Fisnik Zeqiri 4306430
Felix Karg 4342014

9. Januar 2017

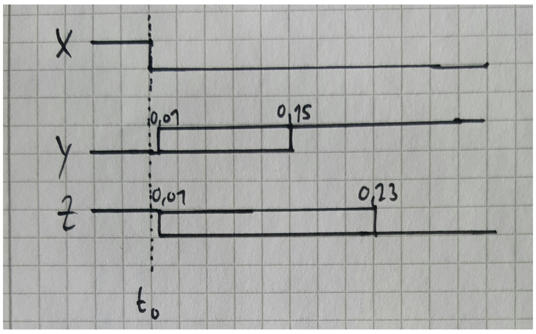
Antworten zum Übungsblatt Nr. 9

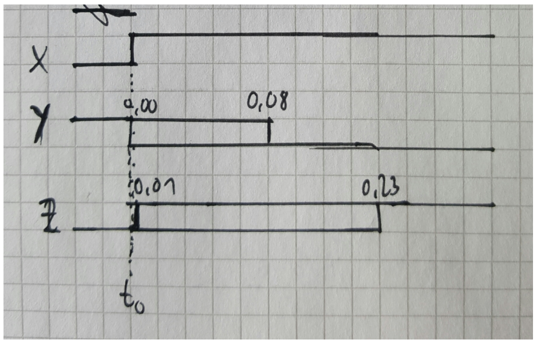
also erstmal kannst du mit "absoluter konvergenz" argumentieren .. wenn $\sum_{k=1}^{\infty} |a_k|$ konvergiert, dann konvergiert auch $\sum_{k=1}^{\infty} a_k$ .. damit musst du nicht mehr beweisen, dass $\sum_{k=1}^{\infty} \frac{(-1)^k}{k}$ konverrgiert, sondern nur noch, dass $\sum_{k=1}^{\infty} \frac{1}{k}$ konvergiert .. das sollte schonmal einfacher sein ^^

Aufgabe 1

a) x -> y: [0.01:0.15]; y -> z: [0.00:0.08] (0 -> 0)
   x -> y: [0.00:0.08]; y -> z: [0.01:0.15] (1 -> 1)
   maximale Zeit: 0.08 + 0.15 = 0.23 [ns].

b) [Bild 1] [Bild 2]





-> Immer zwischen [0.01:0.15] + [0.00:0.08] = [0.01:0.15] + [0.00:0.08] = [0.01:0.23].

Aufgabe 2

```
x = 0 -> 1
a = 1 -> 0
b = c = 0 -> 1
d = e = 1 -> 0
f = [0.02:0.14] + [0.01:0.07] + [0.02:0.14] = [0.07:0.35] (1 -> 0)
g = [0.02:0.14] + [0.01:0.07] + [0.02:0.14] = [0.07:0.35] (1 -> 0)
h = [0.02:0.14] + [0.01:0.07]               = [0.03:0.21] (0 -> 1)
i = [0.02:0.14] + [0.01:0.07] + [0.02:0.07] = [0.05:0.28] (0 -> 1)
j = [0.02:0.14]                             = [0.02:0.14] (1 -> 0)
k = [0.02:0.14] + [0.03:0.10] + [0.01:0.07] = [0.06:0.31] (0 -> 1)
l = [0.02:0.14] + [0.03:0.10]               = [0.05:0.24] (1 -> 0)
m = [0.02:0.14] + [0.03:0.10] + [0.03:0.10] = [0.08:0.34] (1 -> 0)
n = [0.02:0.14] + [0.03:0.10] + [0.03:0.10] = [0.08:0.34] (1 -> 0)

o = [max(min(f..n)):max(f..n)]+ [0.05:0.15] = [0.13:0.50] (0 -> 1)
p = [0.13:0.50] + [0.01:0.07]               = [0.14:0.57] (1 -> 0)
```

Bei Ausgang o und genauso bei p kann sogar ab 0.06 bzw 0.07 ns das korrekte Ergebnis ausgegeben sein, allerdings hatten zu diesem Zeitpunkt definitiv noch nicht alle gatter die Möglichkeit überhaupt das korrekte ergebnis zu liefern, selbst mit Minimalreaktionszeit. Es kann daher bei o und genauso bei p sein, dass bis 0.50 bzw. 0.57 ns respektive das Ergebnis mehrfach flackert.

Aufgabe 3

a) Ja, δ ist $\geq$ der Unterschied von t_2 bis t_4 (per Definition), also ist $t_2 + \delta \geq t_4$.

b) Nein, Da man über die verteilung von $\delta/2$ keine genaue Aussage treffen kann und im allgemeinen $t_3 < t_4$.

c) Ja, folgt transitiv aus $t_3 < t_4$ (-> a).

d) Ja, da $t_2 \leq t_3$ immer gilt.

e) Nein, siehe b.

f) Nur, falls $\delta \geq t_p$

g) Nein, Da $t_2 < t_3$, aber t_3 benötigt wäre. (1 δ)

h) Ja, da $(t_8 - t_3) = t_p$, sowie $t_p + 2 * \delta$ ein ganzer Spannungsverlauf ist, welcher alleine schon mindestens gleich t_{10} ist.

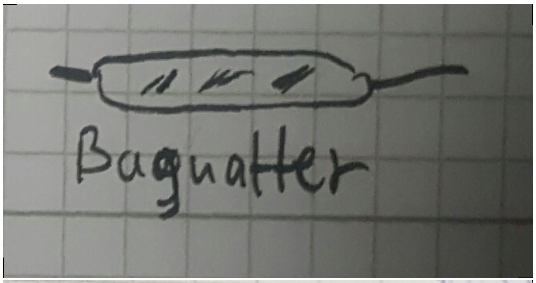
Aufgabe 4

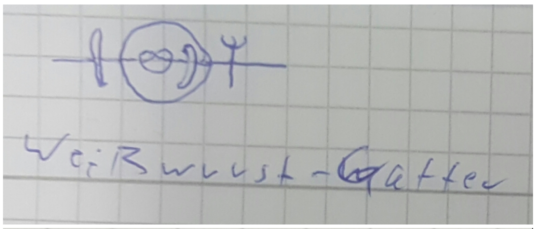
t_{HWD} ergibt sich dadurch, dass das NAND-Gatter spikefrei umschalten muss, wofür es nach dem Absenken von W noch 0.41 [ns] benötigt, damit D ohne sorgen umgeschaltet werden kann.

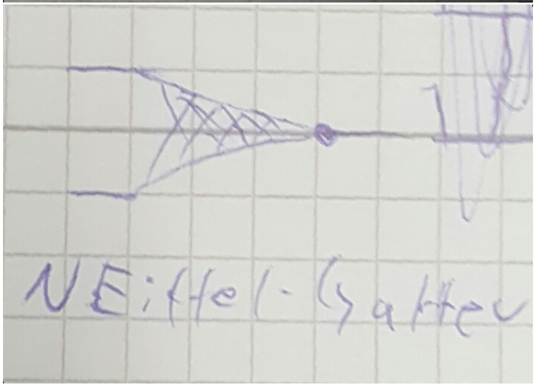
t_{SWD} ist genau das umschalten eines NANDs plus die umschaltzeit des eines NOT-Gatters vor dem /R-NAND, die Volle Zeit wird aber nur benötigt, sollte D zuvor 0 gewesen sein (somit das Not noch bis zu 8ns lang 1, und um den neuen Wert zu spikefrei zu schreiben, benötigt das NAND außerdem noch 41ns -> 49ns).

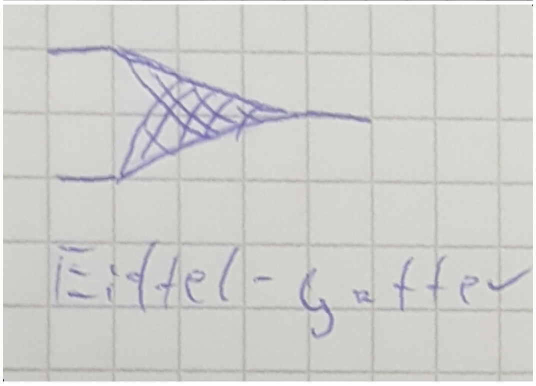
Die Zeit für y setzt sich ein wenig komplizierter zusammen. Und zwar wie folgt:
Wenn W 'kommt', dann braucht ein NAND zwischen [0.01:0.12]ns, um richtig zu schalten. Unser RS-FF benötigt eine Pulsweite von mindestens 0.68ns um das richtige abzuspeichern. Beim Umschalten danach benötigt das selbe NAND wieder mindestens 0.01 ns, weswegen unser FF nun insgesamt 0.69ns bekommen hat. Addieren wir also nur 0.67ns zu den maximalen 0.12 des NANDs, kommen wir auf 0.79ns, wobei das RS-FF immernoch 0.68ns Zeit hat -> $y = 0.79ns$.

Aufgabe 5









Baguatter: Man lege ein ungeröstes Baguett hinein und nach gewisser zeit ist es fertig :)
Weißwurst-Gatter: Mann nehme ein gutes Bayrisches Bier -> Mahlzeit.
Eiffel-Gatter: Führt zwei Teilschaltkreise auf Elegante Art und Weise zusammen.
NEiffel-Gatter: negiertes Eiffel-Gatter