{RAMPE D'ENTRÉE CRIT}) + D * (\text{CHARGE}) + Z * (\text{RAMPE D'ENTRÉE} - \text{RAMPE D'ENTRÉE CRIT}),$$

caractérisé en ce que RAMPE D'ENTRÉE CRIT représente la RAMPE D'ENTRÉE à la limite entre le domaine de rampe rapide (174) et le domaine de rampe lente (175) de la CHARGE et Z est déterminée au moyen de simulations de circuit.

4. Un procédé selon la revendication 1, caractérisé en ce que la première formule a la forme suivante :

$$DT = A + FAT * B * (\text{RAMPE D'ENTRÉE}) + D * (\text{CHARGE})$$

caractérisé en ce que DT représente la première valeur, RAMPE D'ENTRÉE représente la durée de la rampe d'entrée, CHARGE représente la charge capacitive (36) commandée par le dispositif logique (30), A, B, D, et FAT sont des coefficients déterminés au moyen des simulations de circuit.

5. Un procédé selon la revendication 4, caractérisé en ce que la deuxième formule a la forme suivante :

$$DT = DT_{(\text{RAMPE D'ENTRÉE CRIT})} + Z * (\text{RAMPE D'ENTRÉE} - \text{RAMPE D'ENTRÉE CRIT}),$$

caractérisé en ce que RAMPE D'ENTRÉE CRIT représente la RAMPE D'ENTRÉE à la limite entre le domaine de rampe rapide (174) et le domaine de rampe lente (175) de la CHARGE et Z est un coefficient déterminé au moyen de simulations de circuit, et DT(RAMPE D'ENTRÉE CRIT) est la première valeur obtenue lorsque la rampe logique a une rampe d'entrée égale à RAMPE D'ENTRÉE CRIT et une charge égale à CHARGE.

6. Un procédé selon la revendication 1, caractérisé en ce que le procédé comprend le calcul de la rampe d'entrée suivante du dispositif logique suivant (30) couplé à une sortie du dispositif logique (30), le procédé comprenant en outre les étapes suivantes :

(d) lorsque le dispositif logique (30) fonctionne dans le domaine de rampe d'entrée rapide (174), l'utilisation d'une troisième formule permet de générer une deuxième valeur représentant la durée du début d'une rampe d'entrée du dispositif logique (30) à la fin d'une rampe de sortie du dispositif logique (30), la deuxième valeur étant basée sur la durée de la rampe d'entrée du dispositif logique (30) et la charge commandé par le dispositif logique (30) ; et

(e) lorsque le dispositif logique (30) fonctionne dans le domaine de rampe d'entrée lente (175), l'utilisation d'une quatrième formule permet de générer la deuxième valeur ; et

(f) la soustraction de la première valeur à la deuxième valeur permet d'obtenir la rampe d'entrée suivante ;

7. Un procédé selon la revendication 6, caractérisé en ce que la première formule a la forme suivante :

$$DT_0 = A_0 + B * (\text{RAMPE D'ENTRÉE}) + D_0 * (\text{CHARGE})$$

où DT_0 représente la première valeur, RAMPE D'ENTRÉE représente la durée de la rampe d'entrée, CHARGE représente la charge capacitive (36) commandée par le dispositif logique (30), et A_0, B et D_0 sont les coefficients déterminés au moyen des simulations de circuit ;

8. Un procédé selon la revendication 7, caractérisé en ce que la troisième formule a la forme suivante :

$$DT_1 = A_1 + B * (\text{RAMPE D'ENTRÉE}) + D_1 * (\text{CHARGE})$$

où DT_1 représente la deuxième valeur et A_1 et D_1 sont les coefficients déterminés au moyen des simulations de circuit ;

9. Un procédé selon la revendication 8, caractérisé en ce que la deuxième formule a la forme suivante :

$$DT_0 = A_0 + B * (\text{RAMPE D'ENTRÉE CRIT}) + D_0 * (\text{CHARGE}) + Z * (\text{RAMPE D'ENTRÉE} - \text{RAMPE D'ENTRÉE CRIT})$$

caractérisé en ce que RAMPE D'ENTRÉE CRIT représente la RAMPE D'ENTRÉE à la limite entre le domaine de rampe d'entrée rapide (174) et le domaine de rampe d'entrée lente (176) de la CHARGE et Z est déterminé au moyen des simulations de circuit ;

10. Un procédé selon la revendication 9, caractérisé en ce que la quatrième formule a la forme suivante :

$$DT1 = A1 + B * (\text{RAMPE D'ENTRÉE}) + D1 * (\text{CHARGE})$$

11. Un procédé selon la revendication 6, caractérisé en ce que la première formule a la forme suivante :

$$DT_0 = A_0 + FAT * B * (\text{RAMPE D'ENTRÉE}) + D_0 * (\text{CHARGE})$$

caractérisé en ce que DT_0 représente la première valeur, RAMPE D'ENTRÉE représente la durée de la rampe d'entrée, CHARGE représente la charge capacitive (36) commandée par le dispositif logique (30), et A_0, B, D_0 et FAT sont les coefficients déterminés au moyen des simulations de circuit ;

12. Un procédé selon la revendication 11, caractérisé en ce que la troisième formule a la forme suivante :

$$DT_1 = A_1 + FAT * B * (\text{RAMPE D'ENTRÉE}) + D_1 * (\text{CHARGE})$$

caractérisé en ce que DT_1 représente la deuxième valeur et A_1 et D_1 sont les coefficients déterminés au moyen des simulations de circuit ;

13. Un procédé selon la revendication 12, caractérisé en ce que la deuxième formule a la forme suivante :

$$DT_0 = DT_0(\text{RAMPE D'ENTRÉE CRIT}) + Z * (\text{RAMPE D'ENTRÉE} - \text{RAMPE D'ENTRÉE CRIT}),$$

caractérisé en ce que RAMPE D'ENTRÉE CRIT représente la RAMPE D'ENTRÉE à la limite entre le domaine de rampe rapide (174) et le domaine de rampe lente (175) de la CHARGE et Z est un coefficient déterminé au moyen de simulations de circuit, et DT(RAMPE D'ENTRÉE CRIT) est la première valeur obtenue lorsque la rampe logique a une rampe d'entrée égale à RAMPE D'ENTRÉE CRIT et une charge égale à CHARGE.

14. Un procédé selon la revendication 13, caractérisé en ce que la quatrième formule a la forme suivante :

$$DT_1 = DT_1(\text{RAMPE D'ENTRÉE CRIT}) + B * (\text{RAMPE D'ENTRÉE} - \text{RAMPE D'ENTRÉE CRIT})$$

15. Un procédé selon la revendication 6, caractérisé par le fait que, dans l'étape (a), la délimitation (173) entre le domaine de rampe d'entrée rapide (174) et le domaine de rampe d'entrée lente (175) se produit lorsque, pour chaque valeur de la charge capacitive (38), la rampe de sortie est à moitié complète au terme de la rampe d'entrée.

16. Un procédé selon la revendication 1, caractérisé par le fait que dans l'étape (a), la délimitation (173) entre le domaine de rampe d'entrée rapide (174) et le domaine de rampe d'entrée lente (175) se produit lorsque, pour chaque valeur de la charge capacitive (38), une rampe de sortie du dispositif logique (30) est à moitié complète au terme de la rampe d'entrée.