

quine mc cluskey

David Vajda

26. Januar 2026

(C) David Vajda Sat Jan 10 11:51:31 2026 4 Network - TTL - Disjunktive Normalform

nr.	x_3	x_2	x_1	x_0	y
0	f	f	f	f	f
1	f	f	f	w	w
2	f	f	w	f	f
3	f	f	w	w	w
4	f	w	f	f	w
5	f	w	f	w	w
6	f	w	w	f	f
7	f	w	w	w	f
8	w	f	f	f	f
9	w	f	f	w	f
10	w	f	w	f	f
11	w	f	w	w	w
12	w	w	f	f	w
13	w	w	f	w	f
14	w	w	w	f	w
15	w	w	w	w	f

1 Ein wenig von dem was uns zu erwarten hätte

...

ich musste den text auf grund seiner laenge streichen... keine sorge.... vergessen sie nicht - darstellung ist wichtig... ein paar hinweise

jetzt gleich 1.) hauptsache: lange mathematischer text zum auswendig lernen + physik am ende teilweise nicht richtig aber doch nicht unwahr, am anfang ganz fuehrt zum richtigen =, als pdf 2.) dazu extrem viele uebungen - computersysteme 1 u. 2 schaltung.

- uebung - uebung - uebung - etc. wdh.

2.) will nicht vorenthalten: keine homepage, keine gueter, sondern: pdf book without quatschen ... und: icloud link + pdf book vielleicht wenn finden = (german) icloud link

3.) dazu chemie 3.1.) laenge: punkt, linie, finger = zoll = inch, handbreit, fuss, ... 3.2.) cent, gran, karat, skrupel... 3.3.) periodensystem lernen: H, He, Li, Be, B, C, N, O, F, Ne, Na, Mg, Al, Si, P, S, Cl, ... Au und Ag!!! Cu und Fe... fuehrt zu: ingenieure tabelle lernen, please - trotz statistischer auswertung!!!

3.4.) Mol: 6.022×10^{23} oder 22 teilchen molare masse: gramm pro mol
 molares volumen: liter pro mol stoffmengenkonzentration: mol/liter
 why: easy: merke: guenstig/moeglich guenstig pro moeglich
 moeglichst viel, guenstigst klein
 klein oben, gross unten g/m ok und gramm/mol
 perfekt:
 4.) benzin und wasser auf die wage 5.) weitere versuche:
 1.) die schwerheit in der naturwissenschaft: cyankali
 deutsch: blaukraut bleibt blaukraut und brautkleid bleibt brautkleid...
 why? wg. beim kochen, in blaukraut, nikotin, aber bittermandel, riecht suess,
 ist blausaeure
 cyan - fuehrt zu: gelbes, rotes, weisses, ... kreuz: moderne
 what is cyankali
 kali bergwerk: leute arbeiten, die zu viel energie haben... ohne boese, aber
 spaeter. und ist blausaeure weiss durchsichtig: sind kalium pulver darin binde-
 mittel??
 die seife
 reaktion im alltag anders: einfach butter oder oel zu aetznatron
 vorsichtig: sehr gefaehrlich: ducken im versuch am anfang normal!!! sehr sehr
 sehr gefaehrlich
 trotzdem: einfach zu sammen tun: james alder reaktion? what ever: zusam-
 men tun: cyankali wahrscheinlich aehnlich
 im netz... eine beschreibung, mit destille, aber nicht mit wasser sicher mach-
 bar?
 easy? denkste, lsa und diethylamid sicher auch oder nicht, fuer chemiker oder
 fuer nicht chemiker, oder fuer gar nicht? keine ahnung
 was hat das mit wissenschaft zu tun? zu viel geredet, in der soc waschkueche?
 viele schaften es vorher nicht
 waere wunder...
 wasser destillieren: nietzsche wasser tut es auch? bei was? was ist die stapel-
 einheit 1024? am ende darueber?
 gut, richtig verstanden oder nicht: Au Gold und Ag Silber dazu Cupfer und
 so weiter..
 das waere auch eine schoene uebung oekonomisch nutzbar
 alles in pdf, wollte ich sagen, nachher
 nachher, icloud
 versuch in pdf reader unter zu bringen
 ueben ueben ueben... jetzt uebung skript, weil 5V adapter weg.. nachher
 schaltung...

2 quine mc cluskey

nr.	x_3	x_2	x_1	x_0	y
0	f	f	f	f	f
1	f	f	f	w	w
2	f	f	w	f	f
3	f	f	w	w	w
4	f	w	f	f	w
5	f	w	f	w	w
6	f	w	w	f	f
7	f	w	w	w	f
8	w	f	f	f	f
9	w	f	f	w	f
10	w	f	w	f	f
11	w	f	w	w	w
12	w	w	f	f	w
13	w	w	f	w	f
14	w	w	w	f	w
15	w	w	w	w	f

```

y    <=  (x3 and x2 and not x0) or
          (not x3 and not x1 and x0) or
          (x2 and not x1 and not x0) or
          (not x2 and x1 and x0);

```

$$y <= (x_3 \text{ wedge } x_2 \text{ wedge } \neg x_0) \vee (\neg x_3 \text{ wedge } \neg x_1 \text{ wedge } x_0) \vee (x_2 \text{ wedge } \neg x_1 \text{ wedge } \neg x_0) \vee (\neg x_2 \text{ wedge } x_1 \text{ wedge } x_0);$$

Daraus wird: ... gerade ungeduld

3 wie oft wiederholt sich die uebung...

3 Bit, Eingangsvariablen: $2^3 = 8$. ok, also 8 bit ausgang. wie viele moeglichkeiten gibt es? $8! = 8 \cdot 7 \cdot \dots \cdot 2 \cdot 1$???, nein, da wir nicht haben, permutation! $Y = \{8, 7, 6, \dots, 2, 1\}$, also $\{7, 8, \dots\}, \dots$, sondern $Y = (0, 0, 0, \dots, 0) \dots (1, 1, 1, \dots, 1)$ ist das ein byte. bei 2^{16} naehmlich eingangsvektor: 4 bit haben wir: $64k$ moeglichkeiten, also $64k + 256$ und jetzt frage: wahrscheinlichkeitsrechnung, wie oft kommt die selbe uebung?

also: $64k + 256$ sind durch 365 zu teilen, bei einer uebung am tag. wenn man aber nur alle 2 tage eine neue uebung, sind das: $\frac{64k + 256}{365} \cdot 2$ sind: 360,504109589

also, damit sich die uebungen wiederholen, waeren, 360 jahre notwendig. werden nur 3 Bit verwendet, weniger als ein jahr, etwa 0.7 jahre. ok, doch wie gross ist die wahrscheinlichkeit??? erwartungswert nutzt hier nichts, ich kann den durchschnittswert ausrechnen. wert von 0..255 und jedes wahrscheinlichkeit

von $\frac{1}{256}$

$$\sum_{i=0}^{255} \frac{1}{256} \cdot i$$

4 Wahrheitstabeln und Gatter

4.1 AND $\wedge$

4.1.1 Schaltalgebra

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

4.1.2 Boolesche Algebra

x_1	x_0	$y = x_1 \wedge x_0$
f	f	f
f	w	f
w	f	f
w	f	w

4.2 OR $\vee$

4.2.1 Schaltalgebra

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

4.2.2 Boolesche Algebra

x_1	x_0	$y = x_1 \vee x_0$
f	f	f
f	w	w
w	f	w
w	f	w

4.3 NOT $\neg$

4.3.1 Schaltalgebra

A	B
0	1
1	0

4.3.2 Boolesche Algebra

x_0	$y = \neg x_0$
f	w
w	f

01/23/2026
 (c) David Vajda
 Kleine Übung für zwischen
 drin Seite 1

Dim 44300 ~ Schaltnetz, Schaltfunktion, ...
 Dim 40700 ~ Gatter... : Grundverknüpfungen
 Dim 40900

Dim 44300/83: Schaltnetz
 " " /89: Schaltfunktion
 " " /181: Kodierglied...

iec60617-12? : Die modernen
 Schaltzeichen (Grundver-
 knüpfungen)

aus aus i 90-1984 : US-American

HE-Gatter:

mand, nor, and, or, xor, negation,
 arithmetic, demux, decode, mux,
 Dec-(32) - 70-7-seg...
 fl, parallel reg, seriell, ram, mono-
 flop

04,05,14: not, 32: or 08,09,11,21:

83,85,181,182,183: arith, 7447: Dec to-
 and 7-seg...

nam: 16x4: 4 bit wortleitung..

and:  or:  xor:  not: 

a	b	c
0	0	0
0	1	1
1	1	1
1	0	1

or

a	b	c
0	0	0
0	1	0
1	0	0
1	1	1

and

a	b
0	0
1	1

insert

01/24 22626

(c) David Vajda

Seite 7, Homepage...

6.) HE... Inhaltsplan:

baue

7.) Oszilloskop

8.) atmega8

9.) quelltest

10.) gal

dazu lernen

zusatz:

Linux übungprogramme

David Vajda (c)
01/26/2026
Seite 2, gal...

ändern:

1.) pal

2.) gal

eines ist und programmierbar,
das andere oder oder von
prinzip beides...

...

oder und programmierbar: gucken...

aber: aber:

das hängt damit nicht
zusammen...

damit kann ich den meinsten
mi mo (Oemco), m₁ (olmc₁),
noch mal negieren.

Sonst: Das hat damit nicht
zu tun:

eingang: negativ o. nicht...

Samuel Byde (c) Kunde
Seite 6, gal
01/26/2026
Kombinatorisch,

also: Bildung
ohne direkter
kombinatorisch.

$o_{yn} = 1$
 $a_{CO} = 1$
 $a_{CT}(u) = 1$

also:

+ registered output $\hat{=}$ mit ff
o. feedback...

row (output) pol...

also:

Wahrheitstabelle

01/24/2006 + 2110
 Kuchner - 1000 (5)
 Seite 2 (c) beweis
 Day 20
 factor:
 $a, 0, x^2, [a \cdot x^2]$
 $5, 4$
 Kern:
 $ax^2, bx^3, 4a, 4ax$
 expression:
 $ax^2 = bx$
 $ax^3 - bx = 0$
 expression
 min + max bei:
 1) Produktieren
 das ist das richtige!!!
 gel
 2.) gel

01/24/2026
(c) David V. Ka
Seite 4, Homepage
40 +

Idee:

man kann überlegen:

1.) was passiert in der

Stufe: Zucker

... nachher: Zucker

Restigungsprozess

Reinigung / Zyklus

1.) Zyklus

2.) Zucker:

1.) Zyklus: groß (größer)

2.) Zyklus: kleiner

1.) Zyklus: beim Pöken

in der Beiträge gehören

zusammen

01/24/2026
(c) Kunst. Vgds
Seite 6, Homepage

+++ Einschluss:

Homepage:

Könnte ein Fertigstellungsprozess
zyklus sein...

Jasodde auf facebook jeden
Tag...

1x auf Homepage...

und ...

dann gleich lin
alles weitere:

mipo 32
Qualitwerk
später bis dahin:
bleibt...

Handwritten notes and diagrams on a piece of paper, likely related to a technical or scientific project. The text is written in a cursive, handwritten style.

Diagrams:

- Top left: A small schematic diagram showing a box with a vertical line and a horizontal line, possibly representing a component or a connection.
- Top center: A small schematic diagram showing a box with a vertical line and a horizontal line, possibly representing a component or a connection.
- Top right: A small schematic diagram showing a box with a vertical line and a horizontal line, possibly representing a component or a connection.
- Middle left: A small schematic diagram showing a box with a vertical line and a horizontal line, possibly representing a component or a connection.
- Middle center: A small schematic diagram showing a box with a vertical line and a horizontal line, possibly representing a component or a connection.
- Middle right: A small schematic diagram showing a box with a vertical line and a horizontal line, possibly representing a component or a connection.

Text:

The text is written in a cursive, handwritten style. It appears to be a list of items or a set of instructions. The text is mostly illegible due to the handwriting and the angle of the paper.

Table:

A small table is visible on the right side of the page. It has two columns and several rows. The data is mostly illegible due to the handwriting.

01/23/2026

(c) David Vayda

Kleine Übung für zwischen

drin Seite 1

Die 44300 ~ Schaltnetz, Schaltfunktion, ...

Dim 40700 ~ Gatter... : Grundverkeüpfungen

2 in 40900

Lin 44300/93: Schaltnetz

" " / 89: Schaltfunktion

" 1181: Kodierglied...

iec60617-12? : Die modernen

↗ Schaltzeichen (grün ver-)

knüpfungen

aus anni 90 - 1984 : UY-American

He - Gable:

nand, nor, and, or, xor, negation,

arithmetic, demure, decode, muse,

Ord. (32) - 70. 7 - 229...

ft, parallel reg, seriell, ram, mono-flop

04, 05, 44: not, 32: or 08, 08, 11, 21:

83, 85, 181, 182, 183: ^{and} write, 7447: lrd fo-

nam: $16 \times 4 : 4$ bit wortleitung, $\frac{2 \text{ seg} \dots}{2}$

and: $\rightarrow$ or:

205

not: ok


$$= 1$$

$$-\frac{1}{2}$$
$$\begin{array}{ccc|c} a & b & c & \\ \hline 0 & 0 & 0 & \\ 0 & 1 & 1 & 205 \\ 1 & 1 & 1 & -1 \end{array} \oplus$$
$$\begin{vmatrix} a & b & c \\ 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{vmatrix} \text{ or}$$

05

a	b	c	and
0	0	0	
0	1	0	
1	0	0	
1	1	1	

$\frac{a}{i} \mid \frac{a}{o}$ invert

David Meyer (c)
 01/26/2026
 Seite 5, gal...
 + 1x2 müchfristung:
 pro olmc...
 8x 200...
 001 2 8
 000 01
 00 01
 Wert + -

David Ujda (c)

01/26/2026

Seite 4, gal...

200 möglichkeit:

not am ende
ein u. aus zu
schalten...

ok: jetzt 19 generell...
mit oder den existierenden...
Ausgang...

19 invertiert o. nicht

$\Rightarrow 28$

6 eingänge. jeden der 6 kann
mit 28

also: $8 \times 28 \Rightarrow$

224

01/24/2026
(c) David Vyda
fertigung: gal + ...
Seite 3

3.) Time exceeded:

Today: planning + co...
immer überlegen
(1.)

(2.) jetzt boulevard:

7:00 Uhr spätestens
6:30 Uhr besser
dann aufbauen

4.) fertigung faceboer:

Durchlauf...
jetzt Zigarette: Dann
Durchlauf
grob + zusatz...

5.) gal + ...

weg. : mir
man

01/24/2026

(1) David Vajda
fertigung von facebook-
beiträgen + gal +
min max term, Seite 1

- 1.) überlegen bis wann:
soll ich jetzt ausgerechnet
facebook weiter machen



- 1.) keine antwort
2.) ja: weil sonst ist nichts
da
3.) pausen zwischen aufträumen
wichtig...

2.) min + max term

a) auswendig lernen

b) min term: m

max term: M

n und v

Inf u. Sup...

produktterm

dazu: expt. term factor

David Lloyd (c)

01/25/2026

gal, Seite 1

8 eingänge an jedem ome...

generell 8 für and mit

8 eingänge...

o. 7 je nach mode.

- synchronisierbaren

u. die 8 an jedem and and

unabhängig der realen pair...

19 pair...

u.

das xer - am gal - wenn die

erfindung von fatische rot

vermutlich o. bewerte sein,

nicht das gal - o. freigele-

a.: das gal rot nicht empfind

erf, alles darüber...

21.01.26	m8counterledblinkslow01212026.asm	1
----------	-----------------------------------	---

```

;; (c) David Vajda
;; atmega 8, counter led + dnf - 4 Bit word
;; 01/21/2026

.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, 0x00
out PORTB, r16

l1:
out PORTB, r16
rcall c1
dec r16
rjmp l1

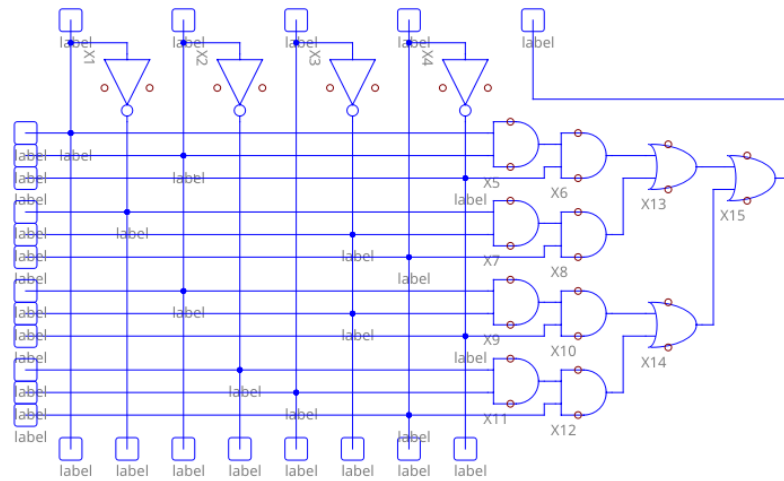
c1:
push r16
push r17
push r18
ldi r16, 0xff
l2:
ldi r17, 0xff
l3:
ldi r18, 0x04
l4:
dec r18
brne l4
dec r17
brne l3
dec r16
brne l2

pop r18
pop r17
pop r16
ret

```

file:///home/david/vhdl01102025/m8counterledblinkslow01212026.asm

label



```

10.01.26                               quine01102026.txt                               1
(C) David Vajda
Sat Jan 10 11:51:31 2026
4 Network - TTL - Disjunktive Normalform

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

1 0 0 0 1 1 1
3 0 0 1 1 1 1
4 0 1 0 0 1 1
5 0 1 0 1 1 1
11 1 0 1 1 1 1
12 1 1 0 0 1 1
14 1 1 1 0 1 1

Gruppe 1:
1 0 0 0 1 1 1
4 0 1 0 0 1 1
Gruppe 2:
3 0 0 1 1 1 1
5 0 1 0 1 1 1
12 1 1 0 0 1 1
Gruppe 3:
11 1 0 1 1 1 1
14 1 1 1 0 1 1

1:3      0 0 - 1
1:5      0 - 0 1
4:5      0 1 0 -
4:12     - 1 0 0
3:11     - 0 1 1
12:14    1 1 - 0

4:5      0 1 0 -
1:3      0 0 - 1
12:14    1 1 - 0
1:5      0 - 0 1
4:12     - 1 0 0
3:11     - 0 1 1

Gruppe 1:
4:5      0 1 0 -

Gruppe 1:

```

file:///home/david/vhdl01102025/quine01102026.txt


```

10.01.26 quine01102026.txt 2
1:3      0  0  -  1
Gruppe 2:
12:14    1  1  -  0

Gruppe 1:
1:5      0  -  0  1

Gruppe 1:
4:12    -  1  0  0
Gruppe 2:
3:11    -  0  1  1

Primimplikantentafel:

      1  3  4  5  11  12  14
4:5      +  +
1:3      +  +
12:14     +  +
1:5      +  +
4:12     +  +
3:11     +  +

      1  3  4  5  11  12  14
12:14     +  +      p
1:5      +  +
4:12     +  +
3:11     +  +      p

OK, verkuerzt:

12:14    1  1  -  0
1:5      0  -  0  1
4:12    -  1  0  0
3:11    -  0  1  1

y <= (x3 and x2 and not x0) and
      (not x3 and not x1 and x0) and
      (x2 and not x1 and not x0) and
      (not x2 and x1 and x0);

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026 is
port (
    x3, x2, x1, x0: in std_logic;
    y: out std_logic
);
end;

architecture behaviour of quine01102026 is
begin
    y <= (x3 and x2 and not x0) or
          (not x3 and not x1 and x0) or
          (x2 and not x1 and not x0) or
          (not x2 and x1 and x0);
end;

library ieee;

```

file:///home/david/vhdl01102025/quine01102026.txt

10.01.26	quine01102026.txt	3
----------	-------------------	---

```
use ieee.std_logic_1164.all;

entity quine01102026testbench is
port (
    y: inout std_logic
);
end;

architecture behaviour of quine01102026testbench is
    component quine01102026
    port (
        x3, x2, x1, x0: in std_logic;
        y: out std_logic;
    );
    end component;
    signal x3, x2, x1, x0: std_logic;
begin
    q: quine01102026 PORT MAP (x3=>x3, x2=>x2, x1=>x1, x0=>x0, y=>y);
```

file:///home/david/vhdl01102025/quine01102026.txt

```

10.01.26 quine01102026.vhdl 1
-- (C) David Vajda
-- Sat Jan 10 11:51:31 2026
-- 4 Network - TTL - Disjunktive Normalform

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026 is
port (
    x3, x2, x1, x0: in std_logic;
    y: inout std_logic
);
end;

architecture behaviour of quine01102026 is
begin
    y <= (x3 and x2 and not x0) or
        (not x3 and not x1 and x0) or
        (x2 and not x1 and not x0) or
        (not x2 and x1 and x0);

end;

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026testbench is
port (
    y: inout std_logic
);
end;

architecture behaviour of quine01102026testbench is
    component quine01102026
    port (
        x3, x2, x1, x0: in std_logic;
        y: inout std_logic
    );
    end component;
    signal x3, x2, x1, x0: std_logic;
begin
    q: quine01102026 PORT MAP (x3=>x3, x2=>x2, x1=>x1, x0=>x0, y=>y);
    x0 <= '0' after 0 ns, '1' after 10 ns, '0' after 20 ns, '1' after 30 ns, '0' after 40 ns,
        '1' after 50 ns, '0' after 60 ns, '1' after 70 ns, '0' after 80 ns, '1' after 90 ns, '0'
        after 100 ns, '1' after 110 ns, '0' after 120 ns, '1' after 130 ns, '0' after 140 ns, '1'
        after 150 ns;

    x1 <= '0' after 0 ns, '0' after 10 ns, '1' after 20 ns, '1' after 30 ns, '0' after 40 ns,
        '0' after 50 ns, '1' after 60 ns, '1' after 70 ns, '0' after 80 ns, '0' after 90 ns, '1'
        after 100 ns, '1' after 110 ns, '0' after 120 ns, '0' after 130 ns, '1' after 140 ns, '1'
        after 150 ns;

    x2 <= '0' after 0 ns, '0' after 10 ns, '0' after 20 ns, '0' after 30 ns, '1' after 40 ns,
        '1' after 50 ns, '1' after 60 ns, '1' after 70 ns, '0' after 80 ns, '0' after 90 ns, '0'
        after 100 ns, '0' after 110 ns, '1' after 120 ns, '1' after 130 ns, '1' after 140 ns, '1'
        after 150 ns;

    x3 <= '0' after 0 ns, '0' after 10 ns, '0' after 20 ns, '0' after 30 ns, '0' after 40 ns,
        '0' after 50 ns, '0' after 60 ns, '0' after 70 ns, '1' after 80 ns, '1' after 90 ns, '1'
        after 100 ns, '1' after 110 ns, '1' after 120 ns, '1' after 130 ns, '1' after 140 ns, '1'
        after 150 ns;
end;

```

file:///home/david/vhdl01102025/quine01102026.vhdl

10.01.26	quine01102026.vhdl	2
----------	--------------------	---

end;

file:///home/david/vhdl01102025/quine01102026.vhdl


```

10.01.26 quine01102026.vhdl 1
-- (C) David Vajda
-- Sat Jan 10 11:51:31 2026
-- 4 Network - TTL - Disjunktive Normalform

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026 is
port (
    x3, x2, x1, x0: in std_logic;
    y: inout std_logic
);
end;

architecture behaviour of quine01102026 is
begin
    y <= (x3 and x2 and not x0) or
        (not x3 and not x1 and x0) or
        (x2 and not x1 and not x0) or
        (not x2 and x1 and x0);

end;

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026testbench is
port (
    y: inout std_logic
);
end;

architecture behaviour of quine01102026testbench is
    component quine01102026
    port (
        x3, x2, x1, x0: in std_logic;
        y: inout std_logic
    );
    end component;
    signal x3, x2, x1, x0: std_logic;
begin
    q: quine01102026 PORT MAP (x3=>x3, x2=>x2, x1=>x1, x0=>x0, y=>y);
    x0 <= '0' after 0 ns, '1' after 10 ns, '0' after 20 ns, '1' after 30 ns, '0' after 40 ns,
        '1' after 50 ns, '0' after 60 ns, '1' after 70 ns, '0' after 80 ns, '1' after 90 ns, '0'
        after 100 ns, '1' after 110 ns, '0' after 120 ns, '1' after 130 ns, '0' after 140 ns, '1'
        after 150 ns;

    x1 <= '0' after 0 ns, '0' after 10 ns, '1' after 20 ns, '1' after 30 ns, '0' after 40 ns,
        '0' after 50 ns, '1' after 60 ns, '1' after 70 ns, '0' after 80 ns, '0' after 90 ns, '1'
        after 100 ns, '1' after 110 ns, '0' after 120 ns, '0' after 130 ns, '1' after 140 ns, '1'
        after 150 ns;

    x2 <= '0' after 0 ns, '0' after 10 ns, '0' after 20 ns, '0' after 30 ns, '1' after 40 ns,
        '1' after 50 ns, '1' after 60 ns, '1' after 70 ns, '0' after 80 ns, '0' after 90 ns, '0'
        after 100 ns, '0' after 110 ns, '1' after 120 ns, '1' after 130 ns, '1' after 140 ns, '1'
        after 150 ns;

    x3 <= '0' after 0 ns, '0' after 10 ns, '0' after 20 ns, '0' after 30 ns, '0' after 40 ns,
        '0' after 50 ns, '0' after 60 ns, '0' after 70 ns, '1' after 80 ns, '1' after 90 ns, '1'
        after 100 ns, '1' after 110 ns, '1' after 120 ns, '1' after 130 ns, '1' after 140 ns, '1'
        after 150 ns;
end;

```

file:///home/david/vhdl01102025/quine01102026.vhdl

10.01.26	quine01102026.vhdl	2
----------	--------------------	---

end;

file:///home/david/vhdl01102025/quine01102026.vhdl

quine mc cluskey

David Vajda

12. Januar 2026

(C) David Vajda Sat Jan 10 11:51:31 2026 4 Network - TTL - Disjunktive Normalform

nr.	x_3	x_2	x_1	x_0	y
0	f	f	f	f	f
1	f	f	f	w	w
2	f	f	w	f	f
3	f	f	w	w	w
4	f	w	f	f	w
5	f	w	f	w	w
6	f	w	w	f	f
7	f	w	w	w	f
8	w	f	f	f	f
9	w	f	f	w	f
10	w	f	w	f	f
1	w	f	w	w	w
12	w	w	f	f	w
13	w	w	f	w	f
14	w	w	w	f	w
15	w	w	w	w	f

1 Ein wenig von dem was uns zu erwarten hätte

...

ich musste den text auf grund seiner laenge streichen... keine sorge.... vergessen sie nicht - darstellung ist wichtig... ein paar hinweise

jetzt gleich 1.) hauptsache: lange mathematischer text zum auswendig lernen + physik am ende teilweise nicht richtig aber doch nicht unwahr, am anfang ganz fuehrt zum richtigen =, als pdf 2.) dazu extrem viele uebungen - computersysteme 1 u. 2 schaltung.

- uebung - uebung - uebung - etc. wdh.

2.) will nicht vorenthalten: keine homepage, keine gueter, sondern: pdf book without quatschen ... und: icloud link + pdf book vielleicht wenn finden = (german) icloud link

3.) dazu chemie 3.1.) laenge: punkt, linie, finger = zoll = inch, handbreit, fuss, ... 3.2.) cent, gran, karat, skrupel... 3.3.) periodensystem lernen: H, He, Li, Be, B, C, N, O, F, Ne, Na, Mg, Al, Si, P, S, Cl, ... Au und Ag!!! Cu und Fe... fuehrt zu: ingenieure tabelle lernen, please - trotz statistischer auswertung!!!

3.4.) Mol: 6.022×10^{23} oder 22 teilchen molare masse: gramm pro mol
molares volumen: liter pro mol stoffmengenkonzentration: mol/liter
why: easy: merke: guenstig/moeglich guenstig pro moeglich
moeglichst viel, guenstigst klein
klein oben, gross unten g/m ok und gramm/mol
perfekt:
4.) benzin und wasser auf die wage 5.) weitere versuche:
1.) die schwerheit in der naturwissenschaft: cyankali
deutsch: blaukraut bleibt blaukraut und brautkleid bleibt brautkleid...
why? wg. beim kochen, in blaukraut, nikotin, aber bittermandel, riecht suess,
ist blausaure
cyan - fuehrt zu: gelbes, rotes, weisses, ... kreuz: moderne
what is cyankali
kali bergwerk: leute arbeiten, die zu viel energie haben... ohne boese, aber
spaeter. und ist blausaure weiss durchsichtig: sind kalium pulver darin binde-
mittel??
die seife
reaktion im alltag anders: einfach butter oder oel zu aetznatron
vorsichtig: sehr gefaehrlich: ducken im versuch am anfang normal!!!! sehr sehr
sehr gefaehrlich
trotzdem: einfach zu sammen tun: james alder reaktion? what ever: zusam-
men tun: cyankali wahrscheinlich aehnlich
im netz... eine beschreibung, mit destille, aber nicht mit wasser sicher mach-
bar?
easy? denkste, lsa und diethylamid sicher auch oder nicht, fuer chemiker oder
fuer nicht chemiker, oder fuer gar nicht? keine ahnung
was hat das mit wissenschaft zu tun? zu viel geredet, in der soc waschkueche?
viele schaften es vorher nicht
waere wunder...
wasser destillieren: nietzsche wasser tut es auch? bei was? was ist die stapel-
einheit 1024? am ende darueber?
gut, richtig verstanden oder nicht: Au Gold und Ag Silber dazu Cupfer und
so weiter..
das waere auch eine schoene uebung oekonomisch nutzbar
alles in pdf, wollte ich sagen, nachher
nachher, icloud
versuch in pdf reader unter zu bringen
ueben ueben ueben... jetzt uebung skript, weil 5V adapter weg.. nachher
schaltung...

2 quine mc cluskey

nr.	x_3	x_2	x_1	x_0	y
0	f	f	f	f	f
1	f	f	f	w	w
2	f	f	w	f	f
3	f	f	w	w	w
4	f	w	f	f	w
5	f	w	f	w	w
6	f	w	w	f	f
7	f	w	w	w	f
8	w	f	f	f	f
9	w	f	f	w	f
10	w	f	w	f	f
11	w	f	w	w	w
12	w	w	f	f	w
13	w	w	f	w	f
14	w	w	w	f	w
15	w	w	w	w	f

$y \leq (x_3 \text{ and } x_2 \text{ and not } x_0) \text{ or}$
 $(\text{not } x_3 \text{ and not } x_1 \text{ and } x_0) \text{ or}$
 $(x_2 \text{ and not } x_1 \text{ and not } x_0) \text{ or}$
 $(\text{not } x_2 \text{ and } x_1 \text{ and } x_0);$

$$y \leq (x_3 \text{ wedge } x_2 \text{ wedge } \neg x_0) \vee (\neg x_3 \text{ wedge } \neg x_1 \text{ wedge } x_0) \vee (x_2 \text{ wedge } \neg x_1 \text{ wedge } \neg x_0) \vee (\neg x_2 \text{ wedge } x_1 \text{ wedge } x_0);$$

Daraus wird: ... gerade ungeduldet

3 wie oft wiederholt sich die uebung...

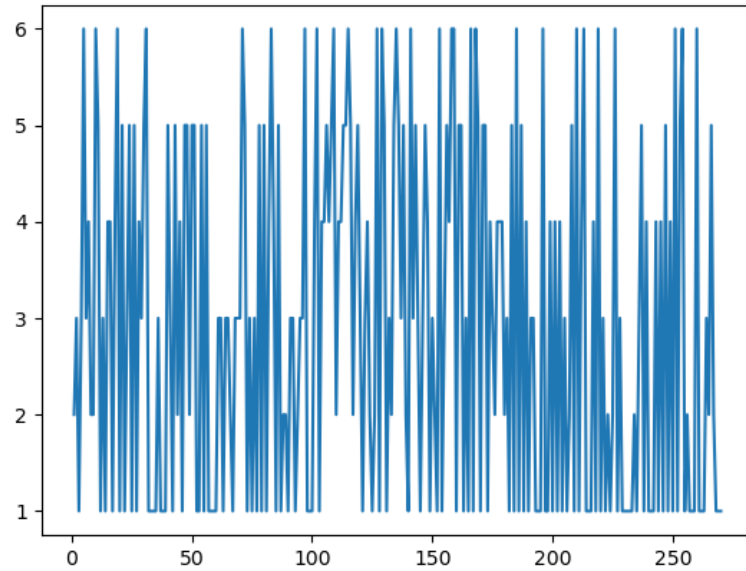
3 Bit, Eingangsvariablen: $2^3 = 8$, ok, also 8 bit ausgang, wie viele moeglichkeiten gibt es? $8! = 8 \cdot 7 \cdot \dots \cdot 2 \cdot 1$???, nein, da wir nicht haben, permutation! $Y = \{8, 7, 6, \dots, 2, 1\}$, also $\{7, 8, \dots\}$, ..., sondern $Y = (0, 0, 0, \dots, 0) \dots (1, 1, 1, \dots, 1)$ ist das ein byte, bei 2^6 naechlich eingangsvektor: 4 bit haben wir: $64k$ moeglichkeiten, also $64k + 256$ und jetzt frage: waerscheinlichkeitsrechnung, wie oft kommt die selbe uebung?

also: $64k + 256$ sind durch 365 zu teilen, bei einer uebung am tag, wenn man aber nur alle 2 tage eine neue uebung, sind das: $\frac{64k + 256}{365} \cdot 2$ sind: 360,504109589 also, damit sich die uebungen wiederholen, waeren, 360 jahre notwendig, werden nur 3 Bit verwendet, weniger als ein jahr, etwa 0.7 jahre. ok, doch wie gross ist die waerscheinlichkeit??? erwartungswert nutzt hier nichts, ich kann den durchschnittswert ausrechnen, wert von 0..255 und jedes waerscheinlichkeit von $\frac{1}{256}$

$$\sum_{i=0}^{255} \frac{1}{256} \cdot i$$

(c) David Vajda
Quine Text
01/21/2026

0 0 0 0 0 0
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 0
7 0 1 1 1 0
8 1 0 0 0 0
9 1 0 0 1 0
10 1 0 1 0 0
11 1 0 1 1 1
12 1 1 0 0 1
13 1 1 0 1 0
14 1 1 1 0 1
15 1 1 1 1 0



```

Datei  Bearbeiten  Ansicht  Lesezeichen  Module  Einstellungen  Hilfe
Neues Unterfenster  Ansicht teilen  Kopieren  Einfügen  Suchen

+2 3 1 5 1 6 1 5 1 4 1 3 3 1 1 1 6 1 1 4 1 4 1 4 1 3 1 2 ,5 1
1 4 ,6 1 1 1 4 1 6 1 3 1 2 1 2 ,6 1 3 1 1 1 1 1 2 1 3 ,5 1 4 1 1 1
1 4 1 5 1 4 1 6 1 5 ,6 1 2 1 1 , 1 ,6 1 1 1 3 ,2 ,5 ,2 1 1 1
david@www001:~$ /bin/bash rand12152025sed.sh rand12152025raw.txt

2 3 1 3 6 3 4 2 2 6 5 1 3 1
4 4 1 4 6 1 5 1 3 5 1 5 1 4 3 5 6 1 1 1 7 1 3 1
1 1 5 3 1 5 2 4 1 5 5 2 5 5 1 1
5 1 5 1 7 1 7 1 7 1 3 3 1 3 3 2 1 3 3 3
8 6 5 1 0 3 1 3 1 5 1 5 1 4 6 4 1 5 1 2 2 1 3 7 3
1 2 3 3 6 1 1 1 4 6 1 4 4 5 4 5 6 2 4 4 5 5 6 5 2 4 5 3 1 3
4 2 1 2
6 1 6 8 5 1 3 2 5 6 5 3 5 2 1 6 3 5 3 1 3 5 4 8 1 3 2 1 6 1 3
5 4
6 6 1 5 5 1 3 1 6 0 1 6 5 1 5 5 1 4 3 2 4 4
4 2 3 1 5 1 6 1 5 1 4 1 3 3 1 1 1 6 1 1 4 1 4 1 4 1 3 1 2 5 1
6 1 4 6 1 1 1 4 1 6 1 3 1 2 1 2 6 1 3 1 1 1 1 2 1 3 5 1 4 1 1 1
4 1 4 1 5 1 4 1 6 1 5 6 1 2 1 1 1 6 1 1 1 3 2 5 2 1 1 1
david@www001:~$

```

```

Datei Bearbeiten Ansicht Lesezeichen Module Einstellungen Hilfe
Neues Unterfenster Ansicht teilen Kopieren Einfügen Suchen

4 1 4 1 5 1 4 1 6 1 5 6 1 2 1 1 1 6 1 1 1 3 2 5 2 1 1 1
david@www001:~$ /bin/bash rand12152025sed.sh rand12152025raw.txt
sed: -e Ausdruck #1, Zeichen 15: »(« oder »\ (« ohne schließende Klammer
david@www001:~$
david@www001:~$ /bin/bash rand12152025sed.sh rand12152025raw.txt
,
, 2, 3, 1, 3, 6, 3, 4, 2, 2, 6, 5, 1, 3, 1,
, 4, 4, 1, 4, 6, 1, 5, 1, 3, 5, 1, 5, 1, 4, 3, 5, 6, 1, 1, 1, 1, 3, 1,
, 1, 1, 5, 3, 1, 5, 2, 4, 1, 5, 5, 2, 5, 5, 1, 1,
, 5, 1, 5, 1, 1, 1, 1, 3, 3, 1, 3, 3, 2, 1, 3, 3, 3,
, 6, 5, 1, 3, 1, 3, 1, 5, 1, 5, 1, 4, 6, 4, 1, 5, 1, 2, 2, 1, 3, 3,
, 1, 2, 3, 3, 6, 1, 1, 1, 4, 6, 1, 4, 4, 5, 4, 5, 6, 2, 4, 4, 5, 5, 6, 5, 2, 4, 5, 3, 1, 3, 4, 2, 1, 2,
,
, 6, 1, 6, 5, 1, 3, 2, 5, 6, 5, 3, 5, 2, 1, 6, 3, 5, 3, 1, 3, 5, 4, 1, 3, 2, 1, 6, 1, 3, 5, 4,
, 6, 6, 1, 5, 5, 1, 3, 1, 6, 1, 6, 5, 1, 5, 5, 1, 4, 3, 2, 4, 4,
, 4, 2, 3, 1, 5, 1, 6, 1, 5, 1, 4, 1, 3, 3, 1, 1, 1, 6, 1, 1, 4, 1, 4, 1, 4, 1, 3, 1, 2, 5, 1,
, 6, 1, 4, 6, 1, 1, 1, 4, 1, 6, 1, 3, 1, 2, 1, 2, 6, 1, 3, 1, 1, 1, 1, 1, 2, 1, 3, 5, 1, 4, 1, 1, 1,
, 4, 1, 4, 1, 5, 1, 4, 1, 6, 1, 5, 6, 1, 2, 1, 1, 1, 6, 1, 1, 1, 3, 2, 5, 2, 1, 1, 1,
david@www001:~$

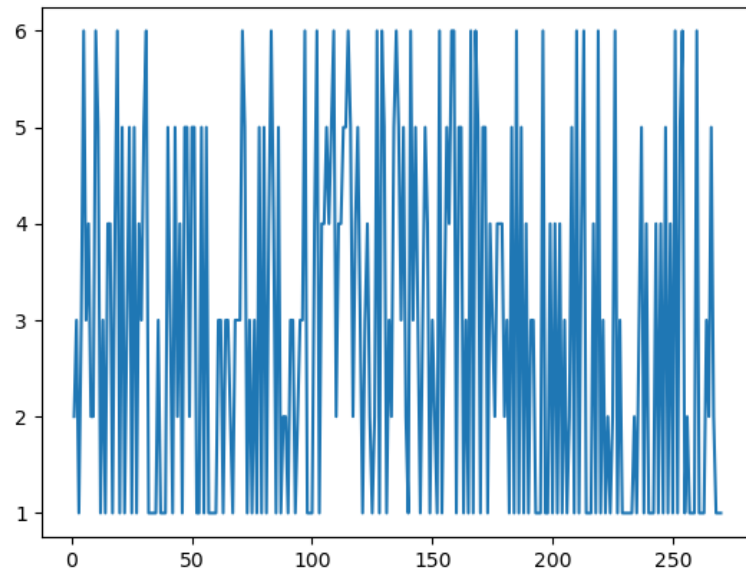
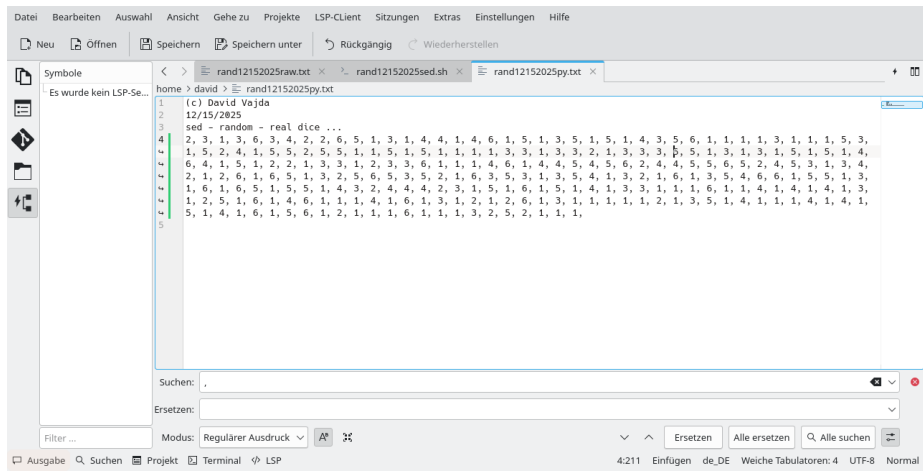
Datei Bearbeiten Ansicht Lesezeichen Module Einstellungen Hilfe
Neues Unterfenster Ansicht teilen Kopieren Einfügen Suchen

david@www001:~$ /bin/bash rand12152025sed.sh rand12152025raw.txt
(c) David Vajda
12/15/2025
sed - random - real dice ...
,
, 2, 3, 1, 3, 6, 3, 4, 2, 2, 6, 5, 1, 3, 1,
, 4, 4, 1, 4, 6, 1, 5, 1, 3, 5, 1, 5, 1, 4, 3, 5, 6, 1, 1, 1, 1, 3, 1,
, 1, 1, 5, 3, 1, 5, 2, 4, 1, 5, 5, 2, 5, 5, 1, 1,
, 5, 1, 5, 1, 1, 1, 1, 3, 3, 1, 3, 3, 2, 1, 3, 3, 3,
, 6, 5, 1, 3, 1, 3, 1, 5, 1, 5, 1, 4, 6, 4, 1, 5, 1, 2, 2, 1, 3, 3,
, 1, 2, 3, 3, 6, 1, 1, 1, 4, 6, 1, 4, 4, 5, 4, 5, 6, 2, 4, 4, 5, 5, 6, 5, 2, 4, 5, 3, 1, 3, 4, 2, 1, 2,
,
, 6, 1, 6, 5, 1, 3, 2, 5, 6, 5, 3, 5, 2, 1, 6, 3, 5, 3, 1, 3, 5, 4, 1, 3, 2, 1, 6, 1, 3, 5, 4,
, 6, 6, 1, 5, 5, 1, 3, 1, 6, 1, 6, 5, 1, 5, 5, 1, 4, 3, 2, 4, 4,
, 4, 2, 3, 1, 5, 1, 6, 1, 5, 1, 4, 1, 3, 3, 1, 1, 1, 6, 1, 1, 4, 1, 4, 1, 4, 1, 3, 1, 2, 5, 1,
, 6, 1, 4, 6, 1, 1, 1, 4, 1, 6, 1, 3, 1, 2, 1, 2, 6, 1, 3, 1, 1, 1, 1, 1, 2, 1, 3, 5, 1, 4, 1, 1, 1,
, 4, 1, 4, 1, 5, 1, 4, 1, 6, 1, 5, 6, 1, 2, 1, 1, 1, 6, 1, 1, 1, 3, 2, 5, 2, 1, 1, 1,
david@www001:~$ /bin/bash rand12152025sed.sh rand12152025raw.txt > rand12152025py.txt
david@www001:~$

Datei Bearbeiten Auswahl Ansicht Gehe zu Projekte LSP-Client Sitzungen Extras Einstellungen Hilfe
Neu Öffnen Speichern Speichern unter Rückgängig Wiederherstellen

Symbole home > david > rand12152025py.txt
1 (c) David Vajda
2 12/15/2025
3 sed - random - real dice ...
4
5
6 , 2, 3, 1, 3, 6, 3, 4, 2, 2, 6, 5, 1, 3, 1,
7 , 4, 4, 1, 4, 6, 1, 5, 1, 3, 5, 1, 5, 1, 4, 3, 5, 6, 1, 1, 1, 1, 3, 1,
8 , 1, 1, 5, 3, 1, 5, 2, 4, 1, 5, 5, 2, 5, 5, 1, 1,
9 , 5, 1, 5, 1, 1, 1, 1, 3, 3, 1, 3, 3, 2, 1, 3, 3, 3,
10 , 6, 5, 1, 3, 1, 3, 1, 5, 1, 5, 1, 4, 6, 4, 1, 5, 1, 2, 2, 1, 3, 3,
11 , 1, 2, 3, 3, 6, 1, 1, 1, 4, 6, 1, 4, 4, 5, 4, 5, 6, 2, 4, 4, 5, 5, 6, 5, 2, 4, 5, 3, 1, 3, 4, 2, 1, 2,
12 , 6, 1, 6, 5, 1, 3, 2, 5, 6, 5, 3, 5, 2, 1, 6, 3, 5, 3, 1, 3, 5, 4, 1, 3, 2, 1, 6, 1, 3, 5, 4,
13 , 6, 6, 1, 5, 5, 1, 3, 1, 6, 1, 6, 5, 1, 5, 5, 1, 4, 3, 2, 4, 4,
14 , 4, 2, 3, 1, 5, 1, 6, 1, 5, 1, 4, 1, 3, 3, 1, 1, 1, 6, 1, 1, 4, 1, 4, 1, 4, 1, 3, 1, 2, 5, 1,
15 , 6, 1, 4, 6, 1, 1, 1, 4, 1, 6, 1, 3, 1, 2, 1, 2, 6, 1, 3, 1, 1, 1, 1, 2, 1, 3, 5, 1, 4, 1, 1, 1,
16 , 4, 1, 4, 1, 5, 1, 4, 1, 6, 1, 5, 6, 1, 2, 1, 1, 1, 6, 1, 1, 1, 3, 2, 5, 2, 1, 1, 1,
17

```




```
Datei Bearbeiten Ansicht Lesezeichen Module Einstellungen Hilfe
Neues Unterfenster Ansicht teilen Kopieren Einfügen Suchen

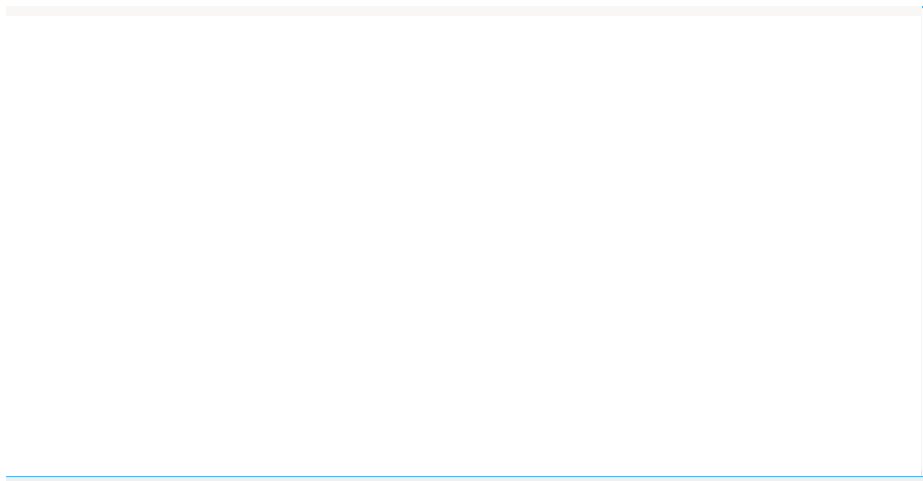
.gitconfig Screenshot_20251215_095521.png
.gnupg/ Screenshot_20251215_095534.png
.gnuplot-wxt Screenshot_20251215_100222.png
.gtkrc-2.0 Screenshot_20251215_100228.png
.idlrc/ Screenshot_20251215_100503.png
.lessshst Screenshot_20251215_100529.png
.lmmsrc.xml Screenshot_20251215_101017.png
.local/ Screenshot_20251215_101554.png
david@www001:~$ python3 rand12152025
rand12152025py.py rand12152025py.txt rand12152025raw.txt rand12152025sed.sh
david@www001:~$ python3 rand12152025
rand12152025py.py rand12152025py.txt rand12152025raw.txt rand12152025sed.sh
david@www001:~$ python3 rand12152025py.py
Traceback (most recent call last):
  File "/home/david/rand12152025py.py", line 15, in <module>
    plot.plot(X, Y)
    ^^^^^
NameError: name 'plot' is not defined. Did you mean: 'plt'?
david@www001:~$ python3 rand12152025py.py

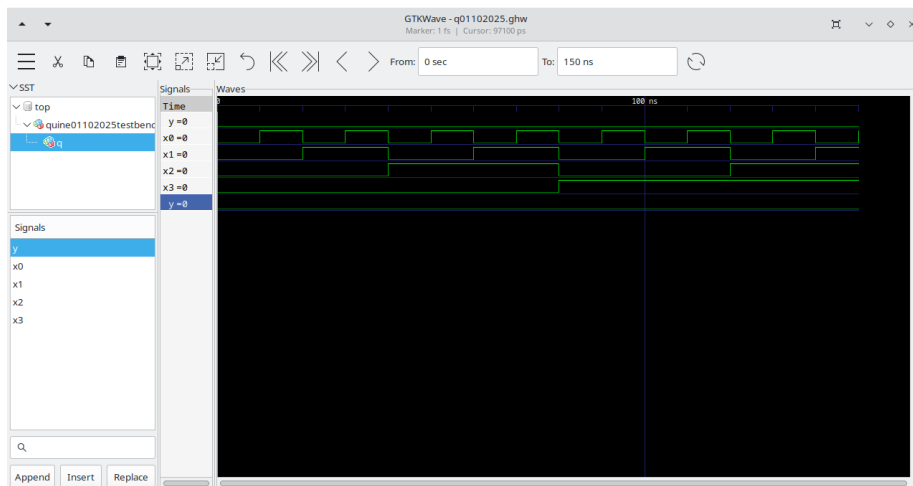
Datei Bearbeiten Ansicht Lesezeichen Module Einstellungen Hilfe
Neues Unterfenster Ansicht teilen Kopieren Einfügen Suchen

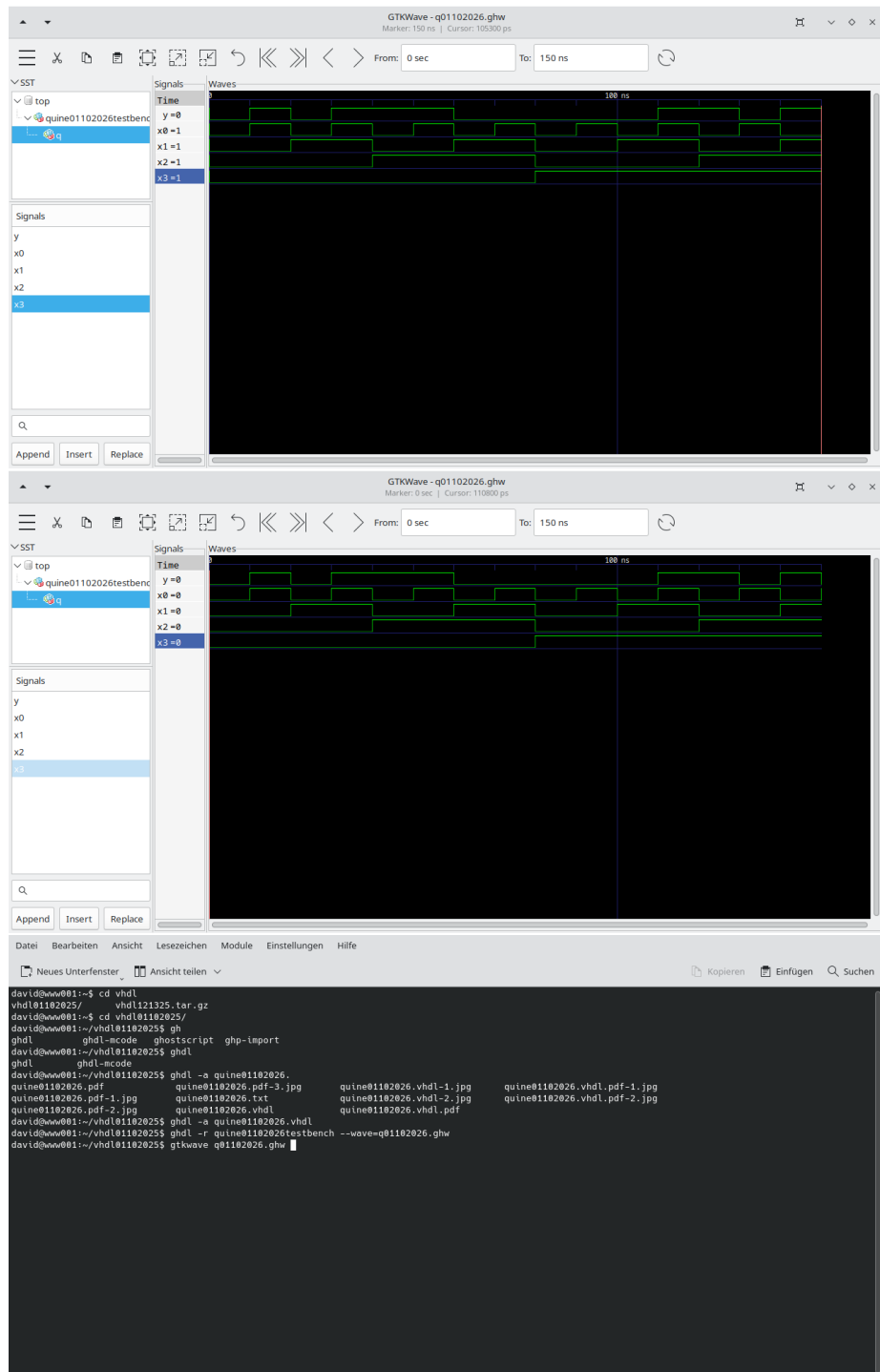
absolute haeufigkeit: 6 : 28
david@www001:~$ python3 rand12152025py.py
  File "/home/david/rand12152025py.py", line 20
    relh []
    ^
SyntaxError: invalid syntax
david@www001:~$ python3 rand12152025py.py
absolute haeufigkeit: 1 : 103
absolute haeufigkeit: 2 : 25
absolute haeufigkeit: 3 : 40
absolute haeufigkeit: 4 : 32
absolute haeufigkeit: 5 : 42
absolute haeufigkeit: 6 : 28
absolute haeufigkeit: 1 : 103 relative: 0.3814814814814815
absolute haeufigkeit: 2 : 25 relative: 0.09259259259259259
absolute haeufigkeit: 3 : 40 relative: 0.14814814814814814
absolute haeufigkeit: 4 : 32 relative: 0.11851851851851852
absolute haeufigkeit: 5 : 42 relative: 0.15555555555555556
absolute haeufigkeit: 6 : 28 relative: 0.1037037037037037
david@www001:~$

Datei Bearbeiten Ansicht Lesezeichen Module Einstellungen Hilfe
Neues Unterfenster Ansicht teilen Kopieren Einfügen Suchen

absolute haeufigkeit: 3 : 40 relative: 0.14814814814814814
absolute haeufigkeit: 4 : 32 relative: 0.11851851851851852
absolute haeufigkeit: 5 : 42 relative: 0.15555555555555556
absolute haeufigkeit: 6 : 28 relative: 0.1037037037037037
Erwartungswert: laplace, 3.5 erwartungswert erfuehlt: 1.0
david@www001:~$ python3 rand12152025py.py
absolute haeufigkeit: 1 : 103
absolute haeufigkeit: 2 : 25
absolute haeufigkeit: 3 : 40
absolute haeufigkeit: 4 : 32
absolute haeufigkeit: 5 : 42
absolute haeufigkeit: 6 : 28
absolute haeufigkeit: 1 : 103 relative: 0.3814814814814815
absolute haeufigkeit: 2 : 25 relative: 0.09259259259259259
absolute haeufigkeit: 3 : 40 relative: 0.14814814814814814
absolute haeufigkeit: 4 : 32 relative: 0.11851851851851852
absolute haeufigkeit: 5 : 42 relative: 0.15555555555555556
absolute haeufigkeit: 6 : 28 relative: 0.1037037037037037
Erwartungswert: laplace, 3.5 erwartungswert erfuehlt: 2.8851851851851853
david@www001:~$
```







```
David@www001:~$ cd vhd1
vhd1@102025/~$ vhd1121325.tar.gz
David@www001:~$ cd vhd101102025/
David@www001:~/vhd101102025$ gh
ghdl ghdl-mcode ghstscript ghp-import
David@www001:~/vhd101102025$ ghdl
ghdl ghdl-mcode
David@www001:~/vhd101102025$ ghdl -a quine01102026.
quine01102026.pdf quine01102026.pdf-3.jpg quine01102026.vhdl-1.jpg quine01102026.vhdl.pdf-1.jpg
quine01102026.pdf-1.jpg quine01102026.txt quine01102026.vhdl-2.jpg quine01102026.vhdl.pdf-2.jpg
quine01102026.pdf-2.jpg quine01102026.vhdl
David@www001:~/vhd101102025$ ghdl -a quine01102026.vhdl
David@www001:~/vhd101102025$ ghdl -r quine01102026testbench --wave-q01102026.gwh
David@www001:~/vhd101102025$ gtwave q01102026.gwh
```

```
AVRA: advanced AVR macro assembler (version 1.4.2)
Pass 1...
Pass 2...
/usr/share/avra/m8def.inc(44) : PRAGMA PARTINC directive currently ignored
/usr/share/avra/m8def.inc(48) : PRAGMA AVRPART directive currently ignored
/usr/share/avra/m8def.inc(53) : PRAGMA AVRPART directive currently ignored
/usr/share/avra/m8def.inc(690) : PRAGMA AVRPART directive currently ignored
/usr/share/avra/m8def.inc(691) : PRAGMA AVRPART directive currently ignored
/usr/share/avra/m8def.inc(692) : PRAGMA AVRPART directive currently ignored
/usr/share/avra/m8def.inc(693) : PRAGMA AVRPART directive currently ignored
/usr/share/avra/m8def.inc(734) : PRAGMA AVRPART directive currently ignored
done

Assembly complete with no errors.
Segment usage:
Code      :      28 words (56 bytes)
Data      :       0 bytes
EEPROM    :       0 bytes
David@www001:~/vhd101102025$
```

PROG-EXPRESS

Projekt

Programmieren

Batronic BX48 Batego II
Seriennummer: PF0910262
Firmware Version: 02.13 Aktualisieren

Atmega8 ISP Atmega8A ISP
Memory: 8 KByte (64 KBit), Data EEPROM: 512 Byte (4 KBit),
Fuses and Lock Bits: 8 Byte (64 Bit) Chip Autoerkennung

m8counterdblinkslow01212026.hex
Pfad: /home/david/vhd101102025/ Durchsuchen

Chip Optionen (optional)
Keine speziellen Optionen aktiviert Einstellen

Seriennummern (optional)
Keine Seriennummern einfügen Einstellen

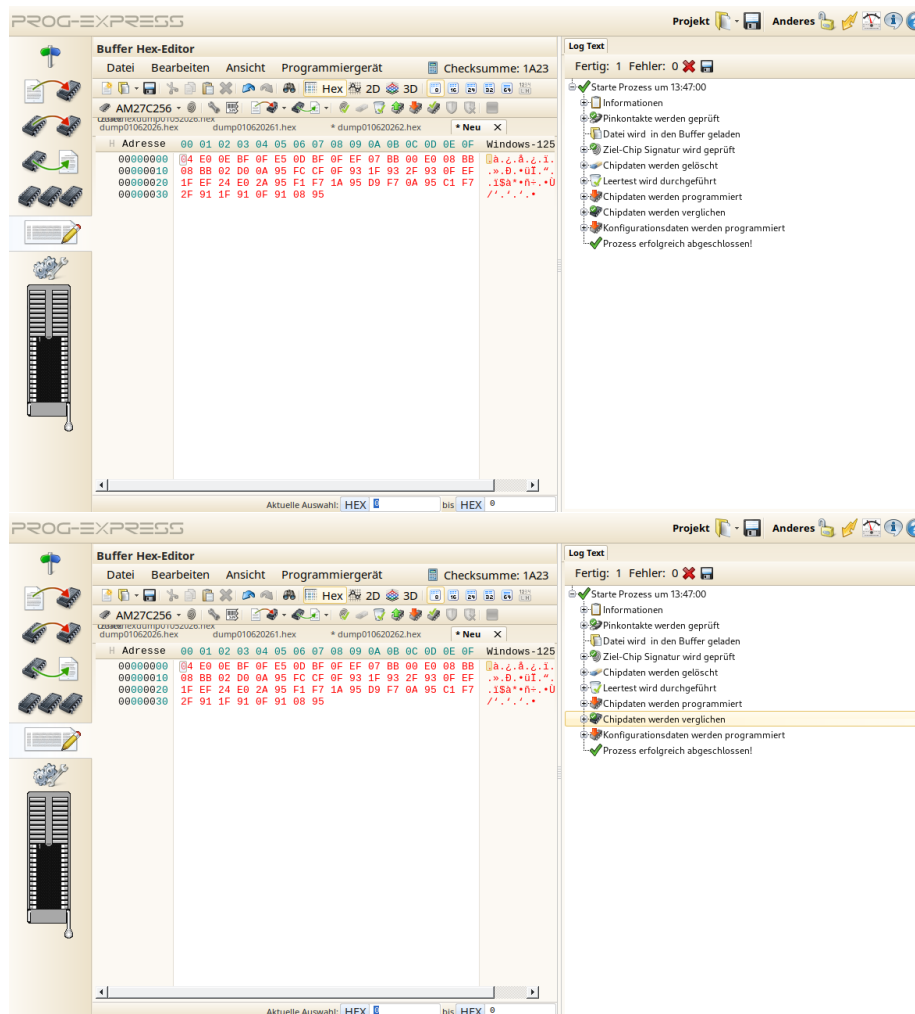
Programmierungsprozess starten Kopieren

- ☒ Ziel-Chip Signatur prüfen
- ☒ 'Protection Bits' entfernen
- ☒ Chipdaten löschen
- ☒ Leertest
- ☒ Chipdaten programmieren
- ☒ Chipdaten vergleichen
- ☒ 'Protection Bits' setzen
- ☒ Chip Optionen vergleichen

Log Text

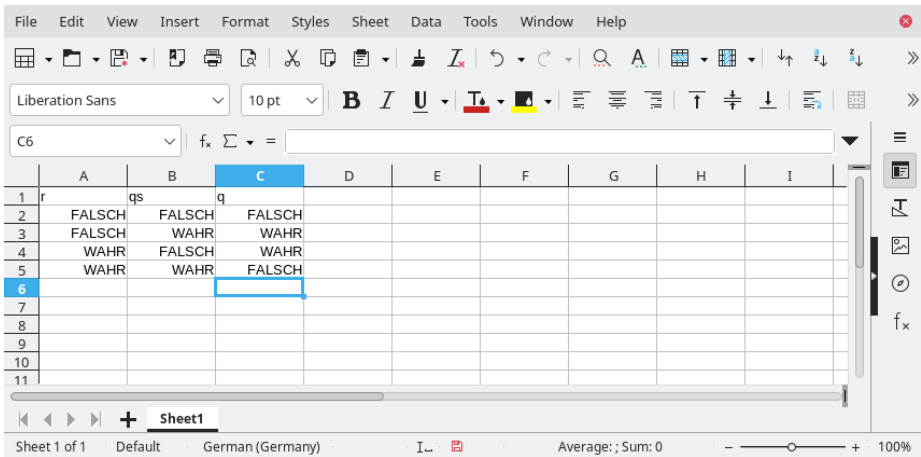
Fertig: 1 Fehler: 0

- Starte Prozess um 13:47:00
- Informationen
- Pinkontakte werden geprüft
- Datei wird in den Buffer geladen
- Ziel-Chip Signatur wird geprüft
- Chipdaten werden gelöscht
- Leertest wird durchgeführt
- Chipdaten werden programmiert
- Chipdaten werden verglichen
- Konfigurationsdaten werden programmiert
- Prozess erfolgreich abgeschlossen!

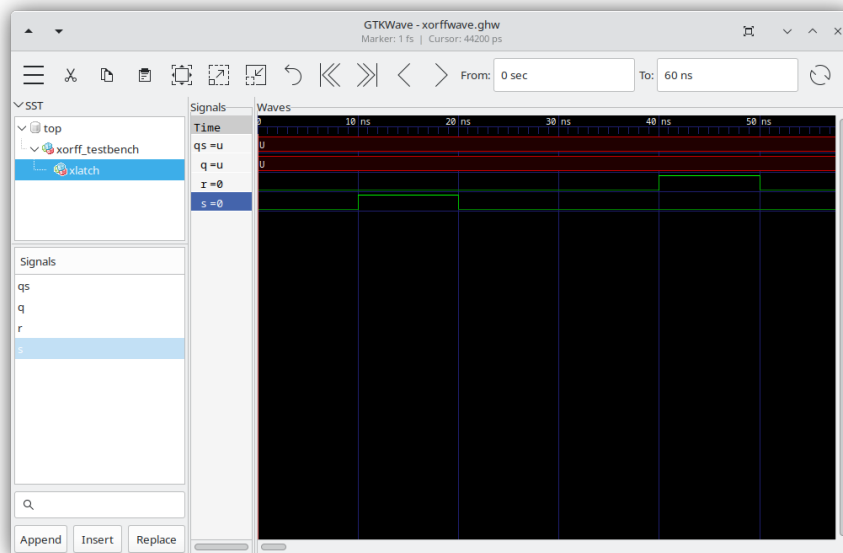


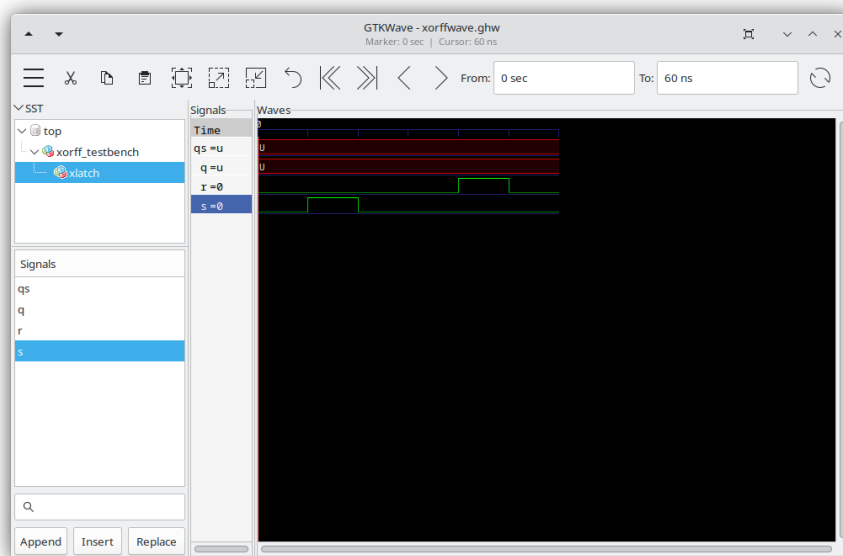
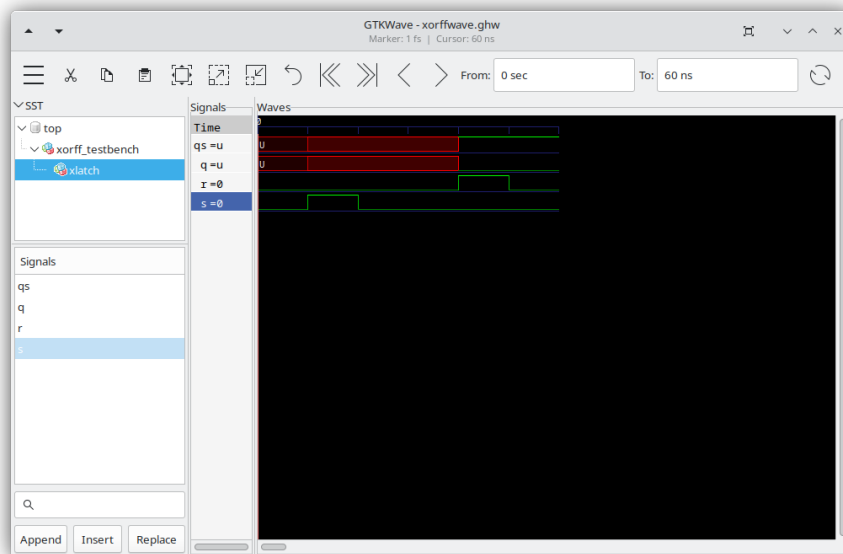
PDIP

(RESET) PC6	1	28	PC5 (ADC5/SCL)
(RXD) PD0	2	27	PC4 (ADC4/SDA)
(TXD) PD1	3	26	PC3 (ADC3)
(INT0) PD2	4	25	PC2 (ADC2)
(INT1) PD3	5	24	PC1 (ADC1)
(XCK/T0) PD4	6	23	PC0 (ADC0)
VCC	7	22	GND
GND	8	21	AREF
(XTAL1/TOSC1) PB6	9	20	AVCC
(XTAL2/TOSC2) PB7	10	19	PB5 (SCK)
(T1) PD5	11	18	PB4 (MISO)
(AIN0) PD6	12	17	PB3 (MOSI/OC2)
(AIN1) PD7	13	16	PB2 (SS/OC1B)
(ICP1) PB0	14	15	PB1 (OC1A)



	A	B	C	D	E	F	G
1	r	qs	q	s	qs (pruefung ueber q und s)	Qs (aus dem vorherigen)	
2	FALSCH	FALSCH	FALSCH	FALSCH			
3	FALSCH	WAHR	WAHR	WAHR			
4	FALSCH	FALSCH	FALSCH	FALSCH			
5	FALSCH	WAHR	WAHR	WAHR			
6	WAHR	FALSCH	WAHR	FALSCH			
7	WAHR	WAHR	FALSCH	WAHR			
8	WAHR	FALSCH	WAHR	FALSCH			
9	WAHR	WAHR	FALSCH	WAHR			
10							
11							





Jetzt muß man versuchen beim Gatter einen Eingang auf 1 zu legen den anderen auf den Taster. Das ist dann unserer nächster Versuch:

Material-Liste

- 1 Taster
- 1 Widerstand 1Kohm
- 2(1) Widerstände 10Kohm
- 1 LED
- 1 4011 (1 Gatter)

PROG-EXPRESS

Projekt

Buffer Hex-Editor

Datei Bearbeiten Ansicht Programmiergerät Checksumme: 52F6

AM27C256 - dump010620261.hex dump010620262.hex * Neu X

Adresse	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Windows-1252
00000000	06	D0	7F	77	00	CF	53	80	5F	D5	CE	C7	0A	37	66	00	w. IS
00000010	95	48	A7	DC	38	35	C9	D3	85	0A	5E	A2	2E	8E	50	00	HSSU;SE
00000020	6C	25	A8	80	09	4C	6C	15	F3	86	E2	F0	90	E2	A2	30	1%w;LL
00000030	91	85	49	6E	17	13	73	02	90	D7	23	64	09	8E	B2	CF	'uIn;..s
00000040	CF	46	9C	03	89	02	81	78	25	76	20	26	72	DC	9C	06	ISe;+0s(
00000050	74	80	35	AC	91	68	46	48	03	BA	4D	4D	45	AA	F1	28	t.5~+hFH
00000060	80	89	52	0C	C6	C7	90	73	82	97	96	28	76	07	DE	EA	..R;Aç;S
00000070	F5	AC	0F	40	99	0C	6A	A5	37	9C	2F	D5	3D	0D	78	94	0~;BES+
00000080	F2	3D	60	43	39	9F	F8	11	08	09	98	98	CE	ED	93	01	0~;C90.
00000090	51	02	51	CF	8A	0C	5F	7E	8D	04	D2	92	BC	AC	2E	26	Q.QIS;..
000000A0	92	15	46	7E	61	21	56	F5	09	C7	6D	1A	18	6A	AC	28	..F-alV0
000000B0	AF	01	05	A0	A6	D4	68	36	18	3E	D9	62	89	19	79	96	..(0n6
000000C0	AC	3A	09	20	DA	93	CE	DC	7D	51	50	F8	B0	17	A0	C0	..;U"U
000000D0	9C	6A	21	E1	88	F1	DC	C8	C6	7C	11	DE	08	84	B8	06	aj;iaen0E
000000E0	F8	E9	03	DE	A5	69	5C	C8	E8	00	98	5F	2E	42	40	9E	0a;B4Vei
000000F0	A9	E2	E7	7E	D6	CF	87	63	94	CE	3F	A1	8F	AE	F5	38	6a;QISc
00000100	18	BE	AC	41	AD	24	D0	8F	68	86	B3	50	E6	A6	19	20	.Z-A SYd
00000110	83	E4	D8	F7	8C	B2	6E	79	23	F4	B6	54	B9	39	D1	F7	fao;E=ny
00000120	18	76	E3	F6	53	20	AA	0B	54	81	D3	8E	51	E5	1A	58	..v80S~U
00000130	67	F3	D6	AC	6A	67	7C	5F	04	B8	F5	77	47	F1	BA	g0d~igl	
00000140	87	CE	C2	C2	C3	AD	58	0F	B5	27	DE	C9	6C	28	12	36	..IAAA X
00000150	0E	38	F9	D9	20	47	82	4E	29	37	8F	B6	57	C6	F5	83	.80b G'n
00000160	89	57	08	93	38	CF	9A	07	01	E2	D0	6D	7C	1D	1E	NA	..;IS
00000170	D8	D0	FF	04	45	47	08	94	30	96	D2	F8	06	B8	3A	06	0Dy;E5;'
00000180	0E	E7	61	DA	CA	D4	68	35	86	7D	83	9A	D4	18	75	27	..caUE0n5
00000190	5A	DE	20	7F	A7	61	E8	52	79	0F	AD	40	83	E2	84	F2	Zp~;SaeR
000001A0	19	58	2E	22	AD	32	1D	7E	DC	91	95	85	40	37	CE	45	..;2..i
000001B0	AC	68	59	F9	39	23	D6	FE	9D	2A	F6	61	4F	86	DA	1A	..kYu900b

Aktuelle Auswahl: HEX 0 bis HEX 0

Datei Bearbeiten Ansicht Lesezeichen Module Einstellungen Hilfe

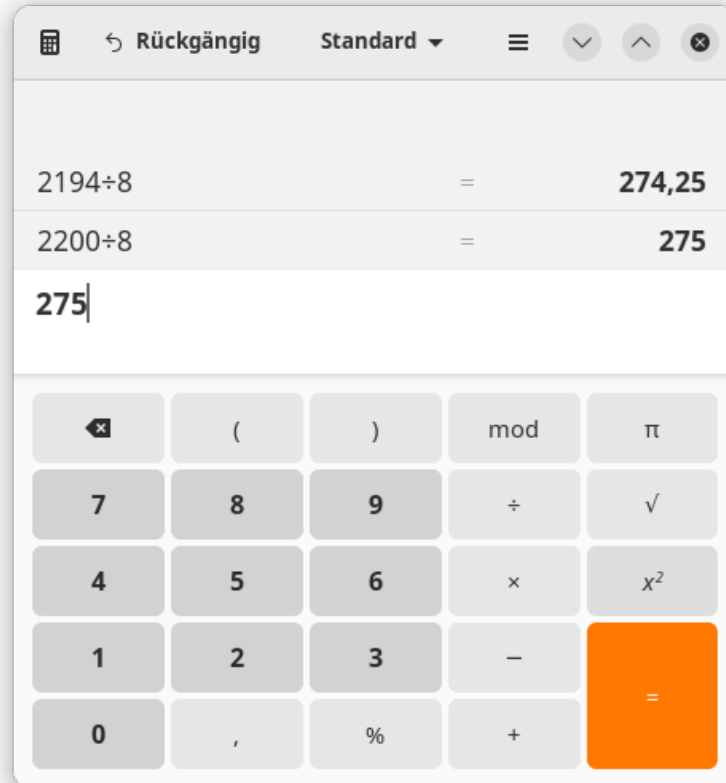
Neues Unterfenster Ansicht teilen Kopieren Einfügen Suchen

```
david@www001:~$ dd if=/dev/random of=/dev/gal16v8.01232026.hex --count=2194
dd: Ungültige Option --c
„dd --help“ liefert weitere Informationen.
david@www001:~$ dd if=/dev/random of=/dev/gal16v8.01232026.hex --count=2194
dd: Unbekannte Option »--count=2194«
„dd --help“ liefert weitere Informationen.
david@www001:~$ dd if=/dev/random of=/dev/gal16v8.01232026.hex count=2194
dd: konnte '/dev/gal16v8.01232026.hex' nicht öffnen: Keine Berechtigung
david@www001:~$ dd if=/dev/random of=gall6v8.01232026.hex count=1 bs=2194
1+0 Datensätze ein
1+0 Datensätze aus
2194 Bytes (2,2 kB, 2,1 KiB) kopiert, 0,000514788 s, 4,3 MB/s
david@www001:~$
```

Die Daten (2.194 Byte / 17.552 Bit) passen nicht komplett in den Chip (275 Byte / 2.200 Bit). Über den Chip Bereich hinaus gehende Daten werden nicht gebrannt und nicht verglichen.

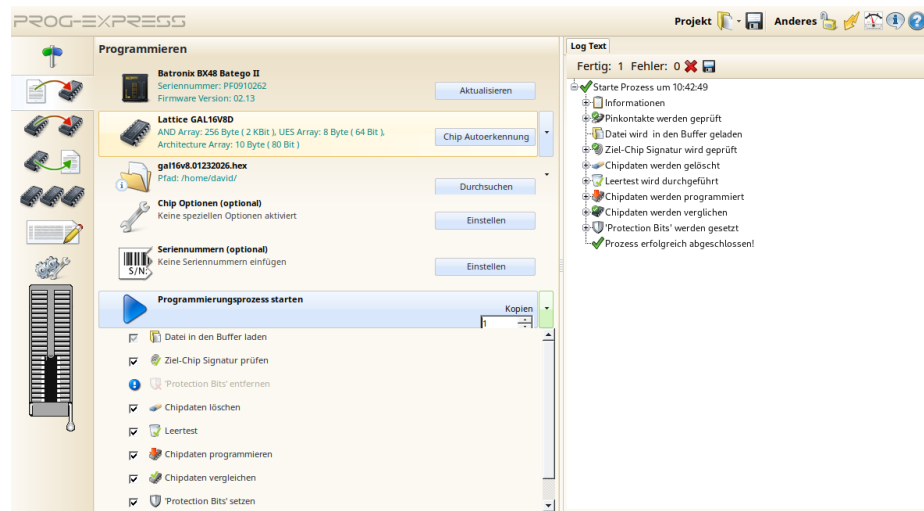
☐ Diesen Dialog nicht mehr anzeigen.

OK Abbrechen



```
David Datei Bearbeiten Ansicht Lesezeichen Module Einstellungen Hilfe
Neues Unterfenster Ansicht teilen Kopieren Einfügen Suchen

david@www001:~$ dd if=/dev/random of=/dev/gal16v8.01232026.hex --count=2194
dd: Ungültige Option --c
„dd --help“ liefert weitere Informationen.
david@www001:~$ dd if=/dev/random of=/dev/gal16v8.01232026.hex --count=2194
dd: Unbekannte Option »--count=2194«
„dd --help“ liefert weitere Informationen.
david@www001:~$ dd if=/dev/random of=/dev/gal16v8.01232026.hex count=2194
dd: konnte '/dev/gal16v8.01232026.hex' nicht öffnen: Keine Berechtigung
david@www001:~$ dd if=/dev/random of=gal16v8.01232026.hex count=1 bs=2194
1+0 Datensätze ein
1+0 Datensätze aus
2194 Bytes (2,2 kB, 2,1 KiB) kopiert, 0,000514788 s, 4,3 MB/s
david@www001:~$ dd if=/dev/random of=gal16v8.01232026.hex count=1 bs=275
1+0 Datensätze ein
1+0 Datensätze aus
275 Bytes kopiert, 0,00064134 s, 429 kB/s
david@www001:~$
```



```

21.01.26                                     testquine01212026.c                               1

#include <stdio.h>

// (c) David Vajda
// Quine Text
// 01/21/2026

#define and &&
#define or ||
#define not !
#define x3 (int)((((i & 0x0f) & 0x08) >> 3) & 0x01)
#define x2 (int)((((i & 0x0f) & 0x04) >> 2) & 0x01)
#define x1 (int)((((i & 0x0f) & 0x02) >> 1) & 0x01)
#define x0 (int)((((i & 0x0f) & 0x01) >> 0) & 0x01)

int main (void) {

    int i = 0;

    printf ("# (c) David Vajda\n");
    printf ("# Quine Text\n");
    printf ("# 01/21/2026\n\n");

    for (i = 0; i < 16; i++)
        printf ("%2i %i %i %i %i\t%i\n", i, x3, x2, x1, x0, (x3 and x2 and not x0) or \
            (not x3 and not x1 and x0) or \
            (x2 and not x1 and not x0) or \
            (not x2 and x1 and x0));

    return 0;
}

```

file:///home/david/testquine01212026.c

Truth Table

David Vajda

11. Januar 2026

1 Wahrheitstafeln und Gatter

1.1 AND $\wedge$

1.1.1 Schaltalgebra

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

1.1.2 Boolesche Algebra

x_1	x_0	$y = x_1 \wedge x_0$
f	f	f
f	w	f
w	f	f
w	f	w

1.2 OR $\vee$

1.2.1 Schaltalgebra

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

1.2.2 Boolesche Algebra

x_1	x_0	$y = x_1 \vee x_0$
f	f	f
f	w	w
w	f	w
w	f	w

1.3 NOT $\neg$

1.3.1 Schaltalgebra

A	B
0	1
1	0

1.3.2 Boolesche Algebra

x_0	$y = \neg x_0$
f	w
w	f

Sheet1

r	qs	q	s	qs (pruefung ueber q und s)
FALSCH	FALSCH	FALSCH	FALSCH	FALSCH
FALSCH	WAHR	WAHR	WAHR	FALSCH
FALSCH	FALSCH	FALSCH	FALSCH	FALSCH
FALSCH	WAHR	WAHR	WAHR	FALSCH
WAHR	FALSCH	WAHR	FALSCH	WAHR
WAHR	WAHR	FALSCH	WAHR	WAHR
WAHR	FALSCH	WAHR	FALSCH	WAHR
WAHR	WAHR	FALSCH	WAHR	WAHR

ok, drueber nachdenken
weil nachfolgezustand

daneben: jetzt streichen... wc

r	qs	q	s	qs (pruefung ueber q und s)
FALSCH	FALSCH	FALSCH	FALSCH	FALSCH
FALSCH	WAHR	WAHR	WAHR	FALSCH
FALSCH	FALSCH	FALSCH	FALSCH	FALSCH
FALSCH	WAHR	WAHR	WAHR	FALSCH
WAHR	FALSCH	WAHR	FALSCH	WAHR
WAHR	WAHR	FALSCH	WAHR	WAHR
WAHR	FALSCH	WAHR	FALSCH	WAHR
WAHR	WAHR	FALSCH	WAHR	WAHR

das waere eine moeglichkeit... aber lassen wir das... erst hingucken, vor allem ueberpru

5 dump010620261.hex

```
:20000000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF00
:20002000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFE0
:20004000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFC0
:20006000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFA0
:20008000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF80
```

```

:2000A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF60
:2000C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF40
:2000E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF20
:20010000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
:20012000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDF
:20014000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBF
:20016000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF9F
:20018000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF7F
:2001A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5F
:2001C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3F
:2001E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1F
:20020000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFE
:20022000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDE
:20024000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBE
:20026000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF9E
:20028000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF7E
:2002A000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF5E
:2002C000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF3E
:2002E000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF1E
:20030000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFD
:20032000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDD
:20034000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFBD
...
:207F8000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF01
:207FA000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFE1
:207FC000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFC1
:207FE000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFA1
:00000001FF

```

6 dump01062026.hex

```

:2000000023202863292064617669642076616A64610A232030312F30362F323032350A2333
:2000200020637265617465206865782064756D700A3135330A3130330A3132320A31353968
:200040000A33340A3131390A3133390A3136360A35390A35320A3135350A3232340A313998
:20006000310A3133330A3230330A3235320A3233350A3135320A3131370A3137340A323071
:20008000360A3136360A3233340A38310A3137360A3137320A34360A3136320A360A313659
..
:207F6000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF21
:207F8000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF01
:207FA000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFE1
:207FC000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFC1
:207FE000FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFA1
:00000001FF

```

7 gal01262026gen.c

```
#include <stdio.h>
```

```

int main (void) {
    char fuses[] = {0x88, 0x84, 0x48, 0x44};
    char ch1, ch2;

    int i;
    i = 0;
    while (!feof(stdin)) {
        ch2 = fgetc (stdin);
        ch1 = fgetc (stdin);
        if ((ch1 == '1') && (ch2 == '1'))
            printf ("%c", fuses [0]);
        else if ((ch1 == '1') && (ch2 == '0'))
            printf ("%c", fuses [1]);
        else if ((ch1 == '0') && (ch2 == '1'))
            printf ("%c", fuses [2]);
        else if ((ch1 == '0') && (ch2 == '0'))
            printf ("%c", fuses [3]);
        else
            break;
        fflush(stdout);
        ch2 = fgetc (stdin);
        ch1 = fgetc (stdin);
        fflush(stdin);
        if ((ch1 == '1') && (ch2 == '1'))
            printf ("%c", fuses [0]);
        else if ((ch1 == '1') && (ch2 == '0'))
            printf ("%c", fuses [1]);
        else if ((ch1 == '0') && (ch2 == '1'))
            printf ("%c", fuses [2]);
        else if ((ch1 == '0') && (ch2 == '0'))
            printf ("%c", fuses [3]);
        else
            break;
        fflush(stdin);
        fflush(stdout);
        printf ("%c%c", 0x00, 0x00);
        fflush(stdout);
    }
    printf ("%c%c%c%c", 0x00, 0x00, 0x00, 0x00);
    return 0;
}

```

8 gal16v8skript01222026.txt

(c) david vajda with lattice ... and ...
start: 01/22/2026
gal 16v8 and co excerpt

OLMC: Output Logic Macrocell

OLMC:
O: Output
L: Logic
MC: Macrocell

6 AND
1 OR

https://www.elektronik-labor.de/Lernpakete/Digi/dig_schalt1.htm

=1

=1 XOR
=1: XOR

IEC 60617-12: deutsch ...
US ANSI 91-1984: Amerikanisch
DIN 40700 (vor 1976)

[https://de.wikipedia.org/wiki/Liste_der_Schaltzeichen_\(Elektrik/Elektronik\)](https://de.wikipedia.org/wiki/Liste_der_Schaltzeichen_(Elektrik/Elektronik))

Kein: aequivalenz ...

7 AND &
1 OR >=1
1 EXOR =1

dann:

D-Latch:

in: D, CLK
out: Q, /Q

7 AND &
1 OR >=1
1 EXOR =1

D-Latch:
in: D, CLK
out: Q, /Q

3 MUX:
1.) mit 4 Bit Selector...

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

SYN, ACO, AC1(n)

Es gibt ... 8 AC1(n), bei 8 OLMC:
AC1(7), AC1(6), ..., AC1(1), AC1(0)

3 MUX:

north-medium
south-medium
medium-eastern

...

SYN
ACO
AC1(n)
AC1(m)

Vorgaenger: m

n: steht fuer die zelle
m: fuer die vorgaengerzelle

/OE: Output Enable - negative logic
VCC: Voltage Current Collector

SYN: globales Bitflag
ACO: globales Bitflag

...

es gibt 2 bit

SYN: globales Bitflag
ACO: globales Bitflag
SYN, ACO: globales Bitflag

AC1(n): zellenweises Bitflag
XOR(n): zellenweises Bitflag
AC1(n), XOR(n): zellenweises Bitflag

1.) globales bitflag
2.) zellenweises bitflag

9 galhexgen01262026.sh

```
#!/bin/bash

i=0
while [ $i -lt 8 ]
do
    cat dummy.txt | ./a.out >> galhexfuses01262026B.hex
    i=$((i+1))
done
```

10 genhexgalcode2.01262026.c

```
#include <stdio.h>

int main (void ) {
    char xor = 0xaa;
    char ues[9] = "dvpinbrd";
    char ac1 = 0xff;
    char or = 0xfe;
    char synac0 = 0xc0;
    printf ("%c%s%c%c%c", xor, ues, ac1, or, synac0);
    return 0;
}
```

11 genmamafernsehchannel011325.sh

```
#!/bin/bash

# (c) David Vajda
# 01/13/2025
# Mamas Fernsehsender DB - without mysql

Name=("Erstes" "ZDF" "Vox" "Sat1" "ZDF Info" "ZDF Neo" "MDR Sachsen" "WDR" "Arte" "NDR" "P
Nachrichten" "Tagesschau24" "HR" "SWR" "Alpha" "SR Fernsehen" "SWR/BW")
Number=(1 2 5 6 10 12 21 24 27 28 29 30 33 35 36 38 39 45 47 49 89)

echo "\\documentclass[a4paper]{article}"
echo "\\usepackage{german}"
echo "\\author{David und Barbara}"
echo "\\title{Fernsehsender Vodafone Kabel in der Judengasse 11}"
echo "\\begin{document}"
echo "\\maketitle"
echo "\\begin{Huge}"
echo "\\begin{tabular}{|c|c|}"
echo "\\hline"
echo "\\hline"
i=0
while [ $i -lt "${#Name[@]}" ]
```

```

do
    echo "${Number[$i]} & ${Name[$i]}\\\\"
    echo "\\hline"
    i=$((i+1))
done
echo "\\end{tabular}"
echo "\\end{Huge}"
echo "\\end{document}"

```

12 hello12122025std.py

```

# (c) david vajda
# 12/12/2025 1936A
# python 3 std uebung every day
# hello world!!!

```

```

print ("hello world")
print ("# (c) david vajda")
print ("# 12/12/2025 1936A")
print ("# python 3 std uebung every day")
print ("# hello world!!!")
## mit dem buch, author titel siehe unten
import time
print (time.localtime())

```

```

a = 3
b = 5
c = a + b

```

```

print (a, "+", b, "=", c)

```

```

divisor1 = 11
dividend1 = 151
quotient1 = dividend1 / divisor1
print (dividend1, "/", divisor1, "=", quotient1)
divisor2 = 151
dividend2 = 11
quotient2 = dividend2 / divisor2
print (dividend2, "/", divisor2, "=", quotient2)

```

```

# ganzzahldivision??? buch - thomas theis - einstieg in python - rheinwerk
# seite 26, ganzzahldivision, mit //

```

```

divisor3 = 11
dividend3 = 151
quotient3 = dividend3 // divisor3
rest3 = dividend3 % divisor3

```

```

print (dividend3, "//", divisor3, "=", quotient3, "rest", rest3)
divisor4 = 151
dividend4 = 11
quotient4 = dividend4 // divisor4
rest4 = dividend4 % divisor4
print (dividend4, "//", divisor4, "=", quotient4, "rest", rest4)

multiplikand3 = divisor3
multiplikator3 = quotient3
produkt3 = multiplikator3 * multiplikand3 + rest3

multiplikand4 = divisor4
multiplikator4 = quotient4
produkt4 = multiplikator4 * multiplikand4 + rest4

print (multiplikator3, "*", multiplikand3, "+", rest3, "=", produkt3)
print (multiplikator4, "*", multiplikand4, "+", rest4, "=", produkt4)

# zins erste rechnung, nach prozent
# G: Grundwert
# W: Prozentwert??
# p: prozentfuss
# p%: prozentwert

G = 500
W = 5
p = W/G

print (W, "/", G, "=", p*100, "%")

# zinsrechnung - und erste funktion

kapital = 2000.0
laufzeit = 12
zinssatz = 0.02

def rendite (kapital, zinssatz, laufzeit):
    if laufzeit > 0:
        renditeaktuell = rendite (kapital, zinssatz, laufzeit - 1)
        renditeaktuell = renditeaktuell + renditeaktuell * zinssatz
    else:
        renditeaktuell = kapital
    return renditeaktuell

print (rendite (kapital, zinssatz, laufzeit))

# jetzt koennen wir davon eine funktion zeichnen, indem wir das
# ganze noch mal machen mit array

kapital = 2000.0

```

```

laufzeit = 12
zinssatz = 0.02
v_renditedb = [2000.00]
v_laufzeitdb = [0]

def f_renditedb (kapital, zinssatz, laufzeit, v_renditedb, v_laufzeitdb):
    if laufzeit > 0:
        renditeaktuell = f_renditedb (kapital, zinssatz, laufzeit - 1, v_renditedb, v_laufzeitdb)
        renditeaktuell = renditeaktuell + renditeaktuell * zinssatz
        v_renditedb = v_renditedb.append (renditeaktuell)
        v_laufzeitdb = v_laufzeitdb.append (laufzeit)
    else:
        renditeaktuell = kapital
    return renditeaktuell

print (f_renditedb (kapital, zinssatz, laufzeit, v_renditedb, v_laufzeitdb))
print (v_renditedb)
print (v_laufzeitdb)

# nun der plot

import matplotlib.pyplot as plt
plt.plot (v_laufzeitdb, v_renditedb, label="zuwachsrate??/rendite, bei 2% auf 2000.00 GE,")
# plt.show ()

kapital = 2000.0
laufzeit = 12
zinssatz = 0.02
v_renditedb = [2000.00]
v_laufzeitdb = [0]

print (f_renditedb (kapital, zinssatz, laufzeit, v_renditedb, v_laufzeitdb))
print (v_renditedb)
print (v_laufzeitdb)

# nun der plot

plt.plot (v_laufzeitdb, v_renditedb, label="zuwachsrate??/rendite, bei 50% auf 2000.00 GE,")

kapital = 2000.0
laufzeit = 12
zinssatz = 0.5
v_renditedb = [2000.00]
v_laufzeitdb = [0]

print (f_renditedb (kapital, zinssatz, laufzeit, v_renditedb, v_laufzeitdb))
print (v_renditedb)
print (v_laufzeitdb)

# nun der plot

```

```

plt.plot (v_laufzeitdb, v_renditedb, label="zuwachsrate??/rendite, bei 50% auf 2000.00 GE,

# plt.show ()

# aus dem buch
valmin = 1
valmax = 16

import random
random.seed ()

M1 = []
M2 = []

def genCol (n):
    i = 0
    m_ = []
    while i < n:
        m_.append (random.randint(valmin, valmax))
        i = i + 1
    return m_

# test
print ("# test")
print (genCol(4))

def genM (n, m):
    i = 0
    M = []
    while i < m:
        M.append (genCol(n))
        i = i + 1
    return M

print (genM(4,4))

def printM (M):
    print ()
    i = 0
    while i < len(M):
        print (M[i])
        i = i + 1

printM(genM(4,4))

def mulM (M1, M2):
    i = 0

```

```

j = 0
M3 = []

while i < len(M1):
    j = 0
    m_ = []
    while j < len (M2[i]):
        l = 0
        v = 0
        while l < len (M1[i]):
            v = M1 [i] [l] * M2[l] [j] + v
            l = l + 1
        j = j + 1
        m_.append (v)
    M3.append (m_)
    i = i + 1
return (M3)

M1 = genM (4,4)
M2 = genM (4,4)
printM(M1)
printM(M2)
printM(mulM (M1,M2))

```

13 m8counterledblinkslow01212026.asm

```

;; (c) David Vajda
;; atmega 8, counter led + dnf - 4 Bit word
;; 01/21/2026

.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, 0x00
out PORTB, r16

l1:
out PORTB, r16
rcall c1
dec r16
rjmp l1

c1:
push r16

```



```

push r17
push r18
ldi r16, 0xff
12:
ldi r17, 0xff
13:
ldi r18, 0x04
14:
dec r18
brne 14
dec r17
brne 13
dec r16
brne 12

pop r18
pop r17
pop r16
ret

```

14 m8counterledblinkslow01212026.eep.hex

```
:00000001FF
```

15 m8counterledblinkslow01212026.hex

```

:0200000020000FC
:1000000004E00EBF0FE50DBF0FEF07BB00E008BB1C
:1000100008BB02D00A95FCCF0F931F932F930FEFCD
:100020001FEF24E02A95F1F71A95D9F70A95C1F741
:080030002F911F910F9108951B
:00000001FF

```

16 m8counteroscifast01212026.asm

```

;; (c) David Vajda
;; atmega 8, osci - natural fast - dnf - 4 Bit word - ttl or cmos 74/54/84xx
;; 01/21/2026

.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, 0x00
out PORTB, r16
```

```
l1:
com r16
out PORTB, r16
com r16
; rcall c1
dec r16
rjmp l1
```

```
c1:
push r16
push r17
push r18
ldi r16, 0xff
l2:
ldi r17, 0xff
l3:
ldi r18, 0x04
l4:
dec r18
brne l4
dec r17
brne l3
dec r16
brne l2
```

```
pop r18
pop r17
pop r16
ret
```

17 m8counteroscifast01212026.eep.hex

```
:00000001FF
```

18 m8counteroscifast01212026.hex

```
:0200000020000FC
:1000000004E00EBF0FE50DBF0FEF07BB00E008BB1C
:10001000009508BB00950A95FBCF0F931F932F9374
:100020000FEF1FEF24E02A95F1F71A95D9F70A95FB
:0A003000C1F72F911F910F91089561
:00000001FF
```

19 quine01102026.txt

(C) David Vajda

Sat Jan 10 11:51:31 2026

4 Network - TTL - Disjunktive Normalform

0	0	0	0	0	0
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	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	0
14	1	1	1	0	1
15	1	1	1	1	0

1	0	0	0	1	1
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
11	1	0	1	1	1
12	1	1	0	0	1
14	1	1	1	0	1

Gruppe 1:

1	0	0	0	1	1
4	0	1	0	0	1

Gruppe 2:

3	0	0	1	1	1
5	0	1	0	1	1
12	1	1	0	0	1

Gruppe 3:

11	1	0	1	1	1
14	1	1	1	0	1

1:3	0	0	-	1
1:5	0	-	0	1
4:5	0	1	0	-
4:12	-	1	0	0
3:11	-	0	1	1
12:14	1	1	-	0

4:5	0	1	0	-
1:3	0	0	-	1
12:14	1	1	-	0
1:5	0	-	0	1
4:12	-	1	0	0
3:11	-	0	1	1

Gruppe 1:

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

Gruppe 1:

1:3	0	0	-	1
-----	---	---	---	---

Gruppe 2:

12:14	1	1	-	0
-------	---	---	---	---

Gruppe 1:

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

Gruppe 1:

4:12	-	1	0	0
------	---	---	---	---

Gruppe 2:

3:11	-	0	1	1
------	---	---	---	---

Primimplikantentafel:

	1	3	4	5	11	12	14
4:5			+	+			
1:3	+	+					
12:14						+	+
1:5	+			+			
4:12			+			+	
3:11		+			+		

	1	3	4	5	11	12	14	
12:14						+	+	p
1:5	+			+				
4:12			+			+		
3:11		+			+			p

OK, verkuerzt:

12:14	1	1	-	0
1:5	0	-	0	1
4:12	-	1	0	0
3:11	-	0	1	1

```

        y    <=  (x3 and x2 and not x0) or
                  (not x3 and not x1 and x0) or
                  (x2 and not x1 and not x0) or
                  (not x2 and x1 and x0);

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026 is
port (
    x3, x2, x1, x0: in std_logic;
    y: out std_logic
);
end;

architecture behaviour of quine01102026 is
begin
    y    <=  (x3 and x2 and not x0) or
              (not x3 and not x1 and x0) or
              (x2 and not x1 and not x0) or
              (not x2 and x1 and x0);
end;

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026testbench is
port (
    y: inout std_logic
);
end;

architecture behaviour of quine01102026testbench iss
    component quine01102026
    port (
        x3, x2, x1, x0: in std_logic;
        y: out std_logic;
    );
    end component;
    signal x3, x2, x1, x0: std_logic;
begin
    q: quine01102026 PORT MAP  (x3=>x3, x2=>x2, x1=>x1, x0=>x0, y=>y);

```

20 quine01102026.vhdl

```

-- (C) David Vajda
-- Sat Jan 10 11:51:31 2026
-- 4 Network - TTL - Disjunktive Normalform

```

```

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026 is
port (
    x3, x2, x1, x0: in std_logic;
    y: inout std_logic
);
end;

architecture behaviour of quine01102026 is
begin
    y <= (x3 and x2 and not x0) or
        (not x3 and not x1 and x0) or
        (x2 and not x1 and not x0) or
        (not x2 and x1 and x0);
end;

library ieee;
use ieee.std_logic_1164.all;

entity quine01102026testbench is
port (
    y: inout std_logic
);
end;

architecture behaviour of quine01102026testbench is
    component quine01102026
    port (
        x3, x2, x1, x0: in std_logic;
        y: inout std_logic
    );
    end component;
    signal x3, x2, x1, x0: std_logic;
begin
    q: quine01102026 PORT MAP (x3=>x3, x2=>x2, x1=>x1, x0=>x0, y=>y);
    x0 <= '0' after 0 ns, '1' after 10 ns, '0' after 20 ns, '1' after 30 ns, '0' after 40
        ns;
    x1 <= '0' after 0 ns, '0' after 10 ns, '1' after 20 ns, '1' after 30 ns, '0' after 40
        ns;
    x2 <= '0' after 0 ns, '0' after 10 ns, '0' after 20 ns, '0' after 30 ns, '1' after 40
        ns;
    x3 <= '0' after 0 ns, '0' after 10 ns, '0' after 20 ns, '0' after 30 ns, '0' after 40
        ns;
end;

```

21 quinetest01212026.txt

```
# (c) David Vajda  
# Quine Text  
# 01/21/2026
```

```
0 0 0 0 0 0  
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 0  
7 0 1 1 1 0  
8 1 0 0 0 0  
9 1 0 0 1 0  
10 1 0 1 0 0  
11 1 0 1 1 1  
12 1 1 0 0 1  
13 1 1 0 1 0  
14 1 1 1 0 1  
15 1 1 1 1 0
```

22 quinetest2.01212026.txt

```
# (c) David Vajda  
# Quine Text  
# 01/21/2026
```

```
0 0 0 0 0 0  
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 0  
7 0 1 1 1 0  
8 1 0 0 0 0  
9 1 0 0 1 0  
10 1 0 1 0 0  
11 1 0 1 1 1  
12 1 1 0 0 1  
13 1 1 0 1 0  
14 1 1 1 0 1  
15 1 1 1 1 0
```

```
9 1 0 0 1 0
```

das ist zeile 9 und wo ist die
nach minimierung und min
restueberdeckung gelandet?

	1	3	4	5	11	12	14
4:5			+	+			
1:3	+	+					
12:14						+	+
1:5	+			+			
4:12			+			+	
3:11		+			+		

also, nirgendwo, war ja falsch...

ok ... logisch ...

1.) untersuchung: ist nirgendwo: weil:
zeile 9 haette 0 sein sollen, jetzt gucken wir uns an,
wie die eingangsvariablen stehen, woher das kommen kann:

1 0 0 1

na und nun? alle kontrollieren?

es gaebe ein einfacheres, mittel: wir stellen
die wahrheitstabelle erneut dar, mit geaenderter
zeile, dann machen wir quine und restueberdeckung
neu und vergleichen die ausdruecke, das was dann anders ist
hat dazu gefuehrt...

0	0	0	0	0	0
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	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	0
14	1	1	1	0	1
15	1	1	1	1	0


```

1 0 0 0 1 1
3 0 0 1 1 1
4 0 1 0 0 1
5 0 1 0 1 1
9 1 0 0 1 1
11 1 0 1 1 1
12 1 1 0 0 1
14 1 1 1 0 1

```

Gruppe 1:

```

1 0 0 0 1 1
4 0 1 0 0 1

```

Gruppe 2:

```

3 0 0 1 1 1
5 0 1 0 1 1
9 1 0 0 1 1
12 1 1 0 0 1

```

Gruppe 3:

```

11 1 0 1 1 1
14 1 1 1 0 1

```

ok:

```

1:3      0  0  -  1
1:5      0  -  0  1
1:9      -  0  0  1
4:5      0  1  0  -
4:12     -  1  0  0
3:11     -  0  1  1
9:11     1  0  -  1
12:14    1  1  -  0

```

```

1:9      -  0  0  1
4:12     -  1  0  0
3:11     -  0  1  1
1:5      0  -  0  1
1:3      0  0  -  1
9:11     1  0  -  1
12:14    1  1  -  0
4:5      0  1  0  -

```

1:9	-	0	0	1
4:12	-	1	0	0
3:11	-	0	1	1

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

1:3	0	0	-	1
9:11	1	0	-	1
12:14	1	1	-	0

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

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

3:11	-	0	1	1
------	---	---	---	---

1:9:3:11	-	0	-	1
----------	---	---	---	---

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

1:3	0	0	-	1
9:11	1	0	-	1
12:14	1	1	-	0

1:3:9:11		-	0	-	1
----------	--	---	---	---	---

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

4:12		-	1	0	0
1:5		0	-	0	1
12:14		1	1	-	0
1:3:9:11		-	0	-	1
4:5		0	1	0	-

	1	3	4	5	9	11	12	14
4:12			+				+	
1:5	+			+				
12:14							+	+
1:3:9:11	+	+			+	+		
4:5			+	+				

	1	3	4	5	9	11	12	14
12:14							+	+
1:3:9:11	+	+			+	+		
4:5			+	+				

12:14	1	1	-	0
1:3:9:11	-	0	-	1
4:5	0	1	0	-

```

y   <=  (x3 and x2 and not x0) or
        (not x2 and x0) or
        (not x3 and x2 and not x1);

```

ok: und das war die urspruengliche:

```

y   <=  (x3 and x2 and not x0) or
        (not x3 and not x1 and x0) or
        (x2 and not x1 and not x0) or
        (not x2 and x1 and x0);

```

wo sind die anders:

der erste min o. max term??? stimmtmt ...

der rest stimmt nicht

23 rand12152025py.py

```

# (c) David Vajda
# 12/15/2025
# sed - random - real dice ...
Y = [2, 3, 1, 3, 6, 3, 4, 2, 2, 6, 5, 1, 3, 1, 4, 4, 1, 4, 6, 1, 5, 1, 3, 5, 1, 5, 1, 4, 3]
X = []

i = 1
while i <= len (Y):
    X.append (i)
    i = i + 1

```

```

import matplotlib.pyplot as plt
import math

plt.plot (X, Y)
plt.show ()

S = [1, 2, 3, 4, 5, 6]
H = [0, 0, 0, 0, 0, 0]
relh = []

i = 0
while i < len (Y):
    H[Y [i] -1] = H[Y [i] -1]+1
    i = i + 1

i = 0
while i < 6:
    relh.append(H [i] / len (Y))
    i = i + 1

i = 0
while i < 6:
    print ("absolute haeufigkeit: ", i+1, ": ", H[i])
    i = i + 1

i = 0
while i < 6:
    print ("absolute haeufigkeit: ", i+1, ": ", H[i], "relative: ", relh[i])
    i = i + 1

E = 0
F = 0
i = 1
while i <= 6:
    E = (i*1/6) + E
    F = (i*relh[i-1]) + F
    i = i + 1

print ("Erwartungswert: laplace, ", E, "erwartungswert erfuehlt: ", F)

```

24 rand12152025py.txt

```

(c) David Vajda
12/15/2025
sed - random - real dice ...
2, 3, 1, 3, 6, 3, 4, 2, 2, 6, 5, 1, 3, 1, 4, 4, 1, 4, 6, 1, 5, 1, 3, 5, 1, 5, 1, 4, 3, 5,

```

25 rand12152025raw.txt

```
2,3, 1, 3, 6, 3, 4, 2, 2, 6, 5, 1, 3, 1,
4,414,61513,51514,3,5,61117131
1, 1,5,3, 1,52,4, 1,5,5, 2,5,5, 11
515171717,1,3313,3,21333
86510.313151514,64.1,512,213,7.3
1, 2, 3, 3, 6, 1, 1, 1,4, 614, 4,5, 4,5, 6, 2, x, 4,4, 5,5,6,5, 2,4,5, 313, 4, 2, 1, 2,
6, 1,6,8,51,3,2,5, 6,5,3,5,216, 3,5, 3, 1, 3, b, 5 4, 8, 1,3, 2, 1,6,1, 3,5, 4,
6,615, 513, 1, 6,016,5, 1,5,51 4, 3, 2, 4, 4,
4+2315161514133111611414141312,51
614,6111416131212,61311111213,514111
41415141615,61211, 1,61 113,2,5,2111
```

26 testquine01212026.c

```
#include <stdio.h>

// (c) David Vajda
// Quine Text
// 01/21/2026

#define and &&
#define or ||
#define not !
#define x3 (int)((((i & 0x0f) & 0x08) >> 3) & 0x01)
#define x2 (int)((((i & 0x0f) & 0x04) >> 2) & 0x01)
#define x1 (int)((((i & 0x0f) & 0x02) >> 1) & 0x01)
#define x0 (int)((((i & 0x0f) & 0x01) >> 0) & 0x01)

int main (void) {

    int i = 0;

    printf ("# (c) David Vajda\n");
    printf ("# Quine Text\n");
    printf ("# 01/21/2026\n\n");

    for (i = 0; i < 16; i++)
        printf ("%2i %i %i %i %i %i\t%i\n", i, x3, x2, x1, x0, (x3 and x2 and not x0) or \
            (not x3 and not x1 and x0) or \
            (x2 and not x1 and not x0) or \
            (not x2 and x1 and x0));

    return 0;
}
```

```
}
```

27 testquine2.01212026.c

```
#include <stdio.h>

// (c) David Vajda
// Quine Text
// 01/21/2026

#define and &&
#define or ||
#define not !
#define x3 (int)((((i & 0x0f) & 0x08) >> 3) & 0x01)
#define x2 (int)((((i & 0x0f) & 0x04) >> 2) & 0x01)
#define x1 (int)((((i & 0x0f) & 0x02) >> 1) & 0x01)
#define x0 (int)((((i & 0x0f) & 0x01) >> 0) & 0x01)

int main (void) {

    int i = 0;

    printf ("# (c) David Vajda\n");
    printf ("# Quine Text\n");
    printf ("# 01/21/2026\n\n");

    for (i = 0; i < 16; i++)
        printf ("%2i %i %i %i %i %i\t%i\n", i, x3, x2, x1, x0, (x3 and x2 and not x0) or
            (not x2 and x0) or
            (not x3 and x2 and not x1));

    return 0;
}
```

28 tosed4gal01262026a.txt

```
1 0 0 0 1    1
3 0 0 1 1    1
4 0 1 0 0    1
5 0 1 0 1    1
11 1 0 1 1   1
12 1 1 0 0   1
14 1 1 1 0   1
```

29 tosed4gal01262026.txt

```
0 0 0 0 0    0
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    0
7 0 1 1 1    0
8 1 0 0 0    0
9 1 0 0 1    0
10 1 0 1 0   0
11 1 0 1 1   1
12 1 1 0 0   1
13 1 1 0 1   0
14 1 1 1 0   1
15 1 1 1 1   0
```

```
cat tosed4gal01262026.txt | sed 's/.*[0-9].*\([0-1]\ [0-1]\ [0-1]\ [0-1]\).* \1/g'
```

```
cat tosed4gal01262026.txt | sed 's/.*[0-9].*\([0-1]\ [0-1]\ [0-1]\ [0-1]\).*\1/g' | tr '\n'
```

30 xor.vhdl.01222026.vhdl

```
-- (c) David Vajda
-- 01/22/2026
-- XOR - Latch - VHDL - test

library ieee;
use ieee.std_logic_1164.all;

entity xorff is
port (
    s: inout std_logic;
    r: inout std_logic;
    q: inout std_logic;
    qs: inout std_logic
);
end;

architecture behaviour of xorff is
begin
    q <= r xor qs;
    qs <= s xor q;
end;

library ieee;
use ieee.std_logic_1164.all;
```

```

entity xorff_testbench is
port (
    q: inout std_logic;
    qs: inout std_logic
);
end;

architecture behaviour of xorff_testbench is
    component xorff
    port (
        s: inout std_logic;
        r: inout std_logic;
        q: inout std_logic;
        qs: inout std_logic
    );
    end component;
    signal r, s: std_logic;
begin
    xlatch: xorff PORT MAP (r=>r, s=>s, q=>q, qs=>qs);
    q <= '1' after 2 ns;
    qs <= '1' after 2 ns;
    r <= '0' after 0 ns, '0' after 10 ns, '0' after 20 ns, '0' after 30 ns, '1' after 40 ns;
    s <= '0' after 0 ns, '1' after 10 ns, '0' after 20 ns, '0' after 30 ns, '0' after 40 ns;
end;

```