

... GAL 16v8 - galasm - ... and more

david vajda

10. Juli 2026

1 destillation ...zum ersten mal in der neuen wohnung

Ich hab mir jetzt mal ein bisschen Spaß erlaubt als zu denken. Das erste ist ja dass dieser Destillationsapparat wohl eher von Natur auch für ätherische Öle geeignet ist also ich persönlich hatte schon ein und den hab ich eigentlich noch immer. Ich hab jetzt dafür noch jede Menge Abzweigung. Die sind wichtig und die Kühler wieder die brauch ich auch dringend wieder vier Kühler und dann halt Abzweigungen aber man hat auch ne Kugel und das ist hier ja ein sehr dickes Gefäß unten muss man sagen und das ist das eine aber ich glaub das ist sehr gut zum Wasser destillieren gemacht ist erstens mal passt platt auf die Herdplatte, wenn man das jetzt anguckt

Ist es ja ein ziemlich großer Behälter und der hat wieder einen großen Behälter. Jetzt könnte man ja schlussfolgern letztenendes dass das gar nicht so losgeht. Jetzt hab ich an kochendes Wasser gedacht und ich hab ihr den Wasserkocher und auch dass sie so langsam kocht. Es steht nicht 100 prozent auf der Herdplatte fällt mir auf mir fällt auf man sollte das Stativ ins mittlere Loch machen, damit es andere Glas wo es rein läuft quasi rein läuft und dann könnte man noch drüber nachdenken wenn man das so anguckt dann kann man ja einen großen Wasser Kochtopf sich vorstellen da da oben ne Kugel ist, droht sozusagen. Ich warne davor, wenn's zu viel Druck hat dann wird's wahrscheinlich sogar durch die Kugel das kann man wahrscheinlich sogar erschießen. Also nicht so gut das Ergebnis haben von der Gesamtsituation sag ich trotzdem wenn man man muss sowieso aufpassen da hier passt ja sozusagen des Thermometer rein. Das sollte nicht drüber gefüllt sein und ich mach jetzt das was man macht sich beim Wasser kochen auch immer macht wenn die Spaghetti mache dann tue ich auch immer das Wasser vor kochen. Jetzt kocht es hier schon aber ich fühl es einfach weiter an und ja also man sieht hier zum Beispiel auch im Thermometer mehr oder weniger die Luft in dem Behälter wo also der Wasserdampf auch drin ist. Also das gasförmig in den Behälter ist etwa bei $100^{\circ}C$ Celsius, ein bisschen größer aber ich muss sagen wenn man's genau anguckt dann könnte man den Behälter einfach mehr füllen auch schon mit gekochten Wasser und dann trotzdem aufpassen ist nicht über gekocht, dass es durch die Kugel geht. Die Kugel sieht aus wie Hindernis bin ich aber dran denke dass er einfach mehr drin ist. Dann könnte es auch wahrscheinlich sehr gut dazu dienen, dass es dann sehr gut zum destillieren geeignet ist von Wasser. Ich probier's jetzt mal aus

Gut wir ersparen uns das weitere und gehen gleich über zu

1. Hash
2. fletchers checksum/adler-32
3. iban und Rest (kleinere Reste)
4. md5
5. crc
6. Zufall
7. parity...

Und am pc: Python 3 Zufall Und mit Würfel rein...

Also ich mache jetzt folgendes

1. Mit python3 zufallszahlen Diagramm, was ich sonst mit Würfeln mache
Als zusätzliche Aufgaben

Schriftliches, dividieren bei Binärzahlen zum Befehl hin

Hash

Und andere prüfsummen

2. Zufallszahlen Generator
3. Der Versuch es dann auch auf dem Baustein zum Beispiel schriftliches, dividieren Umzusetzen. Das brauche ich für Zufallszahlen Generator. Um das als Baustein zu machen, CRC Baustein MD5 und so weiter langfristig
Jetzt erst mal auf dem Papier, jetzt gleich auch schriftliches dividieren plus den Rest Auch python3

```
GAL16V8
MODULE rslatch06272026
    CLOCK X1 X0 NC NC NC NC NC NC GND
    /OE Y Z NC NC NC NC NC NC VCC
    Y = CLOCK & /Z & /X0
    Z = CLOCK & /Y & X0
DESCRIPTION
this is rs latch
```

so geht es:

```
GAL16V8
MODULE rslatch06272026
    CLOCK X1 X0 NC NC NC NC NC NC GND
    /OE Y Z NC NC NC NC NC NC VCC
```



```

        Y = CLOCK & /Z & /X0
        Z = CLOCK & /Y & /X1
DESCRIPTION
this is rs latch

so geht es

GAL16V8
MODULE rslatch06272026
    CLOCK X1 X0 NC NC NC NC NC NC GND
    /OE Y Z NC NC NC NC NC NC VCC
    Y = CLOCK & /Z & /X0
    Z = CLOCK & /Y & /X1
DESCRIPTION
this is rs latch

```

bash:

```

david@www001:~$ ./galasm rslatch07032026c.gal
GALasm 2.1, Portable GAL Assembler
Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved
Original sources Copyright (c) 1991-96 Christian Habermann
Assembler Phase 1 for "rslatch07032026c.gal"
Error in line 7: unknown pinname
Assembling failed.
david@www001:~$ ./galasm rslatch07032026c.gal
GALasm 2.1, Portable GAL Assembler
Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved
Original sources Copyright (c) 1991-96 Christian Habermann
Assembler Phase 1 for "rslatch07032026c.gal"
Assembler Phase 2 for "rslatch07032026c.gal"
GAL16V8; Operation mode: complex; Security fuse off
Error in line 8: mode 2: pins 12, 19 can't be used as input
Assembling failed.
david@www001:~$

```

versuch:

```

GAL16V8
MODULE rslatch06272026
    /OE R1 S1 R2 S2 NC NC NC NC GND
    CLOCK QR1 QS1 QR2 QS2 NC NC NC NC VCC
    QR1 = /S1 & CLOCK & /QS1
    QS1 = /R1 & CLOCK & /QR1
    QR2 = /S2 & CLOCK & /QS2
    QS2 = /R2 & CLOCK & /QR2
DESCRIPTION
this is rs latch

```

output, bash

```

david@www001:~$ ./galasm dlatch07032026.gal
GALasm 2.1, Portable GAL Assembler

```

```

Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved
Original sources Copyright (c) 1991-96 Christian Habermann
Assembler Phase 1 for "dlatch07032026.gal"
Error in line 7: unknown pinname
Assembling failed.
david@www001:~$ ./galasm dlatch07032026.gal
GALasm 2.1, Portable GAL Assembler
Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved
Original sources Copyright (c) 1991-96 Christian Habermann
Assembler Phase 1 for "dlatch07032026.gal"
Assembler Phase 2 for "dlatch07032026.gal"
GAL16V8; Operation mode: complex; Security fuse off
Error in line 8: mode 2: pins 12, 19 can't be used as input
Assembling failed.
david@www001:~$

```

galasm code

```

GAL16V8
MODULE rslatch06272026
    /OE D1 D2 NC NC NC NC NC NC GND
    CLK QR1 QS1 QR2 QS2 NC NC NC NC VCC
    QR1 = D1 & CLK & /QS1
    QS1 = /D1 & CLK & /QR1
    QR2 = D2 & CLK & /QS2
    QS2 = /D2 & CLK & /QR2
DESCRIPTION
this is rs latch

```

code:

```

GAL16V8
MODULE rslatch06272026
    /OE D1 D2 NC NC NC NC NC NC GND
    CLOCK QR1 QS1 QR2 QS2 NC NC NC NC VCC
    QR1 = D & CLK & /QS1
    QS1 = /D & CLK & /QR1
    QR2 = D & CLK & /QS2
    QS2 = /D & CLK & /QR2
DESCRIPTION
this is rs latch

```

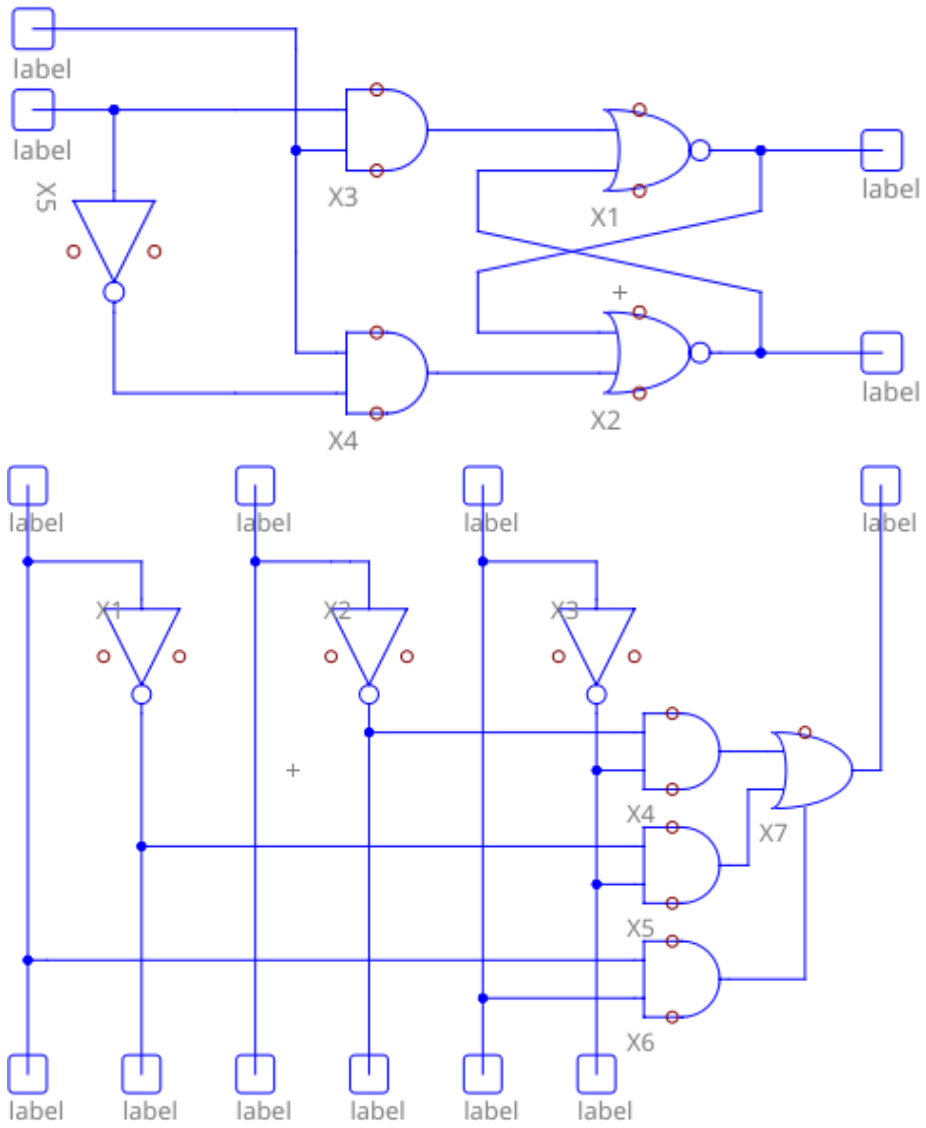
code

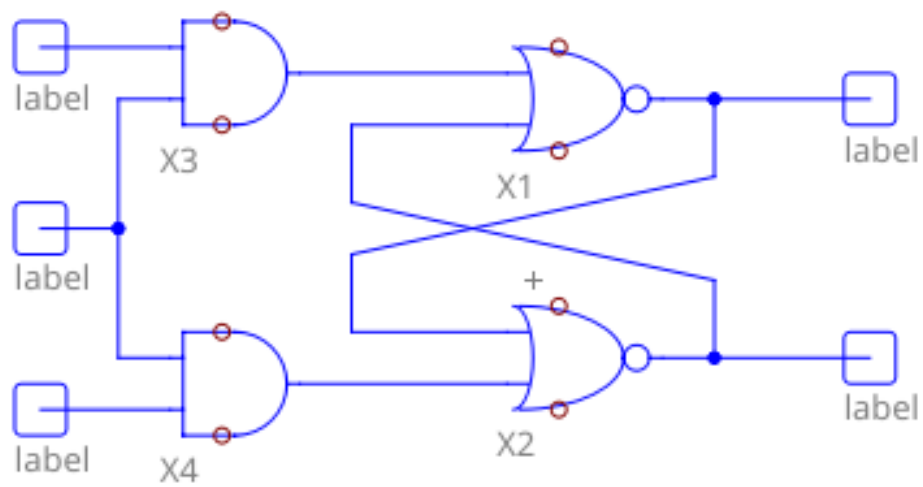
```

GAL16V8
MODULE rslatch06272026
    /OE R1 S1 R2 S2 NC NC NC NC GND
    CLOCK QR1 QS1 QR2 QS2 NC NC NC NC VCC
    QR1 = D & CLK & /QS1
    QS1 = /D & CLK & /QR1
    QR2 = D & CLK & /QS2
    QS2 = /D & CLK & /QR2

```

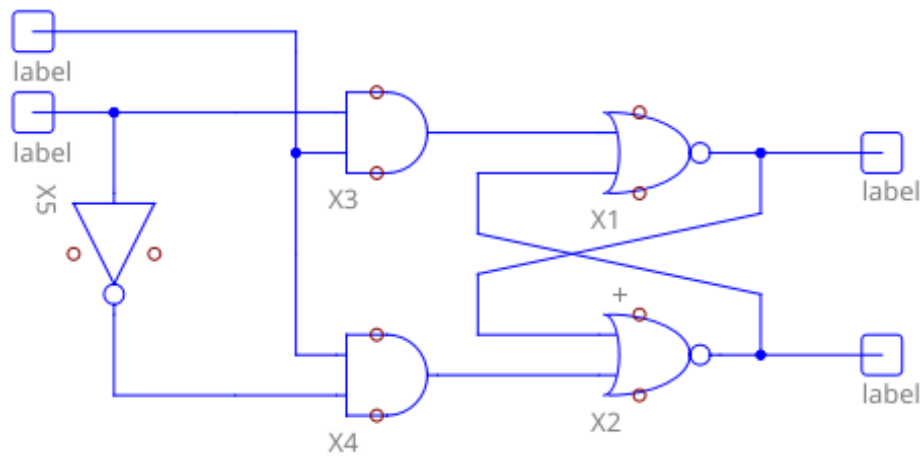
```
DESCRIPTION
this is rs latch
```

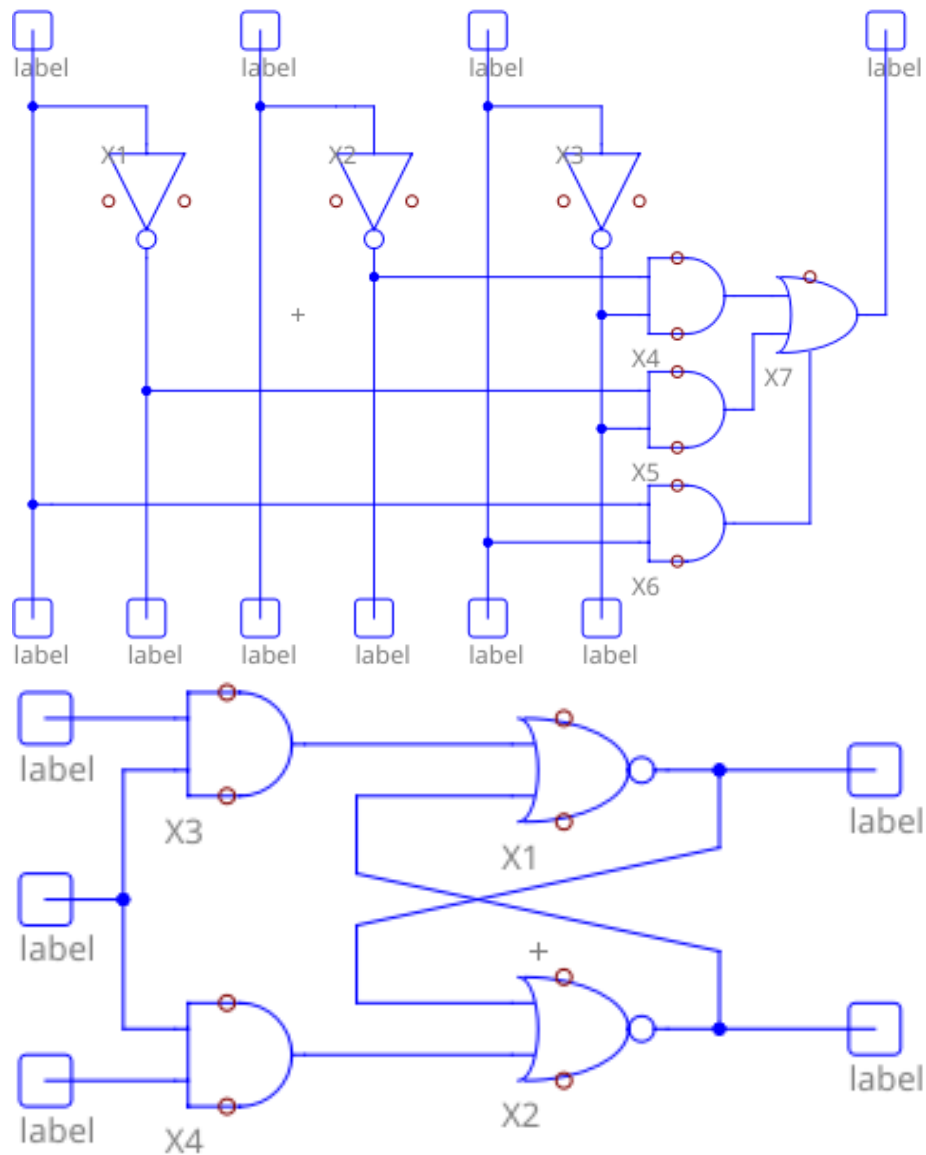




der naechste schritt, waere, aus dem rs, den gleichen eingang zu machen ein mal negiert und ein mal nicht... ohne klammern ergibt sich das:

$$\begin{aligned} QR1 &= /S1 \& CLK \& /QS1 \\ QS1 &= /R1 \& CLK \& /QR1 \end{aligned}$$





2 hier alle dateien

```
GAL16V8
MODULE rslatch06272026
    CLOCK X1 X0 NC NC NC NC NC NC GND
    /OE Y Z NC NC NC NC NC NC VCC
    Y = CLOCK & /Z & /X0
    Z = CLOCK & /Y & /X1
DESCRIPTION
this is rs latch
```

GAL16V8

-----___/-----			
CLOCK	1	20	VCC
X1	2	19	NC
X0	3	18	NC
NC	4	17	NC
NC	5	16	NC
NC	6	15	NC
NC	7	14	NC
NC	8	13	Z
NC	9	12	Y
GND	10	11	/OE

Pin 19 = NC XOR = 0 AC1 = 0

0 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

1 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

2 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

3 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

4 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

5 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

6 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

7 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

Pin 18 = NC XOR = 0 AC1 = 0

8 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

9 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

10 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

11 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

12 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

13 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

14 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

15 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

Pin 17 = NC XOR = 0 AC1 = 0

16 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

17 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

18 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

19 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

20 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

21 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

22 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

23 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

Pin 16 = NC XOR = 0 AC1 = 0

```

24  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
25  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
26  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
27  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
28  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
29  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
30  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
31  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 15 = NC          XOR = 0    AC1 = 0
32  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
33  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
34  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
35  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
36  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
37  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
38  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
39  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 14 = NC          XOR = 0    AC1 = 0
40  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
41  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
42  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
43  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
44  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
45  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
46  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
47  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 13 = Z           XOR = 1    AC1 = 0
48  -xx- ---- ---- ---- ---- ----x ----
49  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
50  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
51  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
52  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
53  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
54  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
55  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 12 = Y           XOR = 1    AC1 = 0
56  --x- -x-- ---- ---- ---- --x ----
57  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
58  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
59  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
60  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
61  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
62  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
63  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

```

GAL16V8

MODULE rslatch06272026

CLOCK X1 X0 NC NC NC NC NC NC GND

/OE Y Z NC NC NC NC NC NC VCC

Y = CLOCK & /Z & /X0

this is rs latch

4d42

Pin #	Name	Pin Type
1	CLOCK	Input
2	X1	Input
3	X0	Input
4	NC	Input
5	NC	Input
6	NC	Input
7	NC	Input
8	NC	Input
9	NC	Input
10	GND	GND
11	/OE	Input
12	Y	Output
13	Z	Output
14	NC	NC
15	NC	NC
16	NC	NC
17	NC	NC
18	NC	NC
19	NC	NC
20	VCC	VCC

GAL16V8			
CLOCK	1	20	VCC
X1	2	19	NC
X0	3	18	NC
NC	4	17	NC

NC		5	16 NC
NC		6	15 NC
NC		7	14 NC
NC		8	13 Z
NC		9	12 Y
GND		10	11 /OE

Pin 19 = NC XOR = 0 AC1 = 0

0 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

1 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

2 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

3 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

4 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

5 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

6 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

7 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

Pin 18 = NC XOR = 0 AC1 = 0

8 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

9 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

10 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

11 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

12 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

13 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

14 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

15 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

Pin 17 = NC XOR = 0 AC1 = 0

16 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

17 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

18 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

19 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

20 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

21 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

22 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

23 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

Pin 16 = NC XOR = 0 AC1 = 0

24 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

25 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

26 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

27 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

28 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

29 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

30 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

31 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

```

Pin 15 = NC          XOR = 0    AC1 = 0
32  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
33  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
34  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
35  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
36  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
37  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
38  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
39  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 14 = NC          XOR = 0    AC1 = 0
40  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
41  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
42  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
43  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
44  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
45  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
46  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
47  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 13 = Z           XOR = 1    AC1 = 0
48  --x-  x---  ----  ----  ----  ----  ---x  ----
49  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
50  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
51  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
52  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
53  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
54  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
55  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 12 = Y           XOR = 1    AC1 = 0
56  --x-  -x--  ----  ----  ----  ----  ---x  ----
57  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
58  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
59  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
60  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
61  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
62  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
63  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx

```

GAL16V8

MODULE rslatch06272026

CLOCK X1 X0 NC NC NC NC NC NC GND

/OE Y Z NC NC NC NC NC NC VCC

Y = CLOCK & /Z & /X0

Z = CLOCK & /Y & X0

DESCRIPTION

this is rs latch

Used Program: GALasm 2.1

GAL-Assembler: GALasm 2.1

Device: GAL16V8

*QF2194


```

        Y1 = CLOCK & /Z1 & /X1
        Z1 = CLOCK & /Y1 & X1
DESCRIPTION
this is rs latch

GAL16V8
MODULE quine06272026dnf
    CLOCK X2 X1 X0 NC NC NC NC NC GND
    /OE Y NC NC NC NC NC NC NC VCC
    Y = /X1 * /X0 + /X2 * /X0 + X2 * X0
DESCRIPTION
This is dnf

```

GAL16V8			
-----___/-----			
CLOCK	1	20	VCC
R	2	19	NC
S	3	18	NC
NC	4	17	NC
NC	5	16	NC
NC	6	15	NC
NC	7	14	NC
NC	8	13	Q2
NC	9	12	Q1
GND	10	11	/OE

```

Pin 19 = NC      XOR = 0    AC1 = 0
0  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
1  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
2  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
3  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
4  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
5  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
6  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
7  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 18 = NC      XOR = 0    AC1 = 0
8  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
9  xxxx xxxx xxxx xxxx xxxx xxxx xxxx
10 xxxx xxxx xxxx xxxx xxxx xxxx xxxx
11 xxxx xxxx xxxx xxxx xxxx xxxx xxxx
12 xxxx xxxx xxxx xxxx xxxx xxxx xxxx

```

```

13  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
14  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
15  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 17 = NC          XOR = 0    AC1 = 0
16  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
17  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
18  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
19  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
20  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
21  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
22  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
23  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 16 = NC          XOR = 0    AC1 = 0
24  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
25  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
26  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
27  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
28  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
29  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
30  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
31  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 15 = NC          XOR = 0    AC1 = 0
32  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
33  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
34  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
35  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
36  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
37  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
38  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
39  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 14 = NC          XOR = 0    AC1 = 0
40  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
41  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
42  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
43  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
44  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
45  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
46  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
47  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 13 = Q2          XOR = 1    AC1 = 0
48  ----  ----  ----  ----  ----  ----  --x-  ----
49  x---  ----  ----  ----  ----  ----  ----  ----
50  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
51  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
52  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
53  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
54  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
55  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 12 = Q1          XOR = 1    AC1 = 0
56  ----  ----  ----  ----  ----  ----  --x-  ----

```

```
GAL16V8
MODULE rslatch06272026
    CLOCK R S NC NC NC NC NC NC NC GND
    /OE QR QS NC NC NC NC NC NC NC VCC
    QS = QR $ S
    QR = QS $ R
```

```
Used Program:    GALasm 2.1
GAL-Assembler:  GALasm 2.1
Device:         GAL16V8
```

Pin #	Name	Pin Type
1	CLOCK	Input
2	R	Input
3	S	Input
4	NC	Input
5	NC	Input
6	NC	Input
7	NC	Input
8	NC	Input
9	NC	Input
10	GND	GND
11	/OE	Input
12	Q1	Output
13	Q2	Output
14	NC	NC
15	NC	NC

16		NC		NC
17		NC		NC
18		NC		NC
19		NC		NC
20		VCC		VCC

GAL16V8				
-----___/-----				
CLOCK		1	20	VCC
X1		2	19	NC
X0		3	18	NC
NC		4	17	NC
NC		5	16	NC
NC		6	15	NC
NC		7	14	NC
NC		8	13	Z
NC		9	12	Y
GND		10	11	/OE

Pin 19 = NC								XOR = 0		AC1 = 0	
0	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
1	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
2	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
3	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
4	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
5	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
6	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
7	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
Pin 18 = NC								XOR = 0		AC1 = 0	
8	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
9	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
10	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
11	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
12	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
13	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
14	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
15	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
Pin 17 = NC								XOR = 0		AC1 = 0	
16	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx
17	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx

```

18  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
19  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
20  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
21  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
22  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
23  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 16 = NC          XOR = 0    AC1 = 0
24  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
25  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
26  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
27  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
28  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
29  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
30  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
31  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 15 = NC          XOR = 0    AC1 = 0
32  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
33  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
34  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
35  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
36  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
37  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
38  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
39  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 14 = NC          XOR = 0    AC1 = 0
40  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
41  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
42  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
43  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
44  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
45  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
46  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
47  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 13 = Z           XOR = 1    AC1 = 0
48  -x--  ----  ----  ----  ----  ----  ---x  ----
49  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
50  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
51  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
52  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
53  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
54  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
55  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
Pin 12 = Y           XOR = 1    AC1 = 0
56  ----  -x--  ----  ----  ----  ----  ---x  ----
57  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
58  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
59  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
60  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
61  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx
62  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx  xxxx

```


GAL16V8

CLOCK X1 X0 NC NC NC NC NC NC GND

$$Y = /Z \ \& \ /X0$$

DESCRIPTION

GAL-Assembler: GALasm 2.1

*QF2194

```
*L1792 11111011111111111111111110111111111
```

```
*L2056 0100110101001111010001000101010101001100010001010010000001110010
```

[illegible]

*L2193 0

4d73

1	CLOCK	Input
---	-------	-------

2	X1	Input
---	----	-------

3	X0	Input
---	----	-------

4	NC	Input
---	----	-------

5	NC	Input
---	----	-------

6	NC	Input
---	----	-------

```

7      | NC          | Input

```

8	NC	Input
---	----	-------

9	NC	Input
---	----	-------

10 | GND | GND

```
11      | /OE      | Input
```

12	Y	Output
----	---	--------

13	Z	Output
----	---	--------

14 | NC | NC

15 | NC | NC

16 | NC | NC

17 | NC | NC

18 | NC | NC

19 | NC | NC

20 | VCC | VCC

/OE		1		20		VCC
R		2		19		NC
S		3		18		NC
NC		4		17		NC
NC		5		16		NC
NC		6		15		NC
NC		7		14		NC
NC		8		13		QS
NC		9		12		QR
GND		10		11		CLOCK

```

Pin 19 = NC          XOR = 0    AC1 = 0
0  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
1  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
2  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
3  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
4  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
5  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
6  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
7  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 18 = NC          XOR = 0    AC1 = 0
8  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
9  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
10 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
11 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
12 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
13 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
14 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
15 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 17 = NC          XOR = 0    AC1 = 0
16 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
17 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
18 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
19 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
20 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
21 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
22 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
23 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 16 = NC          XOR = 0    AC1 = 0
24 xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

```

```

25  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
26  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
27  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
28  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
29  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
30  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
31  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 15 = NC          XOR = 0    AC1 = 0
32  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
33  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
34  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
35  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
36  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
37  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
38  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
39  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 14 = NC          XOR = 0    AC1 = 0
40  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
41  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
42  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
43  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
44  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
45  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
46  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
47  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 13 = QS          XOR = 1    AC1 = 0
48  -x-- ---- ---- ---- ---- ----x ----
49  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
50  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
51  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
52  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
53  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
54  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
55  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
Pin 12 = QR          XOR = 1    AC1 = 0
56  ---- -x-- ---- ---- ---- ----x ----
57  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
58  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
59  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
60  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
61  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
62  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
63  xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

```

GAL16V8

MODULE rslatch06272026

/OE R S NC NC NC NC NC NC GND

CLOCK QR QS NC NC NC NC NC NC VCC

QR = /S & /QS

QS = /R & /QR

[illegible]

Pin #	Name	Pin Type
1	/OE	Input
2	R	Input
3	S	Input
4	NC	Input
5	NC	Input
6	NC	Input
7	NC	Input
8	NC	Input
9	NC	Input
10	GND	GND
11	CLOCK	Input
12	QR	Output
13	QS	Output
14	NC	NC
15	NC	NC
16	NC	NC
17	NC	NC
18	NC	NC
19	NC	NC
20	VCC	VCC

this is rs latch

dann muss da noch ein CLK mit UND rein, ergibt eine scheinbar komische formulierung, ich mache einen schaltplan

```
GAL16V8
MODULE rslatch06272026
    /OE R1 S1 R2 S2 NC NC NC NC GND
    CLOCK QR1 QS1 QR2 QS2 NC NC NC NC VCC
    QR1 = /S1 & CLK & /QS1
    QS1 = /R1 & CLK & /QR1
    QR2 = /S2 & CLK & /QS2
    QS2 = /R2 & CLK & /QR2
DESCRIPTION
this is rs latch
```

jetzt fehlen noch die und Gatter, zum takt

```
GAL16V8
MODULE rslatch06272026
    /OE R1 S1 R2 S2 NC NC NC NC GND
    CLOCK QR1 QS1 QR2 QS2 NC NC NC NC VCC
    QR1 = /S1 & /QS1
    QS1 = /R1 & /QR1
    QR2 = /S2 & /QS2
    QS2 = /R2 & /QR2
DESCRIPTION
this is rs latch
```

dann machen wir jetzt zwei FF

```
GAL16V8
MODULE rslatch06272026
    /OE R1 S1 NC NC NC NC NC NC GND
    CLOCK QR1 QS1 NC NC NC NC NC NC VCC
    QR1 = /S1 & /QS1
    QS1 = /R1 & /QR1
DESCRIPTION
this is rs latch
```

Flipflop jetzt umzusetzen mit zwei Flipflops und nach dem und davor ist es eine relativ einfache Gleichung. Wenn man die alle einzelnen hinschreiben kann. Das kann man hier wie gesagt nicht es gibt hier keine Signale. Dann müssen wir daraus jetzt einen Ausdruck Dafür reicht die Konzentration aufgrund des didaktischen was ich gesagt habe jetzt nicht mehr deswegen gucken wir jetzt die ganze Zeit des Flipflop an. Das kann auch ein bisschen Spaß machen hoffe ich mal <https://www.vajda-breadboard.de/html/video07102026/IMG4108.mov> <https://www.vajda-breadboard.de/html/video07102026/IMG4107.mov> <https://www.vajda-breadboard.de/html/video07102026/IMG4106.mov> So der Witz bei der Geschichte ist, dass bei einem Flipflop müssen naturgemäß außer es schaltet beide auf GND sein und das ist das Problem Das ist das Problem was passiert denn wenn man die Leitung in einem sozusagen im Eingang deswegen gibt es Pull down und Pull Up das müssen aber erst lernen zu unterscheiden ich auch von beim Eingang müssen wir noch unterscheiden von der Masse trennen. Generell ist

dann sagen wir mal der Pegel. Da bin ich jetzt grad in Schwierigkeiten noch mal das so sagt nicht mehr auf Masse und wenn der nicht mehr auf Masse ist, dann gilt das schon als VCC also als High Level, dann müssen Sie mir die Ausdrücke jetzt gerade verzeihen. Die sind jetzt sehr untechnisch und möglicherweise falsch oder sehr technisch und trotzdem nicht 100 Prozent am richtigen Ort reden wir trotzdem. Wir überreden tun wir aber gerade nicht. Um die Sache zu erklären muss ich jetzt sehr technisch reden also die GND Um die Sache jetzt zu erklären, muss ich technisch reden. Hier sind aber womöglich Ausdrucksschwierigkeiten da so wenn ich die von dem GND trenne. Und hier sieht man mal ein ein Latch, kein Flipflop in der Realität. Dann sieht man, dass sie sofort zu einem seltsamen Verhalten führt, wie wahrscheinlich schon viele vermutet haben, die sich das vielleicht auskennen, denn es zeigt ein typisches seltsames Verhalten, wenn ich die Eingänge von dem GND dann wie gesagt, zeigt das zeigt die Schaltung ein Verhalten und zeigt sonst eher so, dass man meint, es ist ja mal wieder nichts passiert so das täuscht drüber hinweg das ist kein Flipflop das ist ein ganz einfaches RS Latch und das hat seine eigene Art, wenn beide sozusagen auf GND sind, dann bleibt es so stehen und das müssen wir wissen. Diese zwei LEDs sind die Ausgänge. Es bleibt stehen und wenn es stehen bleibt, sind eben entweder die eine LED an oder die andere und nur, wenn wir sozusagen einen der Eingänge von der Masse trennen dann wechselt das Licht, wie wir hier sind, so beziehungsweise sagen wir einer der Ausgänge jetzt lassen wir das dabei und ich mache daraus sofort ein Flipflop

Das Flipflop tut, zeigt aber ein unerwartetes Verhalten was allerdings naturgemäß normal ist. Ich zeige einen Film. Ich habe jetzt das

gal16v8

Programmiert sitzt jetzt hier in der Schaltung ich hab auch die Kabel richtig dran gemacht aber ich muss noch mal kurz Pause machen. Die Spannungsversorgung ist schon da muss aber noch dran, damit ich diese Flipflop jetzt ausprobieren. Ich geh davon aus es funktioniert ist kein Flipflop ist ein Latch <https://www.vajda-breadboard.de/html/video07102026/IMG4108.mov> <https://www.vajda-breadboard.de/html/video07102026/IMG4107.mov> <https://www.vajda-breadboard.de/html/video07102026/>

GAL16V8

```
MODULE rslatch06272026
    /OE R S NC NC NC NC NC NC GND
    CLOCK QR QS NC NC NC NC NC NC VCC
    QR = /S & /QS
    QS = /R & /QR
```

DESCRIPTION

this is rs latch

bash input/output/console/tty

david@www001:~\$./galasm rslatch07032026.gal

GALasm 2.1, Portable GAL Assembler

Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved

Original sources Copyright (c) 1991-96 Christian Habermann

Assembler Phase 1 for "rslatch07032026.gal"

Assembler Phase 2 for "rslatch07032026.gal"

```

GAL16V8; Operation mode: simple; Security fuse off
Assembling successfully completed.
david@www001:~$ ./galasm rslatch07032026b.gal
GALasm 2.1, Portable GAL Assembler
Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved
Original sources Copyright (c) 1991-96 Christian Habermann
Assembler Phase 1 for "rslatch07032026b.gal"
Assembler Phase 2 for "rslatch07032026b.gal"
GAL16V8; Operation mode: simple; Security fuse off
Assembling successfully completed.
david@www001:~$

```

tatsaechlich, jetzt hat es funktioniert, es sind nicht die bezeichnungen, QS und QR

```

GAL16V8
MODULE rslatch06272026
    /OE R S NC NC NC NC NC NC GND
    CLOCK QR QS NC NC NC NC NC NC VCC
    QR = /S & /QS
    QS = /R & /QR
DESCRIPTION
this is rs latch

```

ich verstehe, wo das problem ist ... das problem ist wohl, das problem ist - dass CLK und /OE wohl vertauscht sind das eine ist ein eingang und das ein eingang. gut, weiss nicht, ausprobieren. oder variablen, so geht es ja.. aber das eine QR und QS geht nicht...so, geht ...auch ...

```

GAL16V8
MODULE rslatch06272026
    CLOCK X1 X0 NC NC NC NC NC NC GND
    /OE Y Z NC NC NC NC NC NC VCC
    Y = /Z & /X0
    Z = /Y & /X1
DESCRIPTION
this is rs latch

```

so: jetzt tut das hier nicht...sondern: es gibt gar kein NOR, das ist hier ein XOR tut so oder so nicht, also anderer ausdruck...

Facebook

programmiersprache abel dnf ausdrucks schaltnetz - Google Suche

Beispiel eines ABEL-Quellcodes (DNF)

Angenommen, Sie haben die folgende Wahrheitstabelle für einen Ausgang Y, der genau dann 1 ist, wenn die Eingangsvariablen A, B und C die Werte 001, 011 oder 111 annehmen:

- $Y = (\bar{A} \wedge \bar{B} \wedge C) \vee (\bar{A} \wedge B \wedge C) \vee (A \wedge B \wedge C)$

Der zugehörige ABEL-Code für ein einfaches Schaltnetz sieht so aus:

```

abel
MODULE schaltnetz_dnf

" Pins und Signalnamen definieren
Declarations
  A, B, C    pin 1, 2, 3    ISTYPE 'in';
  Y          pin 14        ISTYPE 'com';

Equations
  " Die Gleichung in DNF (ODER-Verknüpfung von UND-Termen)
  Y = (A & B & C) # (A & B & C) # (A & B & C);

END
        
```

Frage stellen

+

↓

5 Websites

– Informatik I – Modul 3: Schaltnetze - IFI

UZH

2012 Burkhard Stiller. M3 – 32. Disjunktive Normalform. Und damit fällt sich die disjunktive Normalform...

Department of Informatics UZH

Advanced Boolean Equation Language - Wikipedia

ABEL wurde in den 1980er Jahren von der „Data I/O Corporation“ in Redmond (Washington) entwickelt...

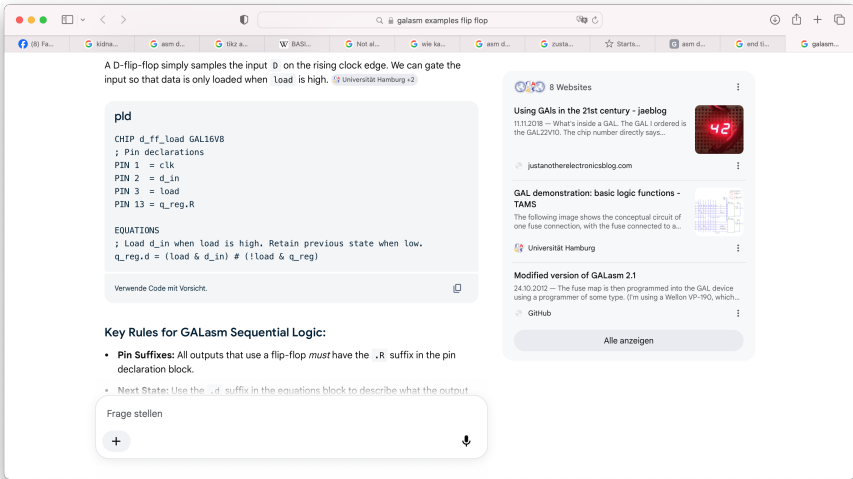
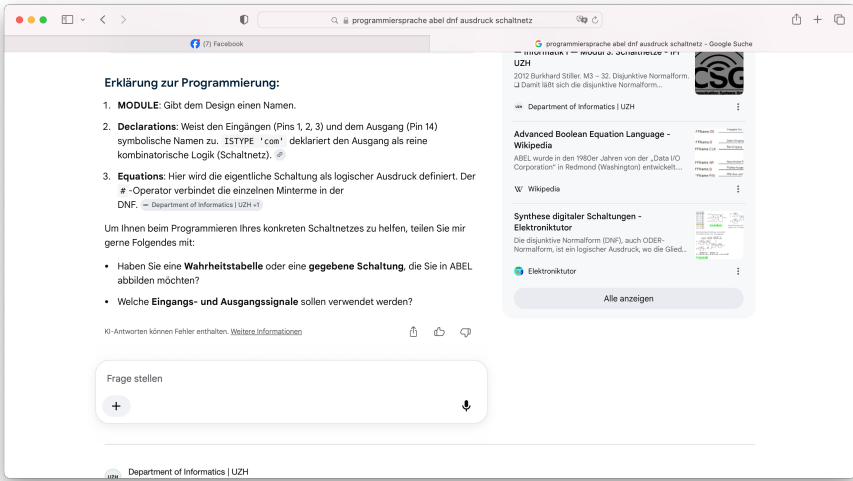
W Wikipedia

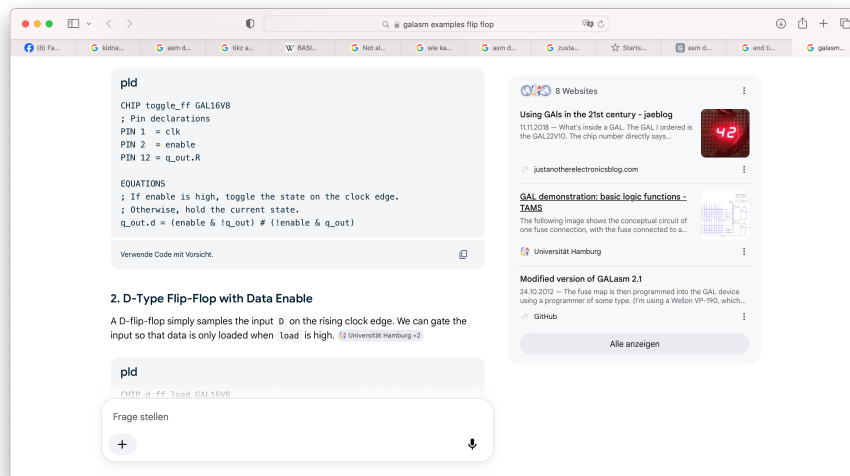
Synthese digitaler Schaltungen - Elektroniktor

Die disjunktive Normalform (DNF), auch ODER-Normalform, ist ein logischer Ausdruck, wo die Glied...

Elektroniktor

Alle anzeigen





- \$ kennt er nicht, also machen wir
- UND - was auch immer das nicht
- /Z UND(zu ersetzen) /X0
- das war tatsaechlich mit den variablenamen, und: QR und QS akzeptiert das nicht...so.

```
GAL16V8
MODULE rslatch06272026
    CLOCK X1 X0 NC NC NC NC NC NC GND
    /OE Y Z NC NC NC NC NC NC VCC
    Y = Z # X0
    Z = Y # X1
DESCRIPTION
this is rs latch
```

das macht er auch nicht, also das sind nicht variablen namen... bash input/output/console/ttl

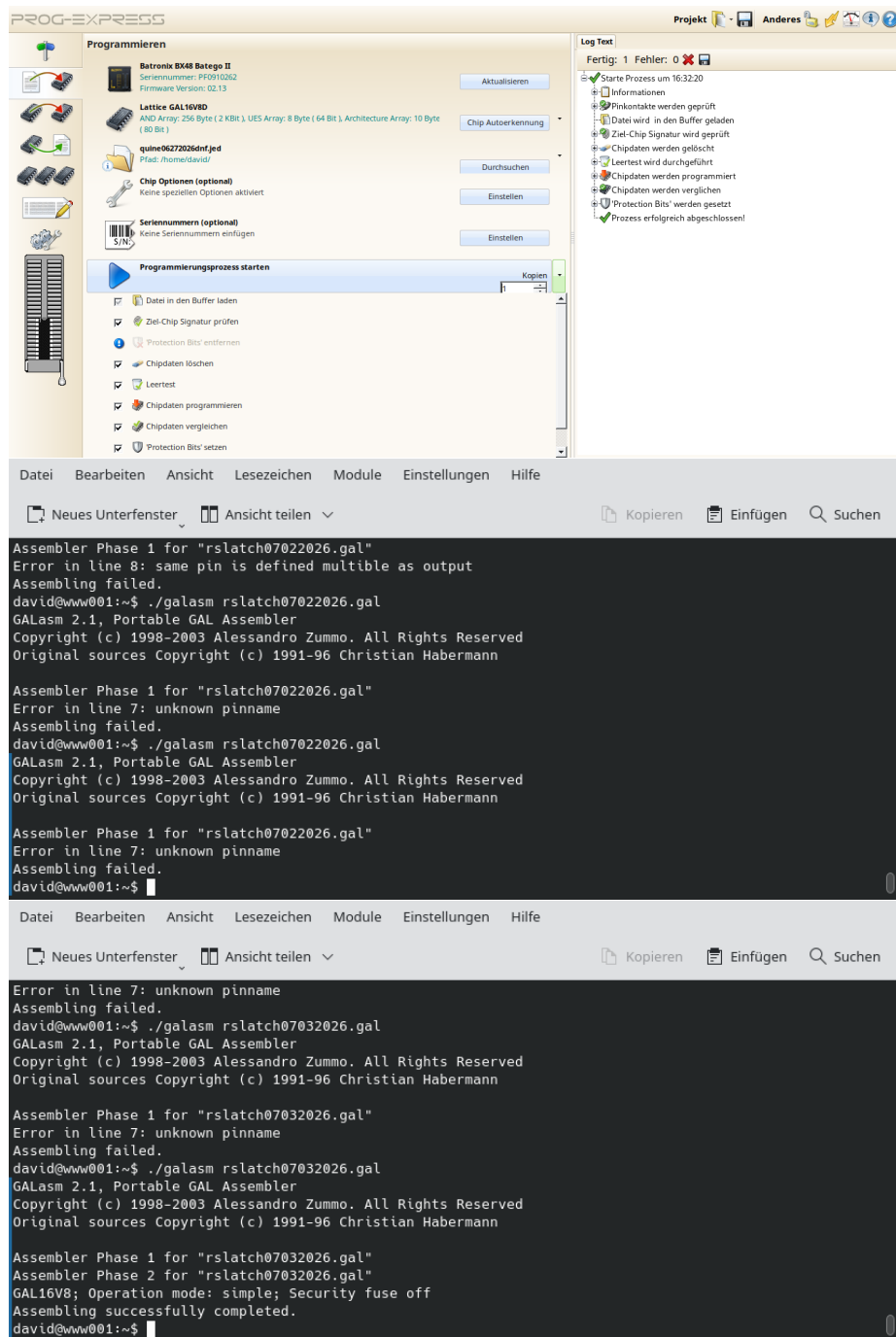
```
Assembler Phase 1 for "rslatch07022026.gal"
Error in line 7: unknown pinname
Assembling failed.
david@www001:~$ ./galasm rslatch07022026.gal
GALasm 2.1, Portable GAL Assembler
Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved
Original sources Copyright (c) 1991-96 Christian Habermann
Assembler Phase 1 for "rslatch07022026.gal"
Error in line 7: unknown pinname
Assembling failed.
david@www001:~$
```

jetzt macht immer noch dasselbe problem, unknown pin name, vielleicht kommt der mit QS dem namen nicht zurecht...

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

Used Program: GALasm 2.1
GAL-Assembler: GALasm 2.1
Device: GAL16V8

*F0
*G0
*QF2194
*L1792 11111011101111111111111111111111
*L1824 10111111101111111111111111111111
*L1856 01111111011111111111111111111111
*L2048 00000001
*L2056 01001101001111010001000101010101001100010001010010000001110001
*L2120 00000000
*L2128 1111111111111111111111111111111111111111111111111111111111111111111111
*L2192 1
*L2193 0
*C181b
*
54ff
david@www001:~$ pdftoppm
```



In den Logikgleichungen ist auf die Reihenfolge der Prioritäten zu achten. Folgende Operatoren werden im Allgemeinen verwendet:

Tabelle 2.2 a – Logische Operatoren

Verknüpfung	Standard	Alternativ	Priorität
NICHT	!	/	1 (höchste)
UND	&	*	2
ODER	#	+	3
XOR	\$	:+	3
XNOR	!\$	:*	3

Tabelle 2.2 b – Arithmetische Operatoren

		Priorität
Zweierkomplement	-A	1 (höchste)
Multiplikation	A*B	2

```

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

GAL16V8; Operation mode: simple; Security fuse off
Assembling successfully completed.
david@www001:~$ ./galasm rslatch07032026.gal
GALasm 2.1, Portable GAL Assembler
Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved
Original sources Copyright (c) 1991-96 Christian Habermann

Assembler Phase 1 for "rslatch07032026.gal"
Error in line 7: unknown pinname
Assembling failed.
david@www001:~$ ./galasm rslatch07032026.gal
GALasm 2.1, Portable GAL Assembler
Copyright (c) 1998-2003 Alessandro Zummo. All Rights Reserved
Original sources Copyright (c) 1991-96 Christian Habermann

Assembler Phase 1 for "rslatch07032026.gal"
Assembler Phase 2 for "rslatch07032026.gal"
GAL16V8; Operation mode: simple; Security fuse off
Assembling successfully completed.
david@www001:~$

```

Da hat es aber glaube ich auch mit einem Dollar geht wir probieren's ausprobieren wir mal aus, ob das jetzt die des Flipflop dann erfüllt ich habe das thema geklaert, gleich einen moment

$$y = /a * /b$$

aber auf papier...jetzt kommt das NOR als DNF...also, ich weiss jetzt was mit dem NOR nicht stimmt. sie brauchen in der booleschen algebra, garantiert keine klammern. sie koennen alles als KNF und DNF ausdruecken. wenn es alles heisst, dann auch das NOR. das heisst, fuer den GAL programmierer technisch

1. bei VHDL - was grosse PLD kommt, sprich: FPGA - und so weiter, kann man sich vorstellen, ein FPGA besteht ausser lauter PAL - sprich GAL ... das wurde in folge - technisch die situation der so zu sagen, umsetzung, wie bei der sprache von sprache nach backend - compiler - wurde vereinfachen, sie muessten die komponenten - otto verbraucher versteht nicht - jeweils auf ein GAL in dem gesamten Ding unterbringen.
2. also: sie haben bei VHDL - immer signal + halt pin, und das ist der unterschied. wenn sie components definieren, sprich - entity + architecture

+

entity ist der aeussere teil, also pins aussen und \verbarchitectureformel innen, dann haben sie: insgesamt die **component**

die **component** ist allerdings mehr oder weniger ein GAL der programmierte GAL.

so, wenn das so ist, dann muessen sie trotzdem **entity** und **architecture** programmieren, weil sie wissen ja nicht wenn das innen viele GAL - viele gibt es in der Informatik nicht, viele sind

nicht 2^n deswegen sagt man nicht viele, sondern blockweise. sie haben nicht 7. koennen sie haben

dann haben sie 8 moeglichkeiten und davon sind 7 benutzt. trotzdem man blockweise.

jedenfalls. sie haben entity und archicture also eine komponente, blos wo landen die, das die frage:

weil: sie haben ja, das so variable, den FPGA - wie der GAL die formel drauf muss. also, haben sie das bauteil im FPGA, so zu sagen ein GAL und das Blockweise. und wenn sie das haben, dann brauchen sie trotzdem

pins: und **signals**. und wenn sie das haben, dann ist das in der schaltung

technisch macht das sind, das ist keine turing maschine, die ist unerschoppflich

weil: sie haben am GAL 16v8, 20 pins, davon VCC und GND und dazu kommt noch: /OE. ok. und ein CLK, man kann nicht zu viele Variablen benutzen, sonst sind das ausgaenge. die muessen wir nicht fuer zwischenrechnung benutzen

so, deswegen brauchen wir auch signals. das sind kabel, sagen wir in der schaltung

das ist beim GAL fragwuerdig, somit beim FPGA - angenommen besteht aus vielen ...jetzt trotzdem: wenn wir eine variable haben, brauchen wir signal

und beim FPGA - eben, sind das einzelne GAL - so zu sagen, nur wissen nicht wo und muessen **entity** definieren und **architecture**

wenn allerdings alles als DNF, dann ohne klammern

ohne klammern: heisst, auch NOR

es gibt NOR: $\$$
und wenn es das gibt, heiter sonne schein

trotzdem kann man NOR auch als DNF ausdruecken, das ist jetzt eine zwischenuebung

zwischenuebung heisst:

wir nehmen die Tabelle von NOR und druecken als DNF aus, danach kommt mit dem richtigen NOR $\$$ NOR in galasm, die schaltung ich habe hier was

https://f-ei.hszg.de/fileadmin/user_upload/Redakteure/PRIVAT/sbischoff/Digitaltechnik/Vers

nor ist $\$$. und jetzt passen sie mal auf! wir brauchen keine klammern! weil wir sind in der booleschen algebra! und wir haben gelernt, alle - booleschen ausdruecke - sind ausdruecken als dnf und knf

dnf: M1 ODER M2 ODER M3
M: Maxterm
m: minterm

ich weiss, was wohin

knf: m1 UND m2 UND m3

wobei die negiert sind, als variablen irgendwie so...

und deswegen geht ohne klammern, geht in arithmetik auch, aber es gibt einen unterschied.

weil sie haben, nacher $x_1 * x_2$

das haben sie auch bei bool

$x_1 * x_2$ ist x_1 UND x_2

waehrend

$x_1 + x_2$ ist x_1 ODER x_2

das problem ist: warum das ungewoehnlich aussieht, weil sie mit mal hohe werte verbinden

$x_1 * x_2 * \dots * x_n = 123123127831283182$ irgendso was. bis 0

und trotzdem sie haben bei 0 und 1, schaltalgebra, wahr und falsch - boolsche aber die vermengen sich nicht mit dem ausdruck, weil die sich vollstaendig reduzieren +

psychologisches denken, das sie auf ihrer mathematik verdraengen, damit die mathematik ad absurdum fuehren, naemlich indem sie unerfuerchtig psychologie und mathematik nicht trennen.

so weiter! das funktioniert nicht...

```

GAL16V8
MODULE rslatch06272026
    CLOCK R S NC NC NC NC NC NC GND
    /OE QR QS NC NC NC NC NC NC VCC
    QS.T = QR # S
    QS = /QS.T
    QR.T = QS # R
    QR = /QR.T
DESCRIPTION
this is rs latch

```

folgendes:

Y3.T

vielleicht dieses .T das koennte eine loesung sein, es gibt nicht das wort signal, aber Y3.T Y3.S ...ach so, vielleicht kann man das so machen...man weiss QR + \verbS einer variablen zu. Y sagen wir und macht dann \verbQS /Y dann brauchen wir signale... weil, diese „variablen“entsprechen den ausgaengen, also wie in VHDL, halt variablen in **entity** ...in **architecture** selber, gibt es signale. signale sind leitungen, umgangssprachlich kabel... die sind halt in der schaltung, ob es das in galasm auch gibt??? also signals, in galasm, sind variablens, die pins aussen... kleine schaltung grosse schaltung...

```

Y1.T = B
Y2.T = C + D
Y3.T = /G
Y1.E = /A
Y3.E = E * F

```

so geht es aber der ausdruck passt nicht...

```

GAL16V8
MODULE rslatch06272026
    CLOCK R S NC NC NC NC NC NC GND
    /OE QR QS NC NC NC NC NC NC VCC
    QS = QR # S
    QR = QS # R
DESCRIPTION
this is rs latch

```

so geht es nicht

```

GAL16V8
MODULE rslatch06272026
    CLOCK R S NC NC NC NC NC NC GND
    /OE QR QS NC NC NC NC NC NC VCC
    QS = /(QR # S)
    QR = /(QS # R)
DESCRIPTION
this is rs latch

```


das problem bei dem galasm quelltext gestern sind nicht die Variablennamen sondern die klammern. galasm erlaubt keine klammern. bei ausdruecken. so die frage ist, wie macht man ein NOR? fuer FF? und latch? dafuer braucht man ein NOR als operator oder ein NICHT vom OR. also hier ist irgendein unknown pin name

```
GAL16V8
MODULE rslatch06272026
    CLOCK R S NC NC NC NC NC NC GND
    /OE QR QS NC NC NC NC NC NC VCC
    QS = /(QR # S)
    QR = /(QS # R)
DESCRIPTION
this is rs latch
```

man kann auch * statt # nehmen... galasm macht beides ist aber nicht abel. . . so, jetzt habe ich eine idee fuer uns alle. ich kann, nachdem, was ich hier an formel gefunden habe und an pins, kann ich hier - . . . - ein flip flop

1. RS-Latch - das ist ganz einfach, mit zwei NOR
2. RS-Latch dasselbe aber, zu einem Taktgesteuerten, mit noch zwei UND - davor, die zusammen einen CLK haben
3. daraus wird das D-Latch, indem man den einen daten eingang mit dem anderen verbindet
4. F-FF- zwei solcher dinger mit invertierten takt.

an einem GAL 16v8 . . . ok: und da die pins da sind

ich habe auch die Formel vom Volladdierer.

kann man auch noch machen, generell passt gal. . . auch eine kleine CPU, vielleicht mit extern 16 Byte speicher oder so, intern 4 x 4 Bit register satz, addition, subtraktion, einen sprung oder so, eine verzweigung, mal gucken. so jetzt fangen an. formel:

```
Q = (R NOR Qs)
Qs = (S NOR Q)
```

das kann man in galasm alternative fuer ABEL gut hinschreiben, als:

```
QT = (R NOR QS)
QS = (S NOR QT)
```

ok. und aber kein NOR das ist so wahrscheinlich nicht da, sondern ein

/(x # y) . . .

ok und die QS und QT muessen ausgaenge sein, R und S eingaege, eingaege oben, ausgaenge unten. . . so. im galasm text...

```

CLK X2 X1 X0 NC NC NC NC NC GND
CLK X2 X1 X0 NC NC NC NC NC GND
CLK ... GND
/OE ... VCC
CLK X2 X1 X0 NC .. GND
CLK X2 X1 X0 NC .. GND

```

usw. have a nice day!!! uebung zu clk NC NC 07/02 also, nach dem letzten: kann ich nur sagen, es gibt maximal 2 schlueselworte: MODULE auch noch, muss es aber nicht...

```

GAL16V8
MODULE quine06272026dnf
    CLOCK X2 X1 X0 NC NC NC NC NC GND
    /OE Y NC NC NC NC NC NC NC VCC
    Y = /X1 * /X0 + /X2 * /X0 + X2 * X0
DESCRIPTION
This is dnf

```

So, jetzt das übliche Zeug unsere DNF - mit gal16v8 und: ... wie immer. Vorher ein paar Übungen für die Pins...

so, jetzt grössere pause, ich lerne heute auch noch mal auswendig, z.B. vielleicht das galasm, und eben atmega8, lcd hd44780 und alles was zu cs i/ii gehoert...

```

(C) David Vajda
Thu Jul  2 12:29:17 2026
3 Network - TTL - Disjunktive Normalform
  x2 x1 x0 y
  0 0 0 0 1
  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 0

```

Aus dem kopf: GAL16v8 mit galasm
Irgendwie: der Baustein, am Anfang, aber was? Weiss ich nicht!
\begin{verbatim}
GAL16v8

Dann die pins Was kommt davor, irgendwie so

```

(C) David Vajda
Thu Jul  2 12:29:17 2026
3 Network - TTL - Disjunktive Normalform
Aus dem kopf: GAL16v8 mit galasm
Irgendwie: der Baustein, am Anfang, aber was? Weiss ich nicht!
GAL16v8
Dann die pins

```

```

Was kommt davor, irgendwie so
EN NC NC NC NC EN NC NC NC GND
EN NC NC NC NC EN NC NC NC VCC
;; es sind auf jeden fall 10 pins auf beiden Seiten?? Gal16v8
;; ich glaube clk ist bei galasm in der zweiten unteren reihe
;; ebenso GND im Norden, VCC im S"uden, hinten ist aber in der realitaet
;; typisches TTL Gatter - IC artiges DIP/DIL Baustein, wg. Z"ahlung... anders herum. Also
;; blos muss gucken, VCC/GND Nord/Sued und: CLK glaube ich unten, dann 2. pin, geht los
;; oben EN, oder so, nein, unten /OE. OE - Output Enable, CE, OE, WE - Chip Enable
;; output enable, write enable und /OE ist invertierte Logik, OE steht fuer Output enable
;; /OE, heisst, der Output ist enable, wenn OE LOW pegel hat...
;; ok, jetzt anfangen, das heisst, die Formel kommt, vor PINS, kommt PINS,
;; glaube ich, dann EQUATIONS, dann DESCRIPTION
;; bevor ich jetzt die Formel habe, kommt, alle 1er raus nehmen muss trotzdem:
EQUATION
y = (/x2 & /x1 & /x0) # (/x2 & /x1 & x0) # (/x2 & x1 & /x0) # (/x2 & x1 & x1) # (x2 & x1 &
DESCRIPTION
DNF on GAL16V8 nicht mit Quine mc cluskey oder - KV Diagramm minimiert
Gruppe 0:
Gruppe 1:
Gruppe 2:
Ok, sortieren:
0:1:2:3
(/x2) # (x1 & /x0)

```

ich kann das jetzt so nicht sagen, ich muss auch galasm erst auswendig lernen, z.B. Pins und equations, wie immer auswendig lernen, so wie sonst, so kann ich mir das erst mal nicht merken, hier stehen Beispiele auf google ai

```

CHIP toggle_ff GAL16V8
; Pin declarations
PIN 1  = clk
PIN 2  = enable
PIN 12 = q_out.R
EQUATIONS
; If enable is high, toggle the state on the clock edge.
; Otherwise, hold the current state.
q_out.d = (enable & !q_out) # (!enable & q_out)
;; also, solange ich galasm nicht auswendig gelernt habe, PINS, EQUATIONS und Abweichungen
was man allerdings als "ubliche Schreibweise zum ueben einf"uhren kann ist
y = (/X1 & X2) # (X0 & /X1)
;; ich glaube danach kommt noch
DESCRIPTION
a simple formular 4 galasm
;; ok, dann vielleicht erst mal das einfachste, doch das ist eine Idee:

```

1. faellt mir ein: ah! heute komplexe Schaltwerke, habe ich frueher schon gelebt
2. ach: galasm und abel ähnliches benutze ich ja seit neuestem, komme aber gerade mit DNF klar

3. wieso? DNF ist einfach zu machen, muesste gelebt werden, no problem, ohne auswendig lernen, meine ich ja, zu grosse Luftsprünge immer wieder einhält gebieten, wenn man

jetzt hier ist, reicht es in einem Monat auch noch

- soll nicht heissen, DNF wird dann nicht geuebt

das mache ich gleich...

gleich, noch das Unterprogramm selber...

ok wenn das geschehen ist, geht es danach weiter, ein bisschen std. abel. wuerde ich sagen, bzw. galasm

und wenn ich das habe, dann Basic und spaeter komplexe Schaltwerke inzwischen zigarattenpause...

```
%%\documentclass[a4paper]{article}
%%\usepackage{tikz,german,amsmath,amsfonts}
%%\usetikzlibrary{shapes.geometric, arrows.meta, positioning}
%%\author{david vajda}
%%\title{m8 wait loop led count 07/02/2026 in asm diagramm TikZ \LaTeX}
%%\begin{document}
%%\maketitle
%%\begin{tikzpicture}[
% Allgemeine Stile definieren
state/.style={rectangle, draw=black, thick, minimum width=2.5cm, minimum height=1cm, a
decision/.style={diamond, draw=black, thick, minimum width=2.5cm, minimum height=1cm,
arrow/.style={-Stealth, thick},
callbox/.style={rectangle, draw=black, thick, double, double distance=1.5pt, minimum w
% Zust"ande und Entscheidungen platzieren
%%\node (start) [state] {Start};
%%\node (z1) [state, below=of start] {Zustand 1\\ \verb"ldi r16, HIGH (RAMEND)"};
%%\node (z2) [state, below=of z1] {Zustand 2\\ \verb"out SPH, r16"};
%%\node (z3) [state, below=of z2] {Zustand 3\\ \verb"ldi r16, LOW (RAMEND) "};
%%\node (z4) [state, below=of z3] {Zustand 4\\ \verb"out SPL, r16"};
%%\node (z5) [state, below=of z4] {Zustand 5\\ \verb"ldi r16, 0xff"};
%%\node (z6) [state, below=of z5] {Zustand 6\\ \verb"out DDRB, r16"};
%%\node (z7) [state, below=of z6] {Zustand 7\\ \verb"ldi r16, 0xff "};
%%\node (z8) [state, below=of z7] {Zustand 8\\ \verb"main001: out PORTB, r16"};
%% \node (z9) [state, below=of z8] {Zustand 9\\ \verb"rcall waitloop001"};
%%\node (z9) [callbox, below=of z8] {Zustand 9\\ \verb"rcall waitloop001"};
%%\node (z10) [state, below=of z9] {Zustand 10\\ \verb"dec r16"};
%%\node (z11) [state, below=of z10] {Zustand 11\\ \verb"rjmp main001"};

%%\draw [arrow] (start) -- (z1);
```

```

%%\draw [arrow] (z1) -- (z2);
%%\draw [arrow] (z2) -- (z3);
%%\draw [arrow] (z3) -- (z4);
%%\draw [arrow] (z4) -- (z5);
%%\draw [arrow] (z5) -- (z6);
%%\draw [arrow] (z6) -- (z7);
%%\draw [arrow] (z7) -- (z8);
%% \draw [arrow] (z8) -- (z9);
%% \draw [arrow] (z9) -- (z10);
%%\draw [arrow] (z10) -- (z11);
%%\draw [arrow] (z11) -- (z8)[right];
%% \draw [arrow] (z11.east) -- ++(1.5,0) |- (z8.east);

```

%%\end{tikzpicture}

hier der code:

```

;; david vajda
;; 07/02/2026
;; TikZ LaTeX Template m8 led count wait loop

.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

waitloop001:
push r16
push r17
push r18
ldi r16, 0x04
waitloop002:
ldi r17, 0xff
waitloop003:
ldi r18, 0xff
waitloop004:

```

```

dec r18
brne waitloop004
dec r17
brne waitloop003
dec r16
brne waitloop002
pop r18
pop r17
pop r16
ret

```

also, ich mache jetzt folgendes, ich mache zunaechst das asm Diagramm, fuer diesen code. so wie gesagt, mit hineinschreiben

dann - ich verweise noch darauf auf meiner homepage sind compiler zu finden, so wie Funktionsplotter - eine teilweise eigens geschriebene sprache, wurde implementiert, die nie fertig wurde, und dann noch: einen scheme compiler. besonders der ist eigentlich ganz lustig. der Funktionsplotter ist in php. ein Funktionsplotter benutzt ausdrücke mit Funktionen in Bakus-Naur Form - die einen ausschnitt aus den formalen sprachen darstellen

ich habe ja frueher bereits komplexe Schaltwerke geuebt, weil mir nichts besseres einfel fuer den Simulator -

keine sorge, eigentlich alle Übungspapier sind noch da, vielleicht ein paar verloren - nahm ich einen Basic compiler, der das umgekehrte tat, tue ich manchmal, wenn man arithmetische ausdrücke erzeugen will, entscheidet man nicht ueber if (...) indem man den code scannt - wo es als naechstes lang geht, sondern ersetzt das durch `Random () mod 2` um zu einer Entscheidung zu kommen

ich hatte ja so oder so vor - jetzt Basic mir an zu gucken, eine relativ einfache sprache, einen besonders abgespeckten compiler will ich selbst fuer den atmega8 schreiben, also die sprache implementieren und auf den Controller drauf tun

also reines kleines basic, selbst schreiben. gut

dann kommt hinzu, sind wir wieder bei komplexen Schaltwerken, will ich noch folgendes ausprobieren, diese enthalten ja registe... lassen sich also mit ABEL, solange der Baustein entsprechend Kapazität hat - im speichertechnischen sinne - auch auf so einen GAL. GAL16v8 kann manchmal ein bisschen knapp sein, vermute ich, aber mal gucken... jetzt das Diagramm, dann Basic...

spaeter komplexes Schaltwerk...

```

;; david vajda
;; 07/02/2026
;; TikZ LaTeX Template m8 led count wait loop

.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

waitloop001:
push r16
push r17
push r18
ldi r16, 0x04
waitloop002:
ldi r17, 0xff
waitloop003:
ldi r18, 0xff
waitloop004:
dec r18
brne waitloop004
dec r17
brne waitloop003
dec r16
brne waitloop002
pop r18
pop r17
pop r16
ret

```

ich probiere es jetzt so: dass ich so zu sagen, das von gerade eben nehme, aber total eckige zustände fuer Anweisungen - und: noch einen assembler Quelltext fuer m8 - zaehschleife mehrfach und dann füge ich den code in $\text{\LaTeX}$ code ein und mache einzeln drum rum die boxen...

```

%% war google ki
%% \documentclass [a4paper]{article}
%% \usepackage{tikz}
%% \usepackage{german,amsmath,amsfonts}
%% \usetikzlibrary{shapes.geometric, arrows.meta, positioning}
%% \begin{document}
%% \begin{tikzpicture}[
%%   state/.style={rectangle, draw, minimum width=2cm, minimum height=1cm, text centered},
%%   % Knoten definieren

```

```

%% \node (state1) [state] {Zustand A};
%% \node (dec1) [decision, below=of state1] {B = 1?};
%% \node (state2) [state, right=of dec1] {Zustand B};

% Pfade/Pfeile zeichnen
%% \draw [arrow] (state1) -- (dec1);
%% \draw [arrow] (dec1) -- node[above] {Ja} (state2);
%% \end{tikzpicture}
%% \end{document}
%%

```

ASM diagramm scan da ist tatsaechlich ein Fehler drin, die raute gibt es an und fuer sich nicht...

davor muss tatsaechlich eine leere breite box...

Weil ich mal wieder nicht in mein Zimmer kommen, kann ich das grad nicht besser machen

Das ist jetzt das eine ASM Diagramm zu einem Quelltext

Binäre Entscheidung Diagramm folgt. Ich hab jetzt grad nur mein Smartphone. Ich mach jetzt die anderen Beiträge mit. Stelle ich an der Stelle dann vor bis gerade auch glaube ich notwendig.

So, ich mach nachher weiter wenn der Computer gut tut wenn ich wieder an den Computer kommen, dann mache ich nachher aus diesem Beispielen hier. Das muss noch ordentlich dargestellt werden und ausgedruckt werden, dass man es in Bilder verwandeln kann. Ordentliches ASM Diagramm ordentliches Flow , Chart und ordentlich alles wie es halt so ist und na das, was der Psychologe sagen würde, wenn er am Papier sitzt und davon ausgeht, dass keine Angreifer der Welt rumlaufen und laut Zeug die ganzen Modelle nachher ordentlich auf Papier danke schön

06:23

57



Start

Ja

Nein

Entscheidung
1

Handlung 2

Handlung 3

Entscheidung
2

06:59

51



ldi r16, HIGH (RAMEND)



out SPH, r16



ldi r16, LOW (RAMEND)



out SPL, r16

ldi r16, 0xff
44



11:26

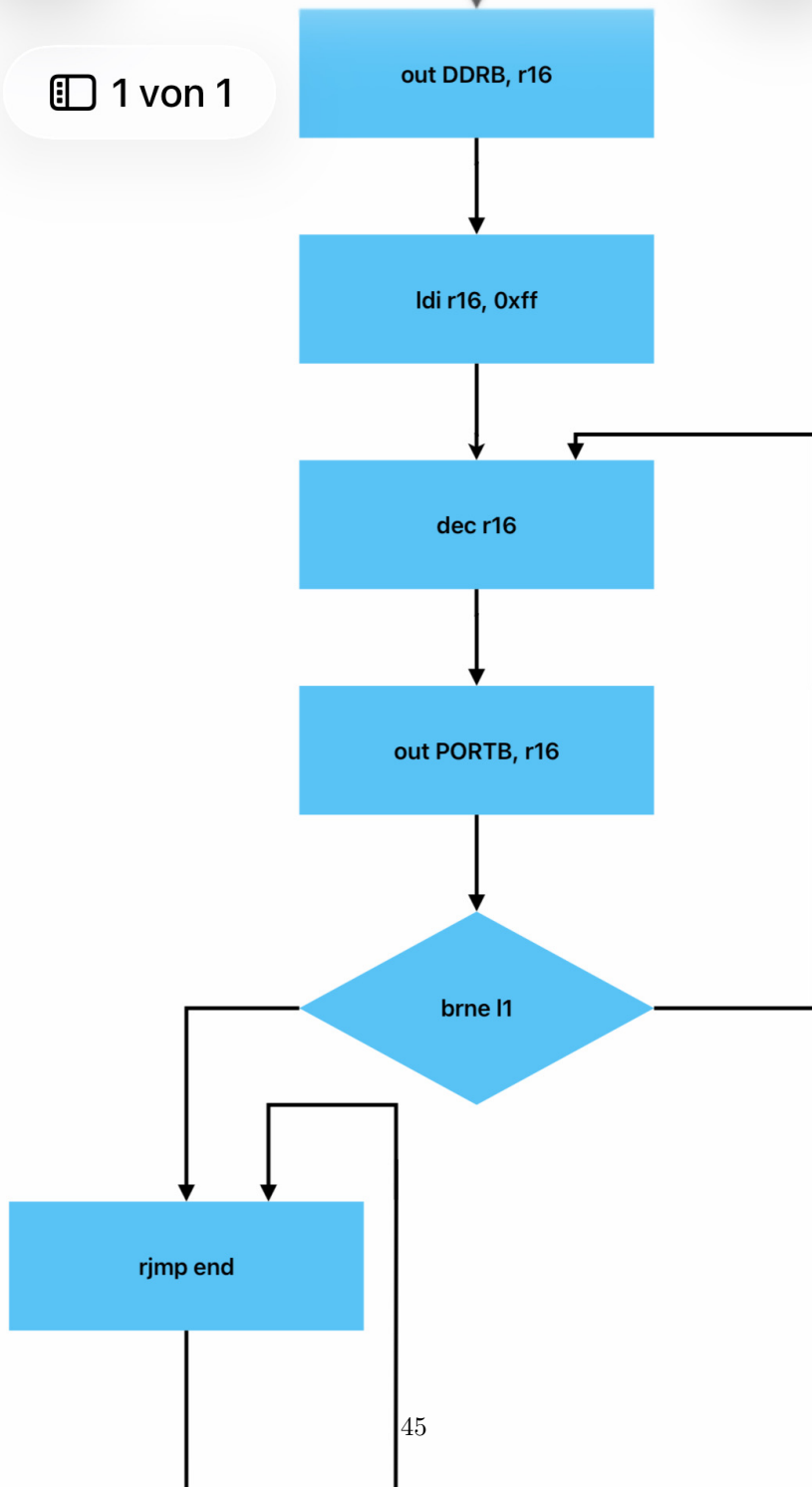
34



countoszilooop07012026



1 von 1



11:26



countosziloo07012026



1 von 1

ldi r16, HIGH (RAMEND)



out SPH, r16



ldi r16, LOW (RAMEND)



out SPL, r16



ldi r16, 0xff



out DDRB, r16

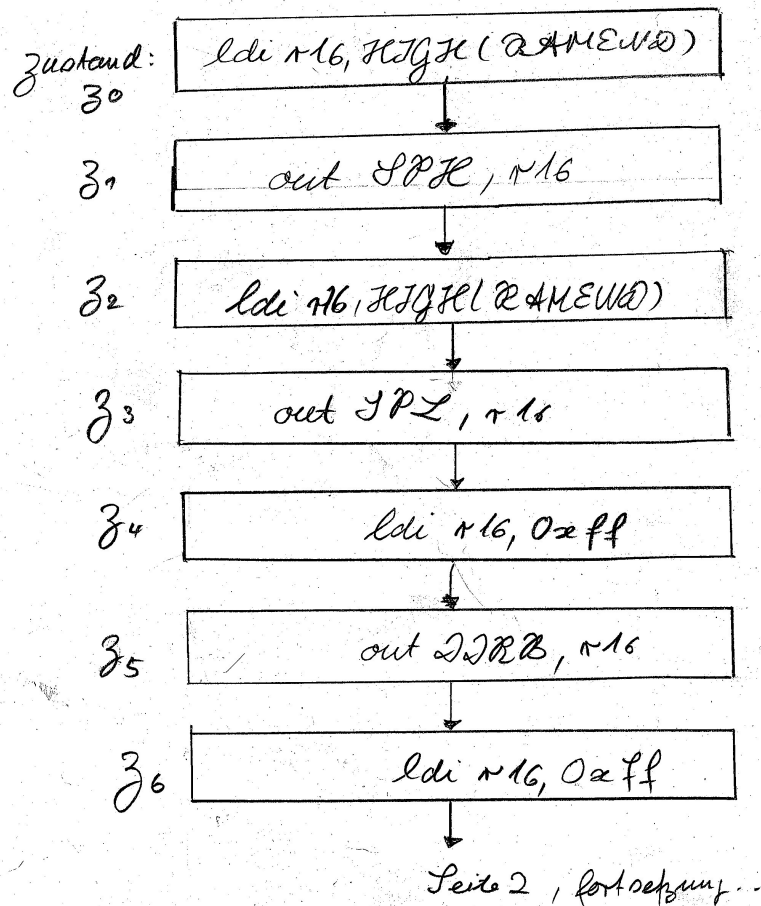


46

ldi r16, 0xff



David Vajda
07/01/2026
asm-Diagramm...
Seite 1 Übung
wait-Loop... per m8...

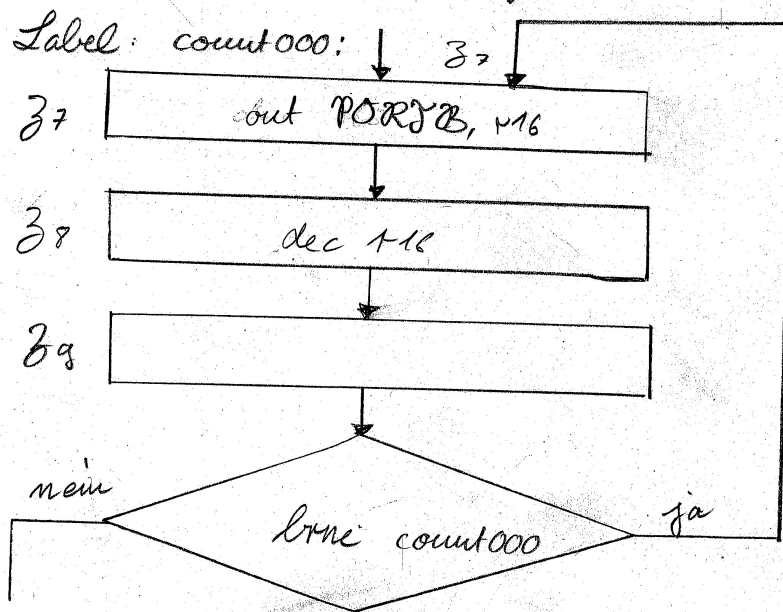


David Vayda

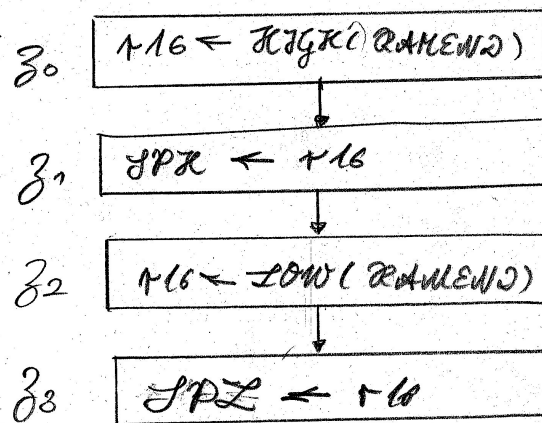
07/01/2026

asm-Diagramm ... Seite 2

übung, wait-Loop ... m8
atmega8



David Vyda
 Seite 3, asm-Diagramme
 m8, für Hardwarealgorithmen...
 mit RTL-Notation...



ok, diese übung mache ich
 heute nicht fertig...

Ich komme gerade nicht weiter morgens Zeit zum aufstehen gerade keinen pc
 um Bild daraus zu machen, deswegen späte...

Also: ausgabe - so weit es geht: als: led up count mit dem atmega8

Binäres entscheidungsdiagramm könnte eigentlich ein asm Diagramm sein, man
 muss sehen das ein binäres entscheidungsdiagramm auch vielleicht ohne Anwei-
 sungen möglich ist Wichtig beim asm Diagramm:

Gerichtete Kanten: es gibt Pfeile (gerichtete Kanten, gerichteter graph)

Ein Graph kann zum Beispiel Zykelfrei oder mit Zyklen sein. Zyklen können auch mit gerichteten kanten vollzogen werden, aber ein asm Diagramm nur in eine Richtung nicht rückwärts

Thema Graph: ein Knoten hat bei einem asm Diagramm immer die valenz drei, damit sind eulertouren nicht möglich und: ein Graph der Knoten enthält die 4, 5, 6, ... kanten aufweisen, valenz, müssen zerlegt werden in 2, 3, 4, ...

Entscheidungen ASM Diagramm enthalten noch die RTL Sprache Register Transfer Level

```
R1 <- R1 + 1
```

- Register: R, durchnummerieren
- Arbeitsspeicher: M
- Bits in Register R1: [7..0] ??? Nicht ganz sicher
- Adressen im RAM M [7:0]
- Stackpointer: SP
- Programm Counter: PC
- Transfer (laden, speichern, Register<-Register) R1 <- R1
- Increment: ++
- Decrement: --
- Linksshift: <<
- Rechtsshift: >>
- And, or, not: &&, ||, !
 1. Zuweisung (Transfer)
 2. arithmetisch
 3. logisch
 4. bitweise
- genauer:
 1. Zuweisung (Transfer) und Vergleich
 - (a) Transfer: <-
 - (b) <, =, >, lt, eq, gt
 - (c) (ge, le, ne)
 2. arithmetisch:
 - (a) Addition: +

- David Vexla 07/06/2026 Seite 2
Linnemittelfahrt...
1 * und es sind die Linnen
und füllt (die) die Linnen
und füllt ... 2 Linnen sind
1 2 3 Glauben Linnen
4 5 6 7 8 9 10
11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30
31 32 33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48 49 50
51 52 53 54 55 56 57 58 59 60
61 62 63 64 65 66 67 68 69 70
71 72 73 74 75 76 77 78 79 80
81 82 83 84 85 86 87 88 89 90
91 92 93 94 95 96 97 98 99 100
101 102 103 104 105 106 107 108 109 110
111 112 113 114 115 116 117 118 119 120
121 122 123 124 125 126 127 128 129 130
131 132 133 134 135 136 137 138 139 140
141 142 143 144 145 146 147 148 149 150
151 152 153 154 155 156 157 158 159 160
161 162 163 164 165 166 167 168 169 170
171 172 173 174 175 176 177 178 179 180
181 182 183 184 185 186 187 188 189 190
191 192 193 194 195 196 197 198 199 200
201 202 203 204 205 206 207 208 209 210
211 212 213 214 215 216 217 218 219 220
221 222 223 224 225 226 227 228 229 230
231 232 233 234 235 236 237 238 239 240
241 242 243 244 245 246 247 248 249 250
251 252 253 254 255 256 257 258 259 260
261 262 263 264 265 266 267 268 269 270
271 272 273 274 275 276 277 278 279 280
281 282 283 284 285 286 287 288 289 290
291 292 293 294 295 296 297 298 299 300
301 302 303 304 305 306 307 308 309 310
311 312 313 314 315 316 317 318 319 320
321 322 323 324 325 326 327 328 329 330
331 332 333 334 335 336 337 338 339 340
341 342 343 344 345 346 347 348 349 350
351 352 353 354 355 356 357 358 359 360
361 362 363 364 365 366 367 368 369 370
371 372 373 374 375 376 377 378 379 380
381 382 383 384 385 386 387 388 389 390
391 392 393 394 395 396 397 398 399 400
401 402 403 404 405 406 407 408 409 410
411 412 413 414 415 416 417 418 419 420
421 422 423 424 425 426 427 428 429 430
431 432 433 434 435 436 437 438 439 440
441 442 443 444 445 446 447 448 449 450
451 452 453 454 455 456 457 458 459 460
461 462 463 464 465 466 467 468 469 470
471 472 473 474 475 476 477 478 479 480
481 482 483 484 485 486 487 488 489 490
491 492 493 494 495 496 497 498 499 500
501 502 503 504 505 506 507 508 509 510
511 512 513 514 515 516 517 518 519 520
521 522 523 524 525 526 527 528 529 530
531 532 533 534 535 536 537 538 539 540
541 542 543 544 545 546 547 548 549 550
551 552 553 554 555 556 557 558 559 560
561 562 563 564 565 566 567 568 569 570
571 572 573 574 575 576 577 578 579 580
581 582 583 584 585 586 587 588 589 590
591 592 593 594 595 596 597 598 599 600
601 602 603 604 605 606 607 608 609 610
611 612 613 614 615 616 617 618 619 620
621 622 623 624 625 626 627 628 629 630
631 632 633 634 635 636 637 638 639 640
641 642 643 644 645 646 647 648 649 650
651 652 653 654 655 656 657 658 659 660
661 662 663 664 665 666 667 668 669 670
671 672 673 674 675 676 677 678 679 680
681 682 683 684 685 686 687 688 689 690
691 692 693 694 695 696 697 698 699 700
701 702 703 704 705 706 707 708 709 710
711 712 713 714 715 716 717 718 719 720
721 722 723 724 725 726 727 728 729 730
731 732 733 734 735 736 737 738 739 740
741 742 743 744 745 746 747 748 749 750
751 752 753 754 755 756 757 758 759 760
761 762 763 764 765 766 767 768 769 770
771 772 773 774 775 776 777 778 779 780
781 782 783 784 785 786 787 788 789 790
791 792 793 794 795 796 797 798 799 800
801 802 803 804 805 806 807 808 809 810
811 812 813 814 815 816 817 818 819 820
821 822 823 824 825 826 827 828 829 830
831 832 833 834 835 836 837 838 839 840
841 842 843 844 845 846 847 848 849 850
851 852 853 854 855 856 857 858 859 860
861 862 863 864 865 866 867 868 869 870
871 872 873 874 875 876 877 878 879 880
881 882 883 884 885 886 887 888 889 890
891 892 893 894 895 896 897 898 899 900
901 902 903 904 905 906 907 908 909 910
911 912 913 914 915 916 917 918 919 920
921 922 923 924 925 926 927 928 929 930
931 932 933 934 935 936 937 938 939 940
941 942 943 944 945 946 947 948 949 950
951 952 953 954 955 956 957 958 959 960
961 962 963 964 965 966 967 968 969 970
971 972 973 974 975 976 977 978 979 980
981 982 983 984 985 986 987 988 989 990
991 992 993 994 995 996 997 998 999 1000
1001 1002 1003 1004 1005 1006 1007

Thunfisch iingung out of
 the sea... ok Last 1
 from a boat..

Unit 1 07/06/2026

Linke Lücke nur 1 Name
 mit Lückenübung auf mit
 Lösung genau richtig from a.
 Wollen nur 1 iingung aber
 meinte Lolo Lolo Lolo
 Lolo Lolo Lolo Lolo..

1 2 3 4 5

6



ich soll hier hingehen

Lücke Linke Lücke
 Lücke nur Lücke
 Linke Name (1) Unit
 wenn ich Lücke mit quere

ist gegeben?

Unit 1 07/06/2026

Wenn Du jst
 von jst hnung...
 mit 2 un
 zu jst hnung...
 zu jst hnung...



Lieber
 Weil ich
 jst hnung...
 ein mit
 zu jst hnung...
 jst hnung...

mit dem jst hnung...
 jst hnung...
 jst hnung...
 jst hnung...
 jst hnung...
 jst hnung...

Nousin' Wajita... (C)
Kommunikationshilfsmittel...

Thema 1 07/06/2026

Kling und fröhlich.

Obwohl ich sehr...

Kling und fröhlich.

Ich... habe ich ein...
Hochkommunikations...

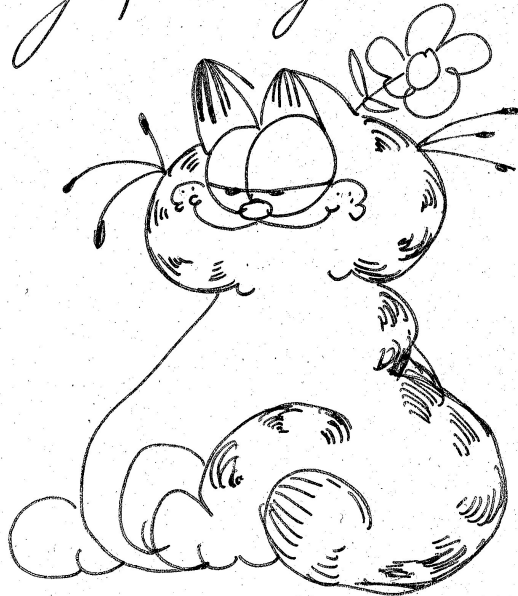
Lauter in der
Lauter Lauter sein.
Auf ein Lauter...

Lauter Lauter o.
Lauter Lauter o. Lauter

Lauter o. Lauter in
Lauter Lauter...

Kennst Duja auf
 dem Land auf dem roten
 in der Mitte. 07/06/2026
 Die Zimmungs-fähigkeit
 beschränkt im Alter auf was
 dem Kind ist Zimmungs-
 fallen sehr. Die
 nur gucken die Mutter
 in den Käse da finden
 die alles um ihnen zu
 sagen. Man sie mit dem jungen
 ungesunden Kindern um
 dem von der Kraft zum
 Lagerung. Inzwischen
 die Le Gegenwart. Die
 Coa Coa für den
 (Mutter) -

Horst Majika.
 Mitte 1. Jahr
 Gefährliche Klettertaste: ... 07106
 12026



und nun mit
 Lulle ihr Herz
 so ist ein zufriedenes
 Kind mit gulli-
 Köpfen Kiste.

Mumukshu iigung out of
the god... ok last 4
fours a four.

Unit 1 07/06/2026

Linke Lombei nun Stamm
mit Lombeinipfen auf mit
Lomgen grossenlipf founn o.
Wollte nur y iigen aber
mainta Lolo Lolo Lolo
Lola lola loLaLo.

1 2 3 4 5

6 7

ich fahn fuff

Lomukht Linke Lomke
Lomliopn nur Lomliopn
Linke Stamm (V) Mit
wasom hat Lom mit qucken

ist gahret?
gast
Lomgeholz allum

David Veyla 07/06/2026
Linnontsef... Seite 2


 out of first the thunder
 pulled (the) the Lüger
 out thugs ...  Lüpling first
   Glauber thinner
from right

A collection of hand-drawn symbols in blue ink. At the top left is a lightning bolt. To its right is a question mark. Below the lightning bolt is a large, stylized character that looks like '5' or 'S'. To the right of that is a large '6'. Further right is a small '4'. At the bottom right is a stylized character that looks like '7' or 'F'. There are also some smaller, less distinct marks scattered around.

Das ist sein nicht
Lammstötter für den kommt

Wird zusammen?
 Dein fin. Grouk furo
 Wirst mittler Zeit 70
 Jahren jetzt noch *¹⁰

Mein Majesta
 your soft humming...
 unfern
 Mitre 2 Lillian sind
 you persönlich!



Lillian
 Weil ich
 persönlich
 bin nicht
 you nur
 gelippen
 you

mit dem folgt nur
 sehr mein Wissen gegen
 Lillian dankt mich
 für ihre aufmerksamkeit gegen
 Lillian (for) um you you
 your Wissen...

Kommt Majda auf
 dem Lande auf dem Boden
 in der Mitte. 07/06/2026
 die Zusammenfassung-fähigkeit
 beschränkt im Alter auf was
 dem Kind ist Zusammenfassung
 falls sehr viele
 nur gucken die Mutter
 in die Käse da sind
 die alle um ihnen zu
 sagen. Wie sie mit dem jungen
 umgehen könnten um
 dem Kind ihre Kraft zu
 Legeinbringung. Inzwischen
 die die Gegenwart. Die
 Lora Lora für den
 Austritt. -

Waisin Wajita... (C)
Waisin Wajita...

Seite 1 07/06/2026

Wing und Frimbu.

Oben ist es so.

Wing und Frimbu.

W... Wabai ist ein Wabai
Waisin Wajita

Wabai in Wabai
Wabai Wabai Wabai Wabai
Auf Wabai Wabai

W... Wabai Wabai o.
W... Wabai o. W... Wabai

Wabai o. W... Wabai in
Wabai Wabai Wabai Wabai...

Karin Majster.
 Seite 1 nur
 Gefühl Melancholie: ... 07/06
 12026



und nun mit
 Lulle nur Herz Art
 ja ja ist ein zufriedener
 ganz und gulli.
 Köpfe Kiste.

Jaund Ujda 62 05/07/2026
 binäres Zählen schriftlich Seite 2

$$\begin{array}{r}
 517 : \quad \text{bin}(517d) \text{ ist: } 1000001010 \\
 517/2 = 258 \text{ Rest } 1 \\
 \begin{array}{r}
 -4 \\
 \hline
 11 \quad -2 \\
 -10 \quad -05 \\
 \hline
 17 \quad -04 \\
 -16 \quad -18 \\
 \hline
 1 \quad 0 \\
 \hline
 1
 \end{array}
 \end{array}$$

$$\begin{array}{r}
 258/2 = 129 \text{ Rest } 0 \\
 129/2 = 64 \text{ Rest } 1 \\
 \begin{array}{r}
 -0 \\
 \hline
 05 \quad -12 \\
 -04 \quad -12 \\
 \hline
 18 \quad -08 \\
 -18 \quad -7 \\
 \hline
 0 \quad 1
 \end{array}
 \end{array}$$

$$\begin{array}{r}
 64/2 = 32 \text{ Rest } 0 \\
 32/2 = 16 \text{ Rest } 0 \\
 16/2 = 8 \text{ Rest } 0 \\
 8/2 = 4 \text{ Rest } 0 \\
 4/2 = 2 \text{ Rest } 0 \\
 2/2 = 1 \text{ Rest } 0 \\
 1/2 = 0 \text{ Rest } 1
 \end{array}$$

also: $\text{bin}(517d) = 1000001010$

als einfache methode der überprüfung

512 sind bit 9 on. bzw. 29 =

512 wg.
 nein da wurde ein 0 vergessen!
 aber oben schaut man. Ja, der
 Rand steht da 6x bei *

das ist logisch.

1111.1111b sind 255

das muß so sein weil:

$2^8 = 256$ wir brauchen
 aber bei bit 8. bit 0 also

1.0000.0000b ist (ab) 256d ok

1.1111.1111b ist 511 und

10.0000.0000b ist 512. ok dann +5

ok 0101b stimmt!

David Meyer (c) 06/07/2026, Seite 3 Kattner's algorithm

i	MLI	sum1	sum2	sum'	sum0'
0	68	0	0	68	68
1	50	68	68	119	186
2	89	167			
3	107	$274 + 68 = 342$			
4	124	$342 + 107 = 449$			
5	24	$449 + 124 = 573$			
6	84	$573 + 24 = 597$			
7	86	$597 + 84 = 681$			
8	12	$681 + 86 = 767$			
9	71	$767 + 12 = 779$			

$$\text{sum1}' = (\text{sum1} + \text{MLI}) \bmod 255 \quad \text{für sum1} = 0$$

$$\text{sum2}' = (\text{sum1} + \text{sum2}) \bmod 255 \quad \text{für sum2} = 0$$

ob in der Tabelle:

mane gleich viele neue Werte so am Ort mal.

sum1 erzeugt werden...

2) wird es wie größer 255 dann die modiert.

$$x = (x + 255) \bmod 255 \quad \text{und die Bewegung MLI} \leq 255$$

dann man es addieren also $x = (x + 255) \bmod 255$

3) in manne neue Tabelle gibt man mal!

David Vajda Fletcher's checksum
06/07/2026, Seite 1

```

sum1' = sum 1 + vu[i] modulo 255, mit sum 1 i=0 = 0
sum2' = sum 2 + sum1' modulo 255, mit sum 1 i=0 = 0

schritt=(1) in [i]
1. schritt i=0      68      sum 1      sum 2      sum1'      sum 2
2. schritt i=1      98      0          0          68      68
3. schritt i=2      99      98      68  →  118      118+68 = 186
i=3      102          217=118+99      186      186
i=4      104          104
i=5      94          94
i=6      96          96
i=7      100         100
i=8      97          97
i=9      91
i=10     71

```

mit Handrechner:

06/07/2026, (c) David Noyes
 Hether's Meadow + Washfield, site 5

Wash-field: Wednesday

order $y_i = ((u[i].m + y_i) \% L)$
 $y_{i+1} = ((u[i].m + y_i) \% L)$
 or $y_0 = 0$ or $y_i = \dots$ kernel size

i	y_i	$u[i]$	m	$u[i].m$	$u[i].m + y_i + 1$	% 100
0	68	68	10	680	680	68
1	50	50	10	500	500	50
2	89	89	10	890	890	89
3	104	104	10	1040	1040	104
4	194	194	10	1940	1940	194
5	74	74	10	740	740	74
6	94	94	10	940	940	94
7	19	19	10	190	190	19
8	71	71	10	710	710	71

$304 + 68 = 568$
 $304 + 68 = 568$
 $1070 + 58 = 1128$
 $1070 + 58 = 1128$

68
 58
 58
 68

or: $m + 1$ right $m = 100$
 and days now chosen

or: 1
 right

man keep next 20 next! Wednesday!

$(y_i \cdot L + u[i]) \% m \leftarrow$ day not ok
 keep next 20 next! Thursday: of not table

David Nyda 07/06/2026 Seite 6
Kard. Funktion + Tabelle...

David Heyes (c) 05/07/2026
 Binäres dividieren, schriftlich, Seite 3

Jetzt: $21a$

$$\text{Div}(21a) =$$

$$21/2 = 10 \text{ Rest } 1$$

$$10/2 = 5 \text{ Rest } 0$$

$$5/2 = 2 \text{ Rest } 1$$

$$2/2 = 1 \text{ Rest } 0$$

$$1/2 = 0 \text{ Rest } 1$$

$$\text{ab: } 16 + 5 = 21$$

$$\text{wird } \text{Div}(16a) = 10000a$$

$$\text{Div}(16a) = 1010a \dots \text{ab!}$$

ab muss schriftliche Division sein

Prüfung:

$$\text{Div}(510a) = 00100000001010$$

$$\text{Div}(21a) = 000000001010$$

$$\text{abot: } 1:01010$$

$$\text{ab: } 001000000010101 = 00$$

$$\begin{array}{r} 00000 \\ - 10101 \\ \hline 0 \end{array}$$

Das ist falsch, muss zusammenfassen!

Der Rest ist

00100 und dann weiter!

David Lloyd (2) Privates Algorithmen
+ Buchstaben + Zahlen + ...

05/07/2020, Seite 7:

of

$0 \oplus 0 \oplus 0 \oplus 0 \oplus 0$ jede Frage

$x_2 \oplus x_1 \oplus x_0 := x$

$(0 \oplus 0) \oplus (0 \oplus 0) =$

$0 \oplus 0 = 0$ d.h. $Q := \text{no}$ (falsch/keine)

$E := \text{yes}$ (falsch/keine)

$(0 \oplus 0) \oplus (0 \oplus 1) =$

$0 \oplus 1 = 1$ $Q := \text{yes}$ $E := \text{no}$

$(0 \oplus 0) \oplus (1 \oplus 0) = 0 \oplus 0 = 0$ $Q := \text{yes}$ $E := \text{no}$

$(0 \oplus 0) \oplus (1 \oplus 1) = 0 \oplus 0 = 0$ $Q := \text{yes}$ $E := \text{no}$

$(0 \oplus 1) \oplus (0 \oplus 0) = 1 \oplus 0 = 1$ $Q := \text{no}$ $E := \text{yes}$

$(0 \oplus 1) \oplus (0 \oplus 1) = 1 \oplus 1 = 0$ $Q := \text{no}$ $E := \text{yes}$

$(0 \oplus 1) \oplus (1 \oplus 1) = 1 \oplus 0 = 1$ $Q := \text{yes}$ $E := \text{no}$

$(1 \oplus 0) \oplus (0 \oplus 0) = 1 \oplus 0 = 1$ $Q := \text{yes}$ $E := \text{no}$

$(1 \oplus 0) \oplus (0 \oplus 1) = 1 \oplus 1 = 0$ $Q := \text{no}$ $E := \text{yes}$

$(1 \oplus 0) \oplus (1 \oplus 1) = 0 \oplus 1 = 1$ $Q := \text{yes}$ $E := \text{no}$

$(1 \oplus 0) \oplus (1 \oplus 0) = 0 \oplus 0 = 0$ $Q := \text{no}$ $E := \text{yes}$

$(1 \oplus 1) = 0$ $Q := \text{falsch}$ $E := \text{wahr}$

$(1 \oplus 0) \oplus (1 \oplus 1) = 0 \oplus 0 = 0$ $Q := \text{falsch}$ $E := \text{wahr}$

$(1 \oplus 0) \oplus (1 \oplus 1) = 0 \oplus 0 = 0$ $Q := \text{falsch}$ $E := \text{wahr}$

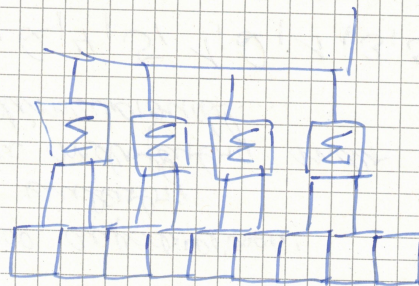
$(1 \oplus 1) \oplus (0 \oplus 0) = 0 \oplus 0 = 0$ $Q := \text{falsch}$ $E := \text{wahr}$

$(1 \oplus 1) \oplus (0 \oplus 1) = 0 \oplus 1 = 1$ $Q := \text{falsch}$ $E := \text{wahr}$

$0 \oplus 0 = 0$ $Q := \text{falsch}$ $E := \text{wahr}$

of ... modulare Frage

David Meyer (c) 05/07/2006
Party Size 9



das kann aber irgendeine
den nutzt man

wie geht heraus

2 Bit Ergebnisse
was?

st. Doppeladdierer

dec $1+1=2$ dec = 10000

$1+1+1+1=4$ dec = 10000

also 10 000 + 10 000

David Hayes (c) Wiley + Standard-
abrechnung + ... 05/07/2006, Zeit

11

Standardabweichung:

1.) Erwartungswert

1.) Ereignis / Ereignismenge

$\Omega = \{1, 2, 3, 4, 5, 6\}$

Ereignismenge: $\{1, 3, 5\}$

$\{2, 3, 4, 5, 6\}$

$\{1, 2, 3, 4, 5, 6\}$

ob: Wahrscheinlichkeit

Zufallsvariable:

$x_1 = 1, x_2 = 2, x_3 = 3$

$\mathcal{X}(1) = x_1, \mathcal{X}(4) = 4, \mathcal{X}(5) = 5$

$\mathcal{X}(6) = 6$ oh. Wahrscheinlichkeit

$p(1) = \frac{1}{6}, p(2) = \frac{1}{6}, p(3) = \frac{1}{6}$

$p(4) = \frac{1}{6}, p(5) = \frac{1}{6}$

$p(6) = \frac{1}{6}$

$\frac{1}{6} \cdot p(\sum_{i=1}^6 p(x_i) \cdot x_i) =$

Seite 14, Daniel Heyda (C)
05/07/2026, Hash tabelle +
Fletcher's checksum...

Algo:

```

sum1 := 0
sum2 := 0
for i := 0 to Length(M) do
    sum1 := (sum1 + M[i]) modulo 255
    sum2 := (sum2 + sum1) modulo 255
end for
checksum := (sum1, sum2)
return checksum

```

also: $M = (68, 50, 99, 107, 124, 74, 94, 96, 10, 21)$

1.) Schritt:

sum1 := 0
sum2 := 0

$M[i]$

↓

sum1 := (sum1 + 68) modulo 255
= sum1 := (68) modulo 255
= 68
sum2 := (0 + 68) modulo 255

ab: Schritt 1: sum1 := 68, sum2 := 68

2.) Schritt:

sum1 := (sum1 + $M[1]$) modulo 255
sum1 := (68 + 50) modulo 255
= 118 modulo 255
= 118

sum2 := (sum2 + sum1) modulo 255
= (68 + 118) modulo 255
= 186 modulo 255

sum1 := 118, sum2 := 186

3.) Schritt:

sum1 := (sum1 + $M[2]$) modulo 255
= (118 + 99) modulo 255
= 217 modulo 255
= 217

sum2 := (sum2 + sum1) modulo 255
= (186 + 217) modulo 255
= 403 modulo 255

$403 - 255 = 148$

sum1 := 217, sum2 := 148

Samuel Vajda (06) 05/07/2026
Erwartungswert + Würfeln...
Seite 12 ok...

$$\begin{aligned} & \frac{1}{6} \cdot 1 + \frac{1}{6} \cdot 2 + \frac{1}{6} \cdot 3 + \frac{1}{6} \cdot 4 \\ & + \frac{1}{6} \cdot 5 + \frac{1}{6} \cdot 6 \\ & = \frac{1}{6} \cdot (1 + 2 + 3 + 4 + 5 + 6) \\ & = \frac{1}{6} (3 + 7 + 11) = \frac{1}{6} \cdot 21 \\ & = 3,5 \text{ ok.} \end{aligned}$$

Standardabweichung + Varianz...

Jana Vajda (L) 05/07/2026

Minifelu, Seite 10
ok

pear, tuple aufschreiben:
erst einferlig
dann mehrferlig mehr
dimensional...

$W = (5, 5, 2, 4, 5, 1, 2, 1, 3, 5,$

$3, 4, 5, 4, 4, 1, 6, 2, 4, 5, 2, 5, 5,$

$3, 6, 1, 3, 4, 1, 6, 1, 3, 5, 3,$

$5, 2, 6, 1, 5, 1, 4, 6, 3, 4, 1, 6,$

$5, 6, 2, 6, 1, 3, 5, 4, 4)$

ok: Die Nadel (feder) von
meinen füller gerade machen
auf dem papier fassen, dass
heißt: keine Lust.

jetzt diagramm:
✓ papier

+ Erwartungswert

+ Standardabweichung..

m.

David Kopf (1) Binäres Addieren
Schriftlitz... + ... + Seite 8
05/07/2026

So: jetzt zählen wir binär.
1er addieren 8 bit

$b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0$

$b_7 + b_6$

$b_5 + b_4$

$b_3 + b_2$

$b_1 + b_0$

zwei addieren

101100010

12 addieren

1+0

zwei addieren

4 bit Register
einfach bitweise

aber darin dasselbe...

wenn wir das mit dezimal
machen

$8+9$ überschritt

modulo 9 oder 8...

David Vigda (1) Seite 6
 05/07/2026 binäres Rechnen
 schriftlich + ... + jetzt parity

xor := $\oplus$?
 oh

try: anzahl der bit
 der bit

addition

+

also

+

+

falls wird bit
 drei größer:

da: ist parity: odd/even

odd: ungerade, even: gerade + tafel

try with xor:

x_3, x_2, x_1, x_0
 0000
 0001
 0010
 0011
 0100
 0101
 0110
 0111
 1000
 1001
 1010
 1011

0 1

$0 \oplus 0 (x_3 \oplus x_2)$
 $= 0$
 $0 \oplus 0 (x_1 \oplus x_0)$
 $= 0$

nächste Seite.

Schriftlich...

beligade *

klar
ist
hier
es muss
zu
zu
gib.

Der Wip $\rightarrow$
binäres
dividieren:
entweder
00000
101010
ist größer
falsch

$$0010.00000101_2 / 010101_2 = 0011000^{*4}$$

~~10101--~~

00000

$$\begin{array}{r} 10000x \\ - 00000 \\ \hline \end{array}$$

100000

-010101

001011⁰*2

10110 * 32 Ergänzung

- 10101

[illegible]

(oh sounds: logical 1 + 10101)

$$10101 =$$

101101

nebenbei das
war die letzte

← aber trotzdem

17..

1	10	
1	10	1

1101

* 2. prole:

$$\begin{array}{r} 101010 \\ + 10110 \\ \hline \end{array}$$

1.0000 oh! ✓!

ok: ? $11000 \cdot 10101 + 1101$

Ang nächste Seite + Ausrechnen!

Summa Seite 4, 06.07.2026, Petrus à Carthago.

1. $\frac{0}{123456789}$ $\frac{0}{123456789}$ $\frac{0}{123456789}$ $\frac{0}{123456789}$

Da ist alles neu aber habe
erkennen gelernt:

die andere ja ist aus
und aus' werden
und die kann ich sicher leichter
finden aus' werden.

2. $\frac{0}{123456789}$ $\frac{0}{123456789}$ $\frac{0}{123456789}$ $\frac{0}{123456789}$

gibst du aus' werden.

David Heyda (1905/07-19006) Zeit 15, Festeins Chodovum $\mu = (68,50, 89, 107, 124, 74, 94, 96, 12, 71)$	4.) Zeit: gibt es eine einfache Methode 3. B: Fester's Chodovum aufge adre: den: ja! tabelle!	1. Zeit: 2. Zeit:
1. Zeit: WLRJ ist bekannt! 2. Zeit: man an WLRJ: ist es eigentlich möglich, alle Werte zusammen zu geben auch Best zu machen	3.) Zeit: man ist dem Fester's Chodovum! aber: table adre! WLRJ ist bekannt!	1. Zeit: 2. Zeit:
ob im WLRJ tabelle den Fester's Zeit	1. Zeit: 2. Zeit:	1. Zeit: 2. Zeit:

Einführung: die Stelle hat laut man sein Grade zu
was es eigentlich irgendwelche Beschaffenheit ist. Es ist die
zu wie auch! Hier wurde und es und zu sagt eine was
und auf den malz erfüllt! Kopf: der ganz Grad ist mehr
mzg's-Beschaffen - die Besondere nur zu sind. Ebenso wie
das werden! Sie schaffen Leben was eine andere
ist. In der Lage viele etwas!
für Tiere und nicht andere!

Ja werde immer: die wurde mag immer ja alle
beugen! Kopf: wurde wie geht von! Es ist nicht
immer das Leben mit etwas noch dazu!
in alle andere: Schupfmaschine. Diese haben verschiedene
Beiden auch: Schupfmaschine? Wenn aber: Schupfmaschine!
in alle ganz - Kopf - Schupfmaschine mehr etwas
an: fast mehr

CE) und Lyda 06/07/2026, Seite 2
Hofes: Leben und Natur!

Jawab: Bayan (c) Brinawati
 Skript Kalkulasi Dividenden, Slide 1
 05/10/2026

atau:

$$\text{quotient} = \frac{\text{dividenda}}{\text{divisor}} + \text{rest}$$

rebt: R
 dividend: a
 quotient: q

$$q := \frac{R}{a} + r$$

$$R = 512 + 5 = 517$$

atau:

$$\frac{R}{a} = q + r$$

atau
 formor
 dengan
 rest

$$q + r = \frac{R}{a} \text{ karena } |r| < |a|$$

manus harus

[q] und [q] ausgerechnet und
 [q] < [rest] sein: [q] ist richtig!
 5,123 Rest

$$5 \text{ und } a: 0,123 = 0,123 \cdot 5$$

oder: $q := \frac{R}{a} + r$
 dann ist diese
 Formel oder
 diese Rechnung
 richtig!

$$q - r := \frac{R}{a}$$

$$q + r = \frac{R}{a} \text{ da } q \text{ ganzzahlig}$$

da auch diese
 Formel
 richtig ist!

15:59



facebook

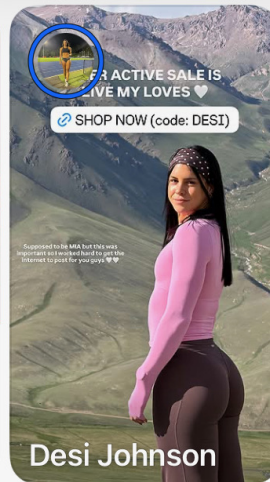
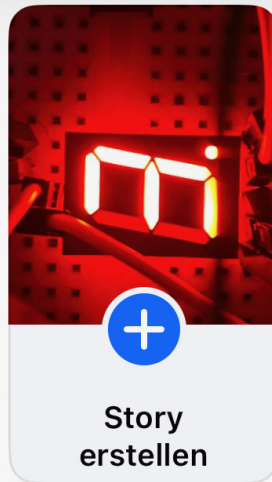
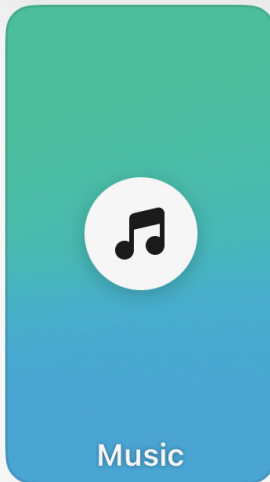


Was machst du gerade?



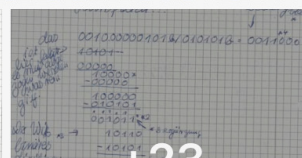
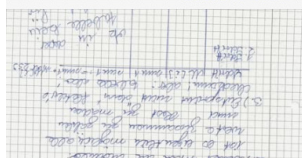
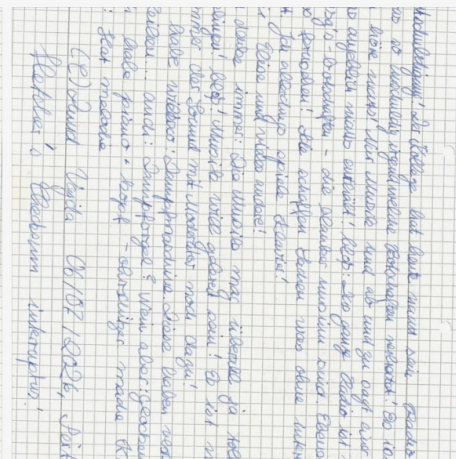
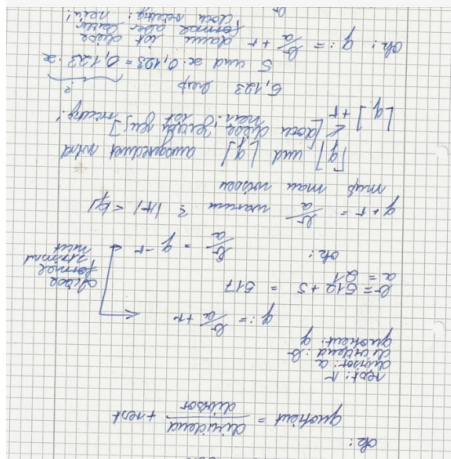
Stories

Reels



David Vajda

Gerade eben • 🌐



David Vyda (c)

07/09/2026, Seite 1

würfeln...

$E = \{1, 5, 4, 6, 1, 6, 2, 6, 5, 1, 6, 3, 5, 3, \backslash$
1, 4, 1, 2, 2, 5, 2, 5, 6, 2, 1, 2, \

2, 5, 5, 5, 1, 2, 6, 3, 3, 4, 1, 1, \

1, 5, 6, 5, 2, 3, 2, 4, 5, 4, 2, 4, \

6, 3, 4, 2, 5, 1, \

//ergänzung ich möchte nun eine andere
//Schreibweise... Ja nämlich 1... 5 sehr
//schwer ab zu tippen ist, besser 15 23...
//noch besser xxx 15 23 xxx...
//würfel ich so weiter + 4 zeilen nächsten
//papier...

62 14, 66 46, 64 55, 54 51, 62 52
24 54, 35 16, 14 42, 54 36, 65 15,
52 31, 66 12, 42 46, 63 52, 12 43,
13 54, 51 34, 64 54, 46 46, 24 56,
56 26, 36 44, 15 42, 21 66, 54 16,
45 66, 22 15, 25 61, 13 33, 55 14,
45 15, 36 52, 65 32, 42 63, 11 44,
55 31, 36 16, 24 51, 45 32, 56 45,
56 43, 56 32, 58 65, 66 44, 51 13,
21 31, 51 44, 55 34, 44 34, 56 56,
45 45, 61 24, 22 62, 13 65, 12 32,

Janet Vajda (c) 07/09/2026

Seite 2, Würfeln...

6 weitere Zeilen würfeln ($5\frac{1}{1} + \frac{1}{2} = 6$) Zeilen

5516, 2554, 1343, 2325, 2633, 5662,
2513, 3526, 5512, 6524, 6661, 1566,
2265, 6346, 6311, 3265, 5332, 6185,
5555, 6611, 4132, 2232, 1122, 6422,
3351, 1653, 5321, 4445, 2312, 1131,
1346, 4561, 6215, 4616, 2363, 41213

ok...

David Hyda (c)

07/09/2026, Seite 1

würfeln...

$E = \{1, 5, 4, 6, 1, 6, 2, 6, 5, 1, 6, 3, 5, 3, \backslash$
 $1, 4, 1, 2, 2, 5, 2, 5, 6, 2, 1, 2, \backslash$
 $2, 5, 5, 5, 1, 2, 6, 3, 3, 4, 1, 1, \backslash$
 $1, 5, 6, 5, 2, 3, 2, 4, 5, 4, 2, 4, \backslash$
 $6, 3, 4, 2, 5, 1, \backslash$

//ergänzung ich möchte nun eine andere
//Schreibweise... Ich nämlich 1... 5 sehr
//schwer ab zu tippen ist, besser 15 23...
//noch besser xxx 15 23 xxx...
//würfel ich so weiter + 4 zeilen nächsten
//papier...

62 14, 66 46, 64 56, 54 51, 62 52
24 54, 35 16, 14 42, 54 36, 65 15,
52 31, 66 12, 42 46, 63 52, 12 43,
13 54, 51 34, 64 54, 46 46, 24 56,
56 26, 36 44, 15 42, 21 66, 54 16,
45 66, 22 15, 25 61, 13 33, 55 14,
45 15, 36 52, 65 32, 42 63, 11 44,
55 31, 36 16, 24 51, 45 32, 56 46,
56 43, 56 32, 58 65, 66 44, 51 13,
21 31, 51 44, 55 34, 44 34, 56 56,
45 45, 61 24, 22 62, 13 65, 12 32,

Lamell Vajda (c) 07/09/2026

Seite 2, Würfeln...

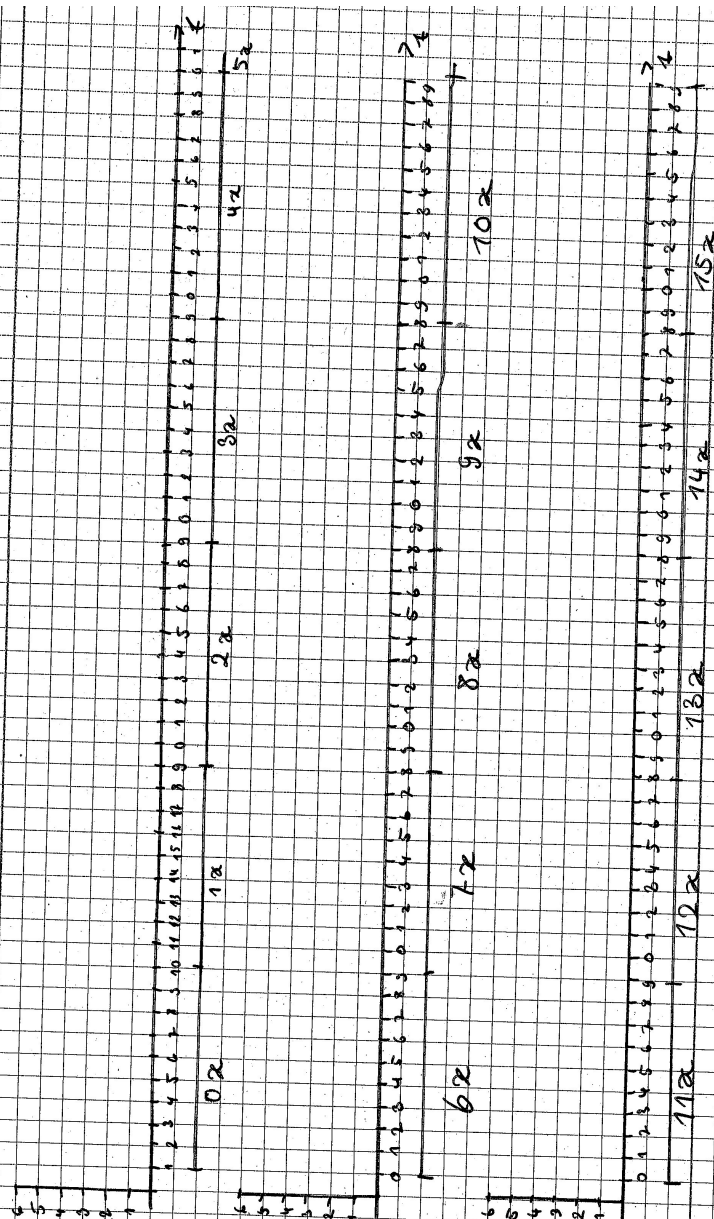
6 weitere Zeilen würfeln ($5\frac{1}{1} + \frac{1}{2} = 6$) Zeilen

5516, 2554, 1343, 2325, 2633, 5662,
2513, 3526, 5512, 6524, 6661, 1566,
2265, 6346, 6311, 3265, 5332, 6165,
5555, 6611, 4132, 2232, 1122, 6422,
3351, 1653, 5321, 4445, 2312, 1131,
1346, 4561, 6215, 4616, 2363, 41213

ch...

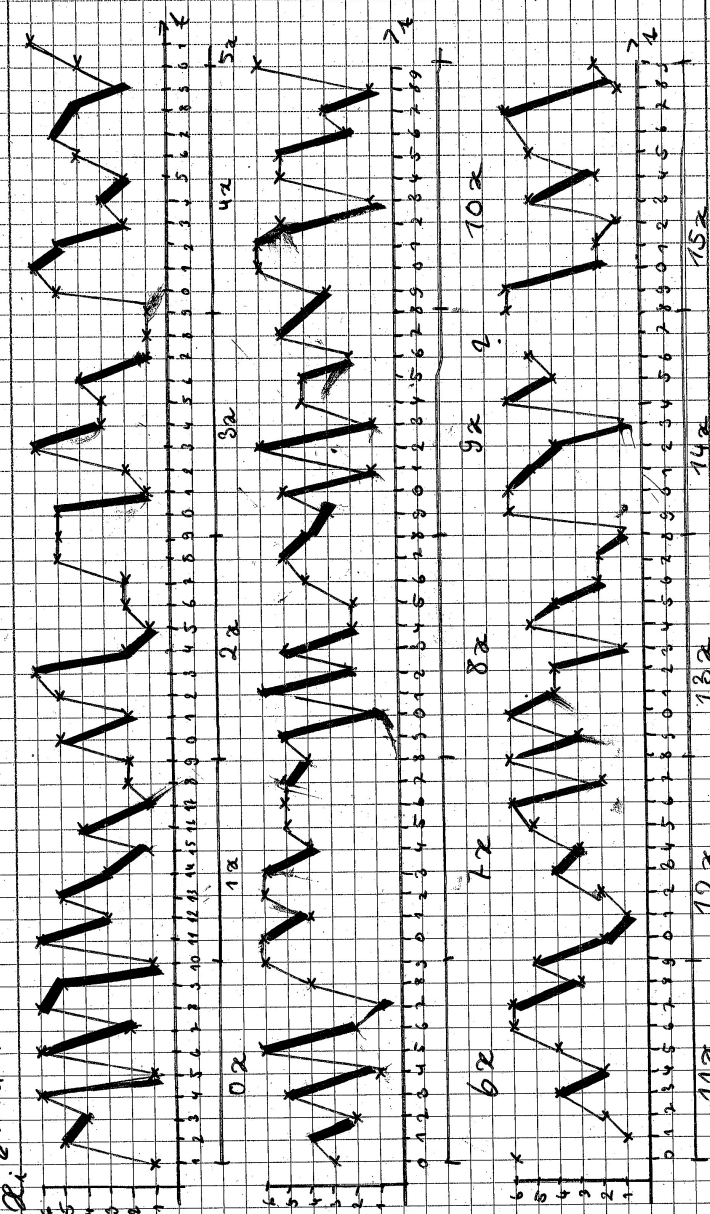
Number... tabella... simul Vajpa Serie 5 07/09/2026

$x_i \leftarrow x(x_i) \cdot x(i)??$



Wintersemester 2020/21 David Vojta Seite 3 OF 109

x_i $x(x_1)$ $x(x_2)$



Zufallszahlen (rand)

07/09/2026

David Vajda (c), Seite 1

D. Lehmer 1951: Zufallszahlen
erzeugen: lineare Kongruenz:

Algo: $a[0] = \text{seed};$
 $\text{for } (i=1; i \leq W; i++)$
 $a[i] = (a[i-1] * b + 1) \% m;$

$\text{seed} = 17$
 $b = 15$
 $m = 30$

gut:

i	$a[i]$	$(a[i-1] * b + 1) \% m$
0	17	x
1		

→ ok
nächste Seite

Zufallszahlen (rand) Saint Hyde

$a[0] = 17$ 07/09/2026, Seite 2

i	$(a[i-1] * b + 1) \% m$	$a[i]$
1	$(17 \cdot 15 + 1) \% 32$	0
2	$(0 \cdot 15 + 1) \% 32$	1
3	$(1 \cdot 15 + 1) \% 32$	16
4	$(16 \cdot 15 + 1) \% 32$	17
5	$(17 \cdot 15 + 1) \% 32$	0
6	$(0 \cdot 15 + 1) \% 32$	1
7	$(1 \cdot 15 + 1) \% 32$	16

Schlechte Zufallszahlenfolge

0, 1, 16, 17, ...

noch ein versuch

primzahl! m muss

primzahl sein wie gewöhnlich

31 wäre ein anplatz

Robert Sedgwick 2er . 10er potenz?

i	$(a[i-1] * b + 1) \% m$	$a[i]$
1	$(17 \cdot 15 + 1) \% 31$	8
2	$(8 \cdot 15 + 1) \% 31$	28
3	$(28 \cdot 15 + 1) \% 31$	18
4	$(18 \cdot 15 + 1) \% 31$	23
5	$(23 \cdot 15 + 1) \% 31$	5
6	$(5 \cdot 15 + 1) \% 31$	14
7	$(14 \cdot 15 + 1) \% 31$	25

Zufallszahlen (Jawid Ujda)
07.10.2026, Seite 3

i	$(a[i-1] \cdot b + 1) \% m$	$a[i]$
8	$(25 \cdot 15 + 1) \% 31$	4
9	$(4 \cdot 15 + 1) \% 31$	30
10	$(30 \cdot 15 + 1) \% 31$	17
11	$(17 \cdot 15 + 1) \% 31$	8
8	ok : repeated ???	

17, 8, 28, 18, 23, 5, 14, 25, 4, 30, 17, 8

$Z_1 = \{17, 8, 28, 18, 23, 5, 14, 25, 4, 30\}$
ok

Stowen's Majstra

07/09/2026

Seite 1

Am-muttschrift

Linbalt u. Lornbalt

beginnen mit einem

Zusatz sein prorsifstent
persönlich o. prorsifstent
unten einen in der prorsifstent

buchstaben mußte nämlich
einen müssen nur & sein

mit & sein in Lin

hott aber einsey f muttschrift

kannte Am-muttschrift for

von prorsifstent... nur f sein in

zusatz ist einsey f. Amozu-

für die Am-muttschrift kann

Novin Majita

07/09/2026

Alte 2 Kinnut-
Krift

mon fuf mfgom
K. Jufmufall

Was y ist auch so ein
interraffantus & Luf-
feln Was ein & ein in
Lufon... Lufon um
Lufon Lufonina...
ob interraffant ist Was
N. von Lufon Kinnutgeln
... aber auch y ein in
Zigornuffmufung o.
Zum Lufon o.
y ein Lufon Gut. v. Majita

Herrn
David Vajda
Marienburger Straße 7
72072 Tübingen

FAKULTÄT FÜR MATHEMATIK UND
INFORMATIK
Prüfungsamt

Ihr Zeichen: 7 828 020
Mein Zeichen: nobe_28E
Auskunft erteilt
C. Dehlinger
Universitätsstr. 1,
Gebäude 3,
58097 Hagen
Fon: +49 (0) 2331 / 987-2598
Fax: +49 (0) 2331 / 987-19240
cornelia.dehlinger@fernuni-hagen.de

29.03.2024

Notenbescheid

Sehr geehrter Herr Vajda,

Sie haben im Bachelorstudiengang Informatik am 21.02.2024 die Modulabschlussprüfung
63013 Computersysteme

im zweiten Versuch bestanden. Die Modulabschlussprüfung wurde mit der Note
"ausreichend" (3,7)

bewertet.

Prüfer/in: Univ.-Prof. Dr. Keller
Prüfer/in: Univ.-Prof. Dr. Oden

Das Modul wird mit 10 ECTS-Punkten gewichtet.



Rechtsbehelfsbelehrung:

Gegen diesen Bescheid können Sie innerhalb eines Monats nach Bekanntgabe Widerspruch erheben. Der Widerspruch ist beim Prüfungsausschuss der Fakultät für Mathematik und Informatik, FernUniversität in Hagen, 58084 Hagen, schriftlich zu erheben oder mündlich zur Niederschrift zu erklären.

Dieser maschinell erstellte Bescheid ist ohne Unterschrift, nicht aber ohne Siegel gültig.

www.fernuni-hagen.de

David Vajda (2)

07/09/2026

Fletcher's Checksum

Übung, Seite 1

$U = (22, \dots)$
per Zufallszahlen in der
Linux bash.

$U = (22, 20, 23, 30, 20, 4, 21, 27, 21,$
 $11, 1, 5)$

Oh. Tabelle:

erst der Algorithmus für
Fletcher's Checksum...

Quelle: Wikipedia

sum 1 := 0
sum 2 := 0

```
for i = 0 to length(U) do
  sum 1 := (sum 1 + U[i]) modulo 255
  sum 2 := (sum 1 + sum 2) modulo 255
endfor
checksum := (sum 1, sum 2)
return checksum.
```

Tabelle

i	U[i]	sum 1	sum 2	sum 1'	sum 2'
0	22	22 (0)	22 (0)	22	22
1	20	22	22	42	64
2	23	42	64	65	129
3	30	64	129	94	223
4	20				
5	4				
6	21				