

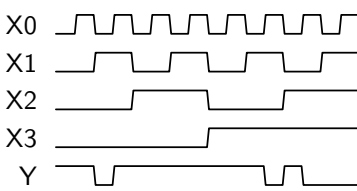
timing table from quine - dnf - 4 in var, galasm

david vajda

11. Juli 2026

1 heute beginnen mit

x_3	x_2	x_1	x_0	y_0
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0



2 gestern

```
;; (c) david vajda
;; wait loop - m8 - led out
;; 07/10/2026
```

```
.include "m8def.inc"
```

```
ldi r16, HIGH (RAMEND)
out SPH, r16
ldi r16, LOW (RAMEND)
out SPL, r16
ldi r16, 0xff
out DDRB, r16
```

```
ldi r16, 0xff
main001:
```

```

out PORTB, r16
rcall waitloop001
dec r16
rjmp main001

```

```

waitloop01:
push r16
push r17
push r18
ldi r16, 0x04
waitloop01l1:
ldi r17, 0xff
waitloop01l2:
ldi r18, 0xff
waitloop01l3:
dec r18
brne waitloop01l3
dec r17
brne waitloop01l2
dec r16
brne waitloop01l1
pop r18
pop r17
pop r16
ret

```

3 quine - dnf

(C) David Vajda
Sat Jul 11 07:48:40 2026
3 Network - TTL - Disjunktive Normalform

```

x2 x1 x0 y
0 0 0 0 0
1 0 0 1 1
2 0 1 0 1
3 0 1 1 1
4 1 0 0 0
5 1 0 1 0
6 1 1 0 1
7 1 1 1 1

```

oder wir probieren es heute mal mit vier eingangsvariablen, egal, ob es mit TTL Gattern einfach zu bauen ist, ist es auch!

interessant ist an der stelle uebrigens mal eine reale abschaetzung innerhalb des mathematischen!

die anzahl der gatter bei 4 variablen wird eindeutig verringert und zwar eher wie bei 3, auf das quine mc cluskey angewendet wurde.
so lautet meine einschaetzung nach augenmass! wie gross ist der unterschied wirklich?

einschaetzung nach augenmass: durch 4 variablen, werden die terme so stark miteinander verarbeitet, dass eben dadurch oft weniger bleibt, oder sie so wenige, mit 1, dass sie in sich geringer sind!

trotzdem dumme aufgabe!

liesse sich technisch informatisch machen, als einschaetzung! wie?

alles generieren und jeden fall durchgehen!...

gut: woran wir allerdings denken muessen, selbst, wenn die anzahl der gatter selbst stark reduziert sein muss, muessen wir daran denken!

ich habe ein elektrotechnik buch. Ein RAM Speicher,
... x-Decoder, y-Dekoder, so einfach ist es bei ROM Bausteinen.
eine RAM arbeitet mit einem: ...

es hat zum beispiel 512 x 16 x 8
dabei gehen vom x-Decoder direkt 512 leitungen.
daran denken:
vom y-Decoder: 16 gehen in ein spaltentor!

spaltentor filtert quasi zusaetzlich heraus???

, mit 8 Bit datenbreite.
es ist kein "symmetrisches" x und y dekoder!
daran denken: 512 leitungen, busse damit sind ziemlich aufwendig

merkt man bei ttl! bei 4 eingangsvariablen an einem 7404, inverter, durch die ganzen kabel daran wird die aufgabe ziemlich ...
anders beim gal! hier sehe ich das nicht, auch wenn aussen vier!

nur: beim gal drinnen? kann man so nicht sagen, aber siehe RAM!
RAM von innen.

ok, nun weiter:

(C) David Vajda
Sat Jul 11 07:59:55 2026
4 Network - TTL - Disjunktive Normalform

0 0 0 0 0 1

1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	0

0	0	0	0	0	1
1	0	0	0	1	1
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
12	1	1	0	0	1

Gruppe 0:

0	0	0	0	0	1
---	---	---	---	---	---

Gruppe 1:

1	0	0	0	1	1
4	0	1	0	0	1
8	1	0	0	0	1

Gruppe 2:

3	0	0	1	1	1
5	0	1	0	1	1
6	0	1	1	0	1
9	1	0	0	1	1
10	1	0	1	0	1
12	1	1	0	0	1

Gruppe 3:

7	0	1	1	1	1
---	---	---	---	---	---

0:1	0	0	0	-
0:4	0	-	0	0
0:8	-	0	0	0
1:3	0	0	-	1

ok, besser wir sortieren die sache anders.

Gruppe 0:

0 0 0 0 0 1

Gruppe 1:

1 0 0 0 1 1

4 0 1 0 0 1

8 1 0 0 0 1

Gruppe 2:

3 0 0 1 1 1

5 0 1 0 1 1

9 1 0 0 1 1

6 0 1 1 0 1

10 1 0 1 0 1

12 1 1 0 0 1

Gruppe 3:

7 0 1 1 1 1

0:1	0	0	0	-
0:4	0	-	0	0
0:8	-	0	0	0

1:3	0	0	-	1
1:5	0	-	0	1
1:9	-	0	0	1
4:5	0	1	0	-
4:6	0	1	-	0
4:12	-	1	0	0

8:9	1	0	0	-
8:10	1	0	-	0
8:12	1	-	0	0

3:7	0	-	1	1
5:7	0	1	-	1
6:7	0	1	1	-

sortieren:

0:1	0	0	0	-
4:5	0	1	0	-
8:9	1	0	0	-
6:7	0	1	1	-

0:4	0	-	0	0
1:5	0	-	0	1
8:12	1	-	0	0

3:7	0	-	1	1
1:3	0	0	-	1
4:6	0	1	-	0
8:10	1	0	-	0
5:7	0	1	-	1
0:8	-	0	0	0
1:9	-	0	0	1
4:12	-	1	0	0

ok, weiter verkuerzen:

Gruppe 0:				
0:1	0	0	0	-
Gruppe 1:				
4:5	0	1	0	-
8:9	1	0	0	-
Gruppe 2:				
6:7	0	1	1	-

Gruppe 0:				
0:4	0	-	0	0
Gruppe 1:				
1:5	0	-	0	1
8:12	1	-	0	0
Gruppe 2:				
3:7	0	-	1	1

Gruppe 1:				
1:3	0	0	-	1
4:6	0	1	-	0
8:10	1	0	-	0
Gruppe 2:				
5:7	0	1	-	1

Gruppe 0:				
0:8	-	0	0	0
Gruppe 1:				
1:9	-	0	0	1
4:12	-	1	0	0

Gruppe 0:				
0:1	0	0	0	-
Gruppe 1:				
4:5	0	1	0	-
8:9	1	0	0	-

Gruppe 2:
6:7 0 1 1 -

0:1:4:5 0 - 0 -
0:1:8:9 - 0 0 -
4:5:6:7 0 1 - -

Gruppe 0:
0:4 0 - 0 0

Gruppe 1:
1:5 0 - 0 1
8:12 1 - 0 0

Gruppe 2:
3:7 0 - 1 1

0:4:1:5 0 - 0 -
0:4:8:12 - - 0 0
1:5:3:7 0 - - 1

Gruppe 1:
1:3 0 0 - 1
4:6 0 1 - 0
8:10 1 0 - 0

Gruppe 2:
5:7 0 1 - 1

1:3:5:7 0 - - 1
4:6:5:7 0 1 - -
8:10 1 0 - 0

Gruppe 0:
0:8 - 0 0 0

Gruppe 1:
1:9 - 0 0 1
4:12 - 1 0 0

0:8:1:9 - 0 0 -
0:8:4:12 - - 0 0

mit dem 8:10 koennte man sich fragen???

allerdings gucken wir nach, ist es die 8?

nein, aber die 10, taucht tatsaechlich nur ein mal auf, also moeglich!

0:1:4:5 0 - 0 -
0:1:8:9 - 0 0 -
4:5:6:7 0 1 - -

0:4:1:5	0	-	0	-
0:4:8:12	-	-	0	0
1:5:3:7	0	-	-	1
1:3:5:7	0	-	-	1
4:6:5:7	0	1	-	-
8:10	1	0	-	0
0:8:1:9	-	0	0	-
0:8:4:12	-	-	0	0

0:1:4:5	0	-	0	-
0:4:1:5	0	-	0	-

0:1:8:9	-	0	0	-
0:8:1:9	-	0	0	-

4:5:6:7	0	1	-	-
4:6:5:7	0	1	-	-

0:4:8:12	-	-	0	0
0:8:4:12	-	-	0	0

1:5:3:7	0	-	-	1
1:3:5:7	0	-	-	1
8:10	1	0	-	0

0:1:4:5	0	-	0	-
0:1:8:9	-	0	0	-
4:5:6:7	0	1	-	-
0:4:8:12	-	-	0	0
1:5:3:7	0	-	-	1
8:10	1	0	-	0

0:1:4:5	0	-	0	-
0:1:8:9	-	0	0	-
4:5:6:7	0	1	-	-
0:4:8:12	-	-	0	0
1:5:3:7	0	-	-	1
8:10	1	0	-	0

ok, primimplikantentafel:

	0	1	3	4	5	6	7	8	9	10	12
--	---	---	---	---	---	---	---	---	---	----	----

0:1:4:5	H	H		H	H							
0:1:8:9	H	H						H	H			
4:5:6:7				H	H	H	H					
0:4:8:12	H			H				H			H	
1:5:3:7		H	H		H		H					
8:10								H		H		

	0	1	3	4	5	6	7	8	9	10	12	
0:1:4:5	H	H		H	H							
0:1:8:9	H	H						H	P			P
4:5:6:7				H	H	P	H					P
0:4:8:12	H			H				H			P	P
1:5:3:7		H	P		H		H					P
8:10								H		P		P

	0	1	3	4	5	6	7	8	9	10	12	
0:1:8:9	H	H						H	P			P
4:5:6:7				H	H	P	H					P
0:4:8:12	H			H				H			P	P
1:5:3:7		H	P		H		H					P
8:10								H		P		P

0:1:8:9	-	0	0	-
4:5:6:7	0	1	-	-
0:4:8:12	-	-	0	0
1:5:3:7	0	-	-	1
8:10	1	0	-	0

ok, damit haben wir eine gleichung, versuchen sie in abel aus zu druecken
und nicht nur das: wir machen unser eigenes ... diagramm

$$Y = \begin{matrix} /X2 \& /X1 \# \ /X3 \& X2 \# \ /X1 \& /X0 \# \ /X3 \& X0 \# \\ X3 \& /X2 \& /X0 \end{matrix}$$

$$Y = /X2 \& /X1 \# /X3 \& X2 \# /X1 \& /X0 \# /X3 \& X0 \# X3 \& /X2 \& /X0$$

bevor wir das ganze als GAL definieren, nehmen wir

- wir machen das Diagramm selber, per hand, und ueberpruefen es so
- wir lassen mit dem computer die formel noch ueberpruefen, indem wir sie nicht per hand hinschreiben, wir nehmen nicht excel, weil das alleine zu teuer ist, sondern nehmen sie in der bash, und erzeugen damit genau das diagramm

was auch in LaTeX verwendet wird.

```
#!/bin/bash
```

```
# david vajda  
# 07/11/2026  
# quine07112026dnf
```

```
## an der stelle muessen wir feststellen
```

ohne kommentar: bash skript laesst sich nicht feststellen. dass sich die Bash an der stelle nicht lohnt, denn wir brauchen eine tabelle. die sich einfacher fuellen laesst. das ist das problem. ob wir diese mit der bash erzeugen koennen oder nicht ist jetzt ein anderes thema, und wuerde zeit verschlingen, also an dieser stelle nicht angebracht, die diskussion...

ein c programm zur ueberpruefung moechte ich nicht verwenden!

;; dann verwende ich linux, weil hier laeuft excel, erst stelle ich die formel um...

nun, muessen wir, um die formel um zu stellen, sie in einzelteile zerlegen, excel stimmt w

ich speiche es dann als vcard, oder vcs, oder was auch immer, gibt tabellen her...

$$Y = /X2 \ \& \ /X1 \ \# \ /X3 \ \& \ X2 \ \# \ /X1 \ \& \ /X0 \ \# \ /X3 \ \& \ X0 \ \# \ X3 \ \& \ /X2 \ \& \ /X0$$

$$\begin{aligned} &/X2 \ \& \ /X1 \ \# \\ &/X3 \ \& \ X2 \ \# \\ &/X1 \ \& \ /X0 \ \# \\ &/X3 \ \& \ X0 \ \# \\ &X3 \ \& \ /X2 \ \& \ /X0 \end{aligned}$$

$$\begin{aligned} &=UND(NICHT(X2); NICHT(X1)) \\ &=UND(NICHT(X3); X2)) \\ &=UND(NICHT(X1); NICHT(X0)) \\ &=UND(NICHT(X3); X0) \\ &=UND(X3; NICHT(X2); NICHT(X0)) \end{aligned}$$

$$=ODER(UND(NICHT(X2); NICHT(X1)); UND(NICHT(X3); X2)); UND(NICHT(X1); NICHT(X0)); UND(NICHT(X3); X0); UND(X3; NICHT(X2); NICHT(X0))$$

vorher ein test, X2 durch Bn ersetzen, X3 durch An usw.

$$=ODER(UND(NICHT(B3); NICHT(C3)); UND(NICHT(A3); B3)); UND(NICHT(C3); NICHT(D3)); UND(NICHT(X3); X0); UND(X3; NICHT(X2); NICHT(X0))$$

ok: fehlerhaft waren die deutschen namen, es muss ins englische uebersetzt werden.

```
=OR(
    AND(NOT(B3);NOT(C3));
    AND(NOT(A3);B3);
    AND(NOT(C3);NOT(D3));
    AND(NOT(A3);D3);
    AND(A3;NOT(B3));
    NOT(D3))
```

```
=OR(AND(NOT(B3);NOT(C3));AND(NOT(A3);B3);AND(NOT(C3);NOT(D3));AND(NOT(A3);D3);AND(A3;NOT(B3);NOT(D3)))
```

```
=OR(AND(NOT(B);NOT(C));AND(NOT(A);B);AND(NOT(C);NOT(D));AND(NOT(A);D);AND(A;NOT(B));NOT(D)))
```

mit lambda ausdruecken und tabelle funktionierte nicht...

begrueundung, libre office und open office calc und all das sind so oder so keine sichere a
 schon auf grund der notwendigkeit ihrer GUI
 einerseits. sicherlich lassen sich daten als CSV sichern.
 trotzdem: Macros und Co Kg werden in Basic Scripts untergebracht. die sache macht keinen s

ich habe wiederum ueber eine alternative nachgedacht.
 warum?

aus einem einzigen grund:

wenn ich ein Bash Skript nehme, dann muss ich die variablen fuellen.
 ich habe keine tabelle vorliegen, das ist dringend notwendig.

eine alternative schafft python3

python3 ist eine sprache, die jetzt alles erlaubt

- a) schon bei der Graphentheorie sieht man - die graphen lassen sich bestens bearbeiten, eu
- b) alles laesst sich graphisch veranschaulichen
 - ... was oft nicht der fall sein sollte, GUI, jedenfalls nicht malerisch, oder auch, aber a
 - ... aber es laesst sich anders veranschaulichen
- c) eine CSV liegt vor?
 - ... in python 3 lassen sich mit der formel diagramme erzeugen
 - ... es laesst sich gleich das erzeugen, was wir ja in LaTeX haben wollten
 - ... damit beisst sich die Katze in den schwanz, warum?
 - ... darum:
 - ... gerade ohne gtkwave - GUI - muss ich python3 benutzen, fuer TikZ ... und das d
 - ... aber ich brauche so oder so python3, tabelle in CSV
 - ... aber: am ende, habe ich die GAL Formel fuer galasm
 - ... GAL Formel fuer galasm???
 - ... richtig: nur, diese sieht ja anders aus

```

... grund:
    ich uebe ja kein VHDL
    aus guten grund: ich glaube in FPGA ist eigentlich ein GAL in grosser mass
    kann ich mir vorstellen:
        entity: sind in unserem GAL pins
        architecture: formel ...
        ok: jeder entity, architecture in einem GAL auf FPGA?
            ... selbst, wenn moegliches modell/denkweise und:

        aber!!! ich uebe keine VHDL
        sondern ABEL, galasm... Lattice gal16v8...

... eben aus dem grund: VHDL war da, weil anderes nicht...
    praktisch wg. grossen prozessor, MIPS32, auf FPGA, blos

... moeglich kleiner prozessor.
... eigentlich besser.
... und: ich uebe ABEL/HDL/... galasm

ok: und die Formel:
    - liegt in galasm vor
    - und python3?
        eben python3, selbst, wandelt diese formel sehr schnell um in etwas,
        was es dann benutzen kann.

# Y = eval(python3formula,{"X3": 0,"X2": 0,"X1": 0,"X0": 0})

import csv
with open('bool4tab.csv', newline='\n') as csvfile:
    bool4tab = csv.reader (csvfile, delimiter=',')
    for row in bool4tab:
        print ('', '.join(row))
galasmformula = "Y = /X2 & /X1 \# /X3 & X2 \# /X1 & /X0 \# /X3 & X0 \# X3 & /X2 & /X0"
python3formula = galasmformula.replace ("/", " not ").replace ("&", " and ").replace ("\#"
print (python3formula)
with open('bool4tab.csv', newline='\n') as csvfile:
    bool4tab = csv.reader (csvfile, delimiter=',')
    i = 0
    for row in bool4tab:
        if i mod 4 == 3:
            x0 = row
        if i mod 4 == 2:
            x1 = row
        if i mod 4 == 1:
            x2 = row
        if i mod 4 == 0:
            x3 = row
            Y = eval(python3formula,{"X3": x3,"X2": x2,"X1": x1,"X0": x0})

```



```

0;0;0;0
0;0;0;1
0;0;1;0
0;0;1;1
0;1;0;0
0;1;0;1
0;1;1;0
0;1;1;1
1;0;0;0
1;0;0;1
1;0;1;0
1;0;1;1
1;1;0;0
1;1;0;1
1;1;1;0
1;1;1;1

```

und das CSV File muss so aussehen:

und noch x3, x2, x1, x0 rein tun, dazu noch:

- 1.) vergleichen mit erster tabelle
- 2.) erzeugen in LaTeX geht nun ohne probleme
- 3.) vielleicht das python programm so machen, dass formel uebergeben wird...

```

not X2 and not X1 or not X3 and X2 or not X1 and not X0 or not X3 and X0 or
0 0 0 0 1
0 0 0 1 1
0 0 1 0 0
0 0 1 1 1
0 1 0 0 1
0 1 0 1 1
0 1 1 0 1
0 1 1 1 1
1 0 0 0 1
1 0 0 1 1
1 0 1 0 1
1 0 1 1 0
1 1 0 0 1
1 1 0 1 0
1 1 1 0 0
1 1 1 1 0

```

3.1 python3 code

```

import csv
import sys

```

```

import sympy as sp
from sympy.logic import boolalg

# (c) david vajda
# 07/11/2026
# quine .. 4 input var .. dnf ..

# with open('bool4tab.csv', newline='\n') as csvfile:
#     bool4tab = csv.reader (csvfile, delimiter=',')
#     for row in bool4tab:
#         print ('', '.join(row))
# galasmformula = "Y = /X2 & /X1 \# /X3 & X2 \# /X1 & /X0 \# /X3 & X0 \# X3 & /X2 & /X0"
galasmformula = sys.argv [1]
print (galasmformula)

python3formula = galasmformula.replace ("/", "not ").replace ("&", " and ").replace ("\#", " or ")
print (python3formula)
print("\begin{tabular}{|c|c|c|c|c|}")
print("\hline")
print (" $x_3$ & $x_2$ & $x_1$ & $x_0$ & $y_0$\\")
print("\hline")
with open('bool4tab.csv', newline='') as csvfile:
    bool4tab = csv.reader (csvfile,delimiter=';')
    for row in bool4tab:
        X0 = bool(int(row [3]))
        X1 = bool(int(row [2]))
        X2 = bool(int(row [1]))
        X3 = bool(int(row [0]))
        Y = eval(python3formula,{"X3": X3,"X2": X2,"X1": X1,"X0": X0})
        # print (not X2 and not X1 or not X3 and X2 or not X1 and not X0 or not Y)
        # print (Y)
        print ("$", int(X3), "$ & $", int(X2), "$ & $", int(X1), "$ & $", int(X0), "$ & $")
        print ("\hline")
print ("\end{tabular}")

print("\begin{tikztimingtable}")
X0TikZTimingStr = "X0 & "
X1TikZTimingStr = "X1 & "
X2TikZTimingStr = "X2 & "
X3TikZTimingStr = "X3 & "
YTikZTimingStr = "Y & "
HighLowPegelStr = ["L", "H"]
with open('bool4tab.csv', newline='') as csvfile:
    bool4tab = csv.reader (csvfile,delimiter=';')
    for row in bool4tab:
        X0 = bool(int(row [3]))
        X1 = bool(int(row [2]))
        X2 = bool(int(row [1]))
        X3 = bool(int(row [0]))

```

```

Y = eval(python3formula,{"X3": X3,"X2": X2,"X1": X1,"X0": X0})
X0TikZTimingStr = X0TikZTimingStr + "1" + str(HighLowPegelStr[X0]) + " "
X1TikZTimingStr = X1TikZTimingStr + "1" + str(HighLowPegelStr[X1]) + " "
X2TikZTimingStr = X2TikZTimingStr + "1" + str(HighLowPegelStr[X2]) + " "
X3TikZTimingStr = X3TikZTimingStr + "1" + str(HighLowPegelStr[X3]) + " "
YTikZTimingStr = YTikZTimingStr + "1" + str(HighLowPegelStr[Y]) + " "
X0TikZTimingStr = X0TikZTimingStr + "\\\\"
X1TikZTimingStr = X1TikZTimingStr + "\\\\"
X2TikZTimingStr = X2TikZTimingStr + "\\\\"
X3TikZTimingStr = X3TikZTimingStr + "\\\\"
YTikZTimingStr = YTikZTimingStr + "\\\\"
print(X0TikZTimingStr)
print(X1TikZTimingStr)
print(X2TikZTimingStr)
print(X3TikZTimingStr)
print(YTikZTimingStr)
print("\\end{tikztimingtable}")

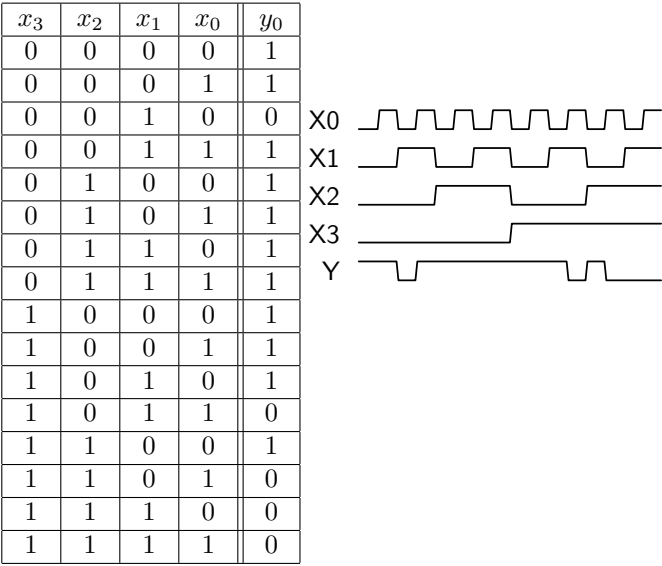
```

3.2 CSV-file

```

0;0;0;0
0;0;0;1
0;0;1;0
0;0;1;1
0;1;0;0
0;1;0;1
0;1;1;0
0;1;1;1
1;0;0;0
1;0;0;1
1;0;1;0
1;0;1;1
1;1;0;0
1;1;0;1
1;1;1;0
1;1;1;1

```



4 Bericht + Internet

17:17



python3 galasm formular

 +1

If you are looking to parse or manipulate **Boolean algebra formulas** for programmable logic (like GALasm files) using Python 3, the standard toolset is `sympy`. You can use its `to_dnf` (Disjunctive Normal Form) function to simplify formulas into standard sums of products, which map perfectly to GAL equations.

justanotherelectron... +1

	Law of Addition (OR)	Law of Multiplication
Identity	$A + 0 = A$	$A \cdot 1 = A$
Null element	$A + 1 = 1$	$A \cdot 0 = 0$
Idempotent	$A + A = A$	$A \cdot A = A$
Complement	$A + \bar{A} = 1$	$A \cdot \bar{A} = 0$
Commutative	$A + B = B + A$	$A \cdot B = B \cdot A$
Associative	$A + (B + C) = (A + B) + C$	$A(B \cdot C) = (A \cdot B) \cdot C$
Absorption	$A + (A \cdot B) = A$	$A(A + B) = A$
Distributive	$A + (B \cdot C) = (A + B)(A + C)$	$A(B + C) = (A \cdot B) + (A \cdot C)$
De Morgan's	$\overline{(A + B)} = \bar{A} \cdot \bar{B}$	$\overline{(A \cdot B)} = \bar{A} + \bar{B}$
Negative	$\bar{\bar{A}} = A$	

18

Frage stellen



python3 galasm

1. Requirements

Ensure you have SymPy installed:

```
bash
```

```
pip install sympy
```

17:17



```
pip install sympy
```

Verwende Code mit Vorsicht.



2. Python 3: Formula Manipulation Script

This script takes a raw Boolean formula and automatically converts it into a `GALasm`-friendly string. GitHub +1

python

```
import sympy as sp
from sympy.logic import boolalg

# Define your logic variables (inputs)
A, B, C = sp.symbols('A B C')

# Example formula (e.g., Output = A AND
formula = A & (B | ~C)

# 1. Simplify the expression (Boolean m
simplified_expr = sp.simplify_logic(for
```

19

Frage stellen



17:17



python

```
import sympy as sp
from sympy.logic import boolalg

# Define your logic variables (inputs)
A, B, C = sp.symbols('A B C')

# Example formula (e.g., Output = A AND (B OR NOT C))
formula = A & (B | ~C)

# 1. Simplify the expression (Boolean simplification)
simplified_expr = sp.simplify_logic(formula)

# 2. Convert to Disjunctive Normal Form (DNF)
# GALasm only supports combinations of AND and OR
dnf_expr = boolalg.to_dnf(simplified_expr)

# 3. Convert SymPy's syntax to GALasm syntax
# (Python uses '&' for AND and '|' for OR)
galasm_str = str(dnf_expr).replace('&', 'AND').replace('|', 'OR')

print("Original Formula:", formula)
print("GALasm Equation:", f"OUTPUT = {galasm_str}")
```

Verwende Code mit Vorsicht.



20

Frage stellen



17:47



nen zum
Datenschutz



an ein Python-Skript in der
ergeben, verwenden Sie das Modul
`sys.argv` (für einfache Argumente) oder das
Modul `argparse` (für komplexe Parameter mit
Optionen/Hilfetexten). [Rheinwerk Verlag +2](#)

Für ein schnelles Skript, das zwei Zahlen addiert:

python

```
import sys

if len(sys.argv) == 3:
    zahl1 = float(sys.argv[1])
    zahl2 = float(sys.argv[2])
    print(f"Summe: {zahl1 + zahl2}")
else:
    print("Bitte genau zwei Zahlen als
```

Verwende Code mit Vorsicht.



Der Aufruf in Ihrem Terminal erfolgt dann so:

```
python3 skript.py 5 7
```

[DataBraineo +1](#)

Für professionellere Programme, die Hilfetexte.

²¹

Frage stellen

17:47



parameter python3 programm console



nen zum
Datenschutz

an ein Python-Skript in der
ergeben, verwenden Sie das Modul
`sys.argv` (für einfache Argumente) oder das
Modul `argparse` (für komplexe Parameter mit
Optionen/Hilfetexten). [Rheinwerk Verlag +2](#)

Für ein schnelles Skript, das zwei Zahlen addiert:

python

```
import sys
```

Frage stellen

22



parameter pytho



David Vajda (c)

07/11/2026

Seite 2, Beschreibung

und zum Erstellen des LaTeX*-Tex
files:
python3

Die Programmiersprache python3
eignet sich zur Manipulation und
Erstellung von Inhalten solcher
Art, wie Tabellen:

usw. Die mathematischen
sind

Latex und python werden im
Debian Betriebssystem (Distribution)
installiert mit

```
apt-get install latex-full  
apt-get install python3
```

Sie wiederum enthalten (p.) Bibliotheken
könnte man in der Programmiersprache
sagen sind

includes, import, packages

David Vajda (c)

07/11/2026.

Seite 4, Beschreibung.

$y = 1x0 \& x1 \# 1x0 \& x2$

www.

Zufälligerweise hat python3
noch ein Paket.

glacon ist eine Software
die Hardwarebeschreibungssprachen auf

programmiert.

pal: programmierbar

gal: reprogrammierbar!

Sie erstellt eine "Map" von fuses-
kurz Binärcode wie auch immer.

Diese kommt auf den
gal per programmiergerät.

David Vajda (c)

07/11/2026

Seite 6, Beschreibung

vor allem Aibz

Aibz

merken als:

Aib wie Aick bei
uhr

und z wie zila
oder Zigarette ... ok

es gibt dieses package mit
abwandlung:

Aibz-timing ..

wiederum ist pythou 3 sehr
mächtig

+++ Latex kann:

Hieroglyphen

noten, Kurrentschrift ..

alles ..

Jann Vajda (c)

07/11/2026

Seite 8, Bedeutung...

Sann ab programm ausführen aber
mit python 3

python3 file.py --parameter
man kann auch hier parameter
argumente übergeben:

argv

argv[0] ist üblicherweise
immer alle prog
sprache!

programmname: ...

oh... und:

ich denke es ist besser ganzen
ausdruck übergeben...

2.) csv

David Vajda (c)

07/11/2026

Seite 10, Beschreibung

✓ Das halbsendigramm ist besser
✓ konnte nicht so hartwerc (kurz)
weit gekommen

dafür pg 3

✓ Dann auf gel 16+8

jetzt oscilloskop:

Das timing diagram ist
gut

ich habe ja timer intervall
atmega8:

und oscilloskop kann
man so:

Das bild passt
aber 10 MHz

und timer:

Dann machen wir test

K, Z

genau 50:50

David Vajda (c)

07/11/2026

Beschreibung, Seite 12

Strecke aus +

Bild Widerstand,

Kondensator

Kondensator z. B. Widerstand

Wirkwiderstand

Blindwiderstand

Scheinwiderstand

$$X = \frac{V}{i} \quad \text{immer noch Ohm.}$$

X aber Blind bei C und L

$$\text{und } X = \frac{1}{2 \cdot \pi \cdot f \cdot C}$$

und: $X = \frac{1}{2 \cdot \pi \cdot f \cdot C}$

Das rein und wo kaufen, wie
fein: Widerstand:

30 Cent und
programmiergerät:
abmega 8 günstig
30€ auch und himmel.

07/11/2026

Land Vayda

Seite 13, Beschreibung
Papier linkeffen

nicht alles durch einander
muss geplant werden

1.) mit jeder Tag
also dann nur

2.) was muss sein

3.) wann wieder

Das geht nicht überlegen

das reicht nicht

Kopf nicht zerbrechen:

ein extra Tag. -

und Auerkennung (e)

Ozillatshop:

Schaubild soll sein passen wie
im pdf durch Inter..

David Vayda (c)

27/11/2026

Seite 9, Beschreibung

bin sider:

Sie kennen

- Smartphone
- Adressbuch -
- Kalender
- Excel

in csv wird Tabelle gespeichert

Trenner : spalte : ";"

und zeile : "\n"

so python 3 kann das:

und: Tabelle boolean
nein

Dann kann es auch
glaube heute mehr
gemacht

Grund Nagda (c)

07.11.2026

Beschreibung, Seite 11

Ziel für Zukunft

Die Leute sind alleine gelassen
spenden lassen

ich habe ein wenig
hingewiesen auf
Situation

Bei Bewe / Kaufland pinwand.
ich weiß problem:

$$P = V \cdot z$$

$$R = \frac{V}{z}$$

U immer oben z U R

$$z = \frac{U}{R} \quad U = R \cdot z \quad R = \frac{U}{z}$$

und U in der Mitte

ich habe Trecker. In wenigen
Tagen das Wichtigste

David Vajda(c)

07/11/2026

Seite 7, Besondere...

python 3:

wichtig bei programmieren in der
bash (konsole)

immer parameter:

mv bewegt

mv file1 file2

file1 nach file2

Diese parameter sind konkrete "Daten"
andere sind Steuerbefehle

echo -n "Hallo"

keine new line...

aber python3 wird interpretiert, entweder
im interpreter oder als text datei

*.py speichern.

David Vajda (c)

07/11/2026

Seite 5, Beschreibung...

Damit:

gabcom benutzt die ausdrücke
die relativ lose liegen...

wg. kleiner baustein

spg groß: spg. mehrere
bausteine auf baustein!

Sprache: nicht verilog...

ob: nun:

pythran3 hat wie

Latex viele "adapter"

Latex benutzt besonders bei

✓ Schaltplänen IEEE 91-1984

(Schaltpläne usw)

✓ Atom Diagramme...

✓ Timing Tables (Diagramme)

also: Zustellungen Elektro +

Info...

David Vajda (c)

07/11/2026

Seite 3, Beendigung...

in Latex: `\usepackage{...}`

in python3:
`import ...`

Die verwendete Software:

Latex (pdf~~late~~x) texlive-full

python3

packages:

latex: tikz-timing

python3-

csv

sys

sympy

in Wirklichkeit ist ja noch
ganzes beteiligt!

Denn die booleschen
Ausdrücke gehen wir ein ab:

David Vajda (v)

07/11/2026

Beschreibung des heutigen

Seite 1

time overloaded/overflow/exceeded

1.) Das erste was ich heute gemacht habe:

✓ ich benutze kein vhdl
kein ... ghdl - kein
gtkwave

✓ vhdl + fpga als alternative
zum:

abel/galsim...

desing. Diagramm neu erstellen

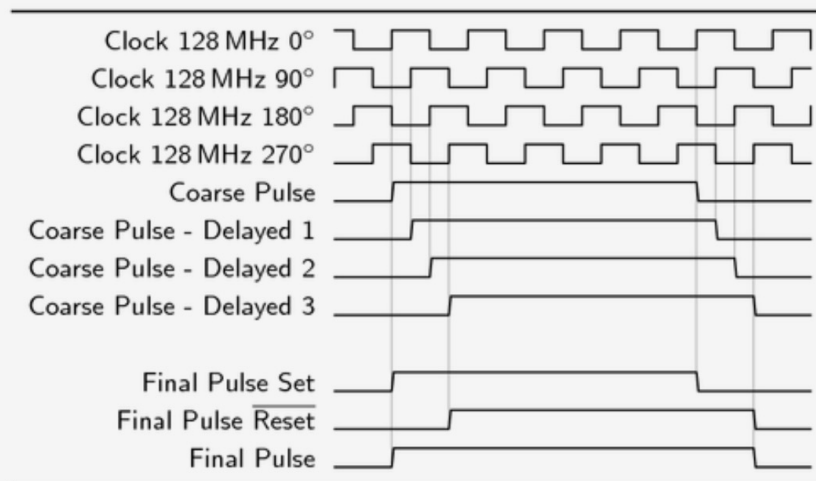
Timing table

Dazu benutze ich LaTeX ...

und das LaTeX paket
von tikz : tikz-timing

information about them.

19:16



Edit and compile if you like:

```
1 % Timing Diagram using the 'tikz-timing' package
2 %
3 % (c) Copyright 2009 - Martin Scharrer
4 %
5 % This code is open source under the GPLv3
6 % Please feel free to use, change and copy
7 %
8
9 \documentclass{article}
10 \usepackage{tikz-timing}[2009/05/15]
11
12 \usepackage[active,tightpage]{preview}
13 \PreviewEnvironment{tikzpicture}
14 \setlength\PreviewBorder{5pt}%
15
16
17 \pagestyle{empty}
18 \def\degr{{}^{\circ}}
19
20 \newif\ifcomment
21
22 % Activate the following line to compile document
23 %\commenttrue
24
25 \begin{document}
26
27 \ifcomment
28 \newcommand{\mycommand}[1]{\textit{#1}}
```



texample.net



19:16



TeXample.net

LaTeX and TikZ examples



Timing diagram with the tikz- timing package

by Martin Scharrer August 3, 2009

This example shows how to make a timing diagram with the [tikz-timing package](#). This timing diagram was used by the package author in a recent work and shows several clock and pulse signals. The relationship between the clock and signal edges is shown using horizontal lines.

Timing diagrams like this can be done using the `tikztimingtable` environment which has the same syntax as a `tabular` environment with two

columns. The first column contains the names of the signals and the second column contains the timing information.



texample.net



19:16



Google

Anmelden

Q python3 galasm formular

KI-M

Suc



DE

Anmelden

Bevor Sie zu Google weitergehen

Wir verwenden [Cookies](#) und Daten, um



Google-Dienste zu erbringen und zu betreiben



Ausfälle zu prüfen und Maßnahmen gegen Spam, Betrug und Missbrauch zu ergreifen



Daten zu Zielgruppeninteraktionen und Webstatistiken zu erheben. Mit den gewonnenen Informationen möchten wir verstehen, wie unsere Dienste verwendet werden, und die Qualität dieser Dienste verbessern.

Wenn Sie „Alle akzeptieren“ auswählen.

Weitere Informationen

High
GLMs with glum 3

07.06.2024 — The new glum release introduces a formula interface that simplifies preprocessing and offers various enhancements for categorical and mixed ...



Q python3 galasm



19:16



python3 galasm formular



To write a logic formula using **GALasm**, you must create a `.pld` (programmable logic device) source file using Boolean equations. The language uses specific logical operators: **AND** (`*`), **OR** (`+`), and **NOT** (`/`).

[www.sh-elektronik....](#) +2

A standard format and a basic example are detailed below.

The Source File Structure

Every `.pld` file for **GALasm** follows this exact sequence:

1. **Header:** Chip name, GAL type (e.g., `GAL16V8`), and an 8-character signature.
2. **Pin Declarations:** Define inputs first, then output pin names.

Frage stellen

39



python3 galasm



19:16



AND (AND_OUT) and an OR (OR_OUT) output:

Break Into Program +2

text

```
CHIP MyChip GAL16V8  
SIGNATURE "TEST0001"
```

```
A B NC NC NC NC NC NC NC GND  
VCC NC NC AND_OUT OR_OUT NC NC NC NC NC
```

EQUATIONS

```
AND_OUT = A * B  
OR_OUT = A + B
```

DESCRIPTION

This chip implements a basic AND and OR

Verwende Code mit Vorsicht.



Supported Syntax

- **NOT / Invert:** Use `/` at the beginning of a pin name. Example: `/A0`

Frage stellen

19:16



Supported Syntax

- **NOT / Invert:** Use `/` at the beginning of a pin name. Example: `/A`
- **AND operator:** Use `*`. Example: `A * B`
- **OR operator:** Use `+`. Example: `A + B`
- **Registered Output:** If you are using a D Flip-Flop in a GAL, use the `.R` suffix. (e.g., `Q0.R = D0 * INC`).
- **Tri-State / Output Enable:** Use the `.E` suffix. (e.g., `Q0.E = 0E`).  Break Into Program +2

How to Compile

`GALasm` is a command-line utility. You can compile the `.pld` file into a `JEDEC` (`.jed`) file, which is the file format required by your physical hardware programmer (such as a TL866II Plus).

 Hackaday.io +1

bash

```
galasm YourFile.pld
```

41

Frage stellen

19:17



3.14.6



Quick search

Go

argparse — Parser for command-line options, arguments and subcommands

Added in version 3.2.

Source code: [Lib/argparse.py](#)

Note: While `argparse` is the default recommended standard library module for implementing basic command line applications, authors with more exacting requirements for exactly how their command line applications behave may find it doesn't provide the necessary level of control. Refer to [Choosing an argument parsing library](#) for alternatives to consider when `argparse` doesn't support behaviors that the application requires (such as entirely disabling support for interspersed options and positional arguments, or accepting option parameter values that start with `-` even when they correspond to another defined option).

The `argparse` module makes it easy to write user-friendly command-line interfaces. The program defines what arguments it requires, and `argparse` will

Tutorial

This page contains the API reference information. For a more