automaten übung

david vajda

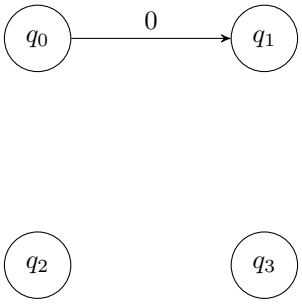
7. September 2026

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4   3   2
3   1   4
2   4   1
1   2   3
```

0.1 mealy automat

andere kodierung folgt

- anders nummeriert
- binär kodiert



ok, was wir hier feststellen, fehlt die ausgabe. auf y, sonst: würde mealy und moore überflüssig. dazu muss eine andere kodierung vorgenommen werden.

1 nächster automat

aus den geraden erwähnten grund hier ein neuer automat:

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 2 , 0 )
3  ( 2 , 0 ) ( 1 , 0 )
2  ( 1 , 0 ) ( 4 , 1 )
1  ( 4 , 1 ) ( 3 , 1 )
```

diesen in einer schöneren tabelle eine möglichkeit der formulierung (meiner meinung nach):

z_n	x ??	$z_{n+1_{x=0}}$	$y_{x=0}$	$z_{n+1_{x=1}}$	$y_{x=1}$
4	$0/1$ (x)	3	1	2	0
3	x	2	0	1	0
2	x	1	0	4	1
1	x	4	1	3	1

1.1 die entstehung dieses automats

zunächst ein mal die frage: wie ist sind diese automaten zu stande gekommen: ich habe ein python3 skript zu grund gelegt. die erste version geht so: die erste version ist diese hier

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach

import random
random.seed ()

V=[1,2,3,4]
V1=[1,2,3,4]
V2=[1,2,3,4]

print ("# (c) david vajda")
print ("# 08/27/2026")
print ("# python3, zustaende, vier zustaende")
print ("# ... graph: schaltwerk, einfach")

i=3
while i >= 1:
 j1 = random.randint (0,i-1)
 j2 = random.randint (0,i-1)
 if i+1 in V1:
  yes1 = True
 else:
```

```
 yes1 = False
if i+1 in V2:
 yes2 = True
else:
 yes2 = False
if yes1:
 V1.remove (i+1)
if yes2:
 V2.remove (i+1)
v1 = V1 [j1]
v2 = V2 [j2]
V1.pop (j1);
V2.pop (j2);
if yes1:
 V1.append (i+1)
if yes2:
 V2.append (i+1)

print (i+1, " ", v1, " ", v2, " ")
i = i - 1

v1 = V1.pop (0);
v2 = V2.pop (0);
print (i+1, " ", v1, " ", v2, " ")
```

erzeugt unter anderem diese ausgabe:

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4   3   2
3   1   4
2   4   1
1   2   3
```

die vorgängerversion vom bisher letzten, der wohl offensichtlich seinen dienst tut ist die

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach

import random
random.seed ()
```

```
V=[1,2,3,4]
V1=[1,2,3,4]
V2=[1,2,3,4]

print ("# (c) david vajda")
print ("# 08/27/2026")
print ("# python3, zustaende, vier zustaende")
print ("# ... graph: schaltwerk, einfach")

i=3
while i >= 1:
 j1 = random.randint (0,i-1)
 j2 = random.randint (0,i-1)
 if i+1 in V1:
  yes1 = True
 else:
  yes1 = False
 if i+1 in V2:
  yes2 = True
 else:
  yes2 = False
 if yes1:
  V1.remove (i+1)
 if yes2:
  V2.remove (i+1)
 v1 = V1 [j1]
 if v1 in V2:
  yes11 = True
 else:
  yes11 = False
 if yes11:
  V2.remove (v1)
 v2 = V2 [j2]
 V1.pop (j1);
 V2.pop (j2);
 if (v1 != v2) and yes11:
  V2.append (v1)
 if yes1:
  V1.append (i+1)
 if yes2:
  V2.append (i+1)

 print (i+1, " ", v1, " ", v2, " ")
 i = i - 1

v1 = V1.pop (0);
```

```
v2 = V2.pop (0);
print (i+1, " ", v1, " ", v2, " ")
```

diese version enthält einen fehler und die ausgabe beim automaten wurden nicht berücksichtigt.

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach

import random
random.seed ()

V=[1,2,3,4]
V1=[1,2,3,4]
V2=[1,2,3,4]

print ("# (c) david vajda")
print ("# 08/27/2026")
print ("# python3, zustaende, vier zustaende")
print ("# ... graph: schaltwerk, einfach")

i=3
while i >= 1:
 j1 = random.randint (0,i-1)
 j2 = random.randint (0,i-1)
 if i+1 in V1:
  yes1 = True
 else:
  yes1 = False
 if i+1 in V2:
  yes2 = True
 else:
  yes2 = False
 if yes1:
  V1.remove (i+1)
 if yes2:
  V2.remove (i+1)
 v1 = V1 [j1]
 if v1 in V2:
  yes11 = True
 else:
  yes11 = False
 if yes11 and (i > 1):
  V2.remove (v1)
```

```
  j2 = random.randint (0,i-2)
 v2 = V2 [j2]
 V1.pop (j1);
 V2.pop (j2);
 if (v1 != v2) and yes11:
  V2.append (v1)
 if yes1:
  V1.append (i+1)
 if yes2:
  V2.append (i+1)

 print (i+1, " (", v1, ",", random.randint(0,1), ") (", v2, ",", random.randint(0,1), ")")
 i = i - 1

v1 = V1.pop (0);
v2 = V2.pop (0);
print (i+1, " (", v1, ",", random.randint(0,1), ") (", v2, ",", random.randint(0,1), ")")
```

dieser automat erzeugt zum beispiel folgende ausgabe:

```
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 0 )
3  ( 1 , 1 ) ( 2 , 0 )
2  ( 4 , 0 ) ( 4 , 1 )
1  ( 3 , 0 ) ( 1 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 1 ) ( 3 , 1 )
3  ( 2 , 1 ) ( 1 , 1 )
2  ( 4 , 0 ) ( 4 , 0 )
1  ( 3 , 1 ) ( 2 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 2 , 1 ) ( 1 , 0 )
3  ( 4 , 0 ) ( 2 , 0 )
2  ( 1 , 0 ) ( 4 , 1 )
```

```
1  ( 3 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 1 , 0 )
3  ( 2 , 0 ) ( 4 , 0 )
2  ( 1 , 1 ) ( 3 , 0 )
1  ( 4 , 0 ) ( 2 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 3 , 0 ) ( 2 , 1 )
3  ( 1 , 1 ) ( 4 , 0 )
2  ( 4 , 0 ) ( 1 , 0 )
1  ( 2 , 0 ) ( 3 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 3 , 0 )
3  ( 4 , 0 ) ( 2 , 1 )
2  ( 3 , 0 ) ( 1 , 0 )
1  ( 2 , 1 ) ( 4 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 2 , 1 )
3  ( 4 , 0 ) ( 1 , 1 )
2  ( 3 , 1 ) ( 4 , 1 )
1  ( 2 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 1 )
3  ( 1 , 1 ) ( 2 , 1 )
2  ( 4 , 1 ) ( 4 , 0 )
1  ( 3 , 1 ) ( 1 , 1 )
```

```
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 1 ) ( 2 , 0 )
3  ( 4 , 0 ) ( 1 , 0 )
2  ( 3 , 0 ) ( 4 , 0 )
1  ( 2 , 0 ) ( 3 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 2 , 0 )
3  ( 4 , 0 ) ( 1 , 1 )
2  ( 3 , 1 ) ( 4 , 1 )
1  ( 2 , 0 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 1 ) ( 2 , 0 )
3  ( 4 , 1 ) ( 1 , 0 )
2  ( 3 , 1 ) ( 4 , 1 )
1  ( 2 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 1 , 0 )
3  ( 1 , 0 ) ( 4 , 0 )
2  ( 4 , 1 ) ( 3 , 1 )
1  ( 3 , 0 ) ( 2 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 3 , 1 )
3  ( 2 , 0 ) ( 1 , 1 )
2  ( 4 , 1 ) ( 4 , 1 )
1  ( 3 , 0 ) ( 2 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
```

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 3 , 1 )
3  ( 2 , 1 ) ( 1 , 0 )
2  ( 4 , 0 ) ( 4 , 0 )
1  ( 3 , 0 ) ( 2 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 0 )
3  ( 1 , 0 ) ( 2 , 0 )
2  ( 4 , 0 ) ( 4 , 0 )
1  ( 3 , 0 ) ( 1 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 2 , 0 )
3  ( 2 , 0 ) ( 1 , 0 )
2  ( 4 , 1 ) ( 4 , 0 )
1  ( 3 , 1 ) ( 3 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 1 )
3  ( 1 , 0 ) ( 2 , 0 )
2  ( 4 , 1 ) ( 4 , 0 )
1  ( 3 , 1 ) ( 1 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 1 , 1 )
3  ( 1 , 1 ) ( 4 , 1 )
2  ( 4 , 1 ) ( 3 , 0 )
1  ( 2 , 1 ) ( 2 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
```

```
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 2 , 0 )
3  ( 2 , 0 ) ( 1 , 0 )
2  ( 1 , 0 ) ( 4 , 1 )
1  ( 4 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 2 , 1 )
3  ( 2 , 1 ) ( 1 , 1 )
2  ( 1 , 0 ) ( 4 , 1 )
1  ( 4 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 0 )
3  ( 1 , 0 ) ( 2 , 0 )
2  ( 4 , 0 ) ( 4 , 1 )
1  ( 3 , 1 ) ( 1 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# ... graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 1 , 1 )
3  ( 4 , 1 ) ( 2 , 0 )
2  ( 1 , 1 ) ( 4 , 1 )
1  ( 3 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ %
```

was haben wir eigentlich getan. in der fernuni-hagen hatte ich bei

- ungerer
- keller
- schiffmann
- bähring

wohl noch weitere. für computersysteme 1.
und prüfung abgenommen, cornelia dallinger, computersysteme 1/2. hier werden, in computersysteme 1, die zustandsgraphen die automaten beschreiben zum thema (ausdruck). in algorithmischer mathematik wird die graphentheorie zum inhalt gemacht.

als ich auf die prüfung computersysteme 1/2 lernte, simulierte ich so mehr schlecht als recht automaten, um es zu üben, um so die prüfung mit einer 3.7 zu bestehen. ich weise sowohl auf die bestandene prüfung hin. zwar sind das benotete prüfungen was an der fernuni-hagen nicht die einzigen sind, sondern für den ba. sind sie mehr oder weniger pflicht und nur studenten mit abitur sind zu gelassen, ob sie sich nun den B.A. holen oder nicht. die unbenoteten prüfungen sind nicht immer einfach, nur: in diesem falle ist computersysteme 1/2 so wollte ich schon sagen, wohl das kalkül der technischen informatik überhaupt. die informatik besteht wohl aus drei säulen: technische, praktische und theoretische. ich denke aber die technische ist am ende die entscheidenste (a) - trotz allem - ist ansichtssache, aus meiner sicht. nicht nur: es gibt vielleicht verschiedene formen technische informatik zu prüfen, ich denke die einzig gültige ist am ende computersysteme 1/2. es ist eher das gesamte kalkül. wenn man an den automaten denkt, indem der computer mit mikroprogrammspeicherwerk, das programm in einer nicht veränderten version, deswegen doppelt so gross, wegen der verzweigung, wie in den eingescanten per hand geschriebenen dokumenten steht: dann ist technische informatik. . . . ein sehr starkes thema. ich möchte mich kurz fassen.

ich bin zu frieden. ich selber gedenke noch ein paar prüfungen zu tun, Bachelor hole ich mir eher nicht. etwas das ich versuchen werde ist wohl: algorithmische mathematik

und hier werden eben graphen beschrieben. das ist so oder so nicht irrelevant. aus folgendem grund:

in algorithmischer mathematik werden graphen vorgestellt. dabei

1. tiefensuche
2. breitensuche
3. eulertour

die eulertour ist sehr interessant, gucken wir in meine homepage. sie bezieht sich auf graphen, die keine bäume sind. tiefen und breitensuche finden einen baum. in jedem graphen, sie verbinden, besonders mit stapel und schlangen, algorithmisch alle möglichen knoten im graph.

über das üben dieser habe ich gelernt die eulertour in python3 um zu setzen. wie hier, mit einer art, schlange oder stapel, jedenfalls elemente herausnehmen,

erreichen wir bei einem graphen unkompliziert, dass mehr oder weniger: elemente mehr als nötig kanten zu nachbarn habe, dafür knoten ausgelassen werden, unzusammenhängende graphen, laut algorithmischer mathematik.

ok: ich möchte an dieser stelle auf unterschiede und gemeinsamkeiten zu meinen bisherigen übungen verweisen.

zunächst bevor ich diese graphentheorie kannte, hatte ich bei den nachbarn das übliche problem. am ende habe ich mehr oder weniger zufallszahlen genommen, durch umsortieren solange, bis es passt. das ist keine besonders gute lösung in der graphentheorie selber, aber nicht untersucht, (beispiel: permutation - umbennenung von knoten, wobei der name eines knotens generell keine rolle spielt - und ich weise darauf hin - da ich arabisch ebenso übe - und auch hieroglyphen hier auch frei schreiben will, aber beschlossen habe nicht mehr handschriftlich zu schreiben, werde ich vielleicht nicht unbedingt aber vielleicht auch diesen graphen hier mit arabischen ziffern ausstatten, z.B. Hieroglyphen, stuhl und tisch? oder eben binärkodiert).

ich möchte auf den unterschied der darstellung hinweisen, bezüglich des „Informatischen“inhalts. bzw. der darstellung in der technischen informatik.

hier sind automaten, das wurde bei meinen bisherigen übungen zunächst vernachlässigt, ich weise auf unterschiede und gemeinsamkeiten hin

1. unär
2. binär

kodiert. unär bedeutet, zunächst habe ich computersysteme 1/2 auswendig gelernt

```
von null beginnend, un"ar kodiert, konsekutiv nummeriert
```

wichtig ist, dass gegenbeispiel IP-Addresse
es taucht oft die verwirrung auf, jedenfalls bei mir damals am technischem gymnasium: gibt es, wenn man diese verwirrung zu wort bringt, IP-Addressen der form

```
1011 1101.0011 1010.0000 0000.0000 0000
```

nein es gibt nur:

```
...
1111 1111.1111 11111.1111 0000.0000 0000
...
1111 1111.0000 0000.0000 0000.0000 0000
1111 1110.0000 0000.0000 0000.0000 0000
```

```
1111 1100.0000 0000.0000 0000.0000 0000
..
0000 0000.0000 0000.0000 0000.0000 0000
```

das heisst, die msb 1er bit werden voll aufgefüllt. ein passendes beispiel von codes ist 2 aus 5. hier gibt es 2 einer bit in 5. dabei rueckt das lsb immer eines vor, bis die immer nach rechts (lsb) weiter vorrueckende grenze erreicht ist. inzwischen rueckt das msb 1er bit vor. darauf hin das lsb 1er bit zu der grenze die darauf nach rechts vorrueckt, waehrend das msb 1er bit wieder auf den anfang zurueck geht.

das entspricht so einem vorgang, als beispiel einer schiften, oder ip-addresse, aber eben zerfaellt die sache hier.

ein anderes beispiel ist die unäre kodierung. striche an die wand machen, beim aller einfachsten zaehlen, bedeutet ip-addresse. denken wir an: dekodierer, ich weise darauf hin, es gibt:

1. kodierer
2. dekodierer
3. multiplexer
4. demultiplexer

ein dekodierer liesse sich noch einigermassen mit einem demultiplexer erreichen, wuerde ich aus dem kopf sagen. ich weise aber auf eindeutigen unterschiede hin.

ein dekodierer ist ein kodierer.

ein multiplexer ist quasi ein element, was serialisiert - als multiplexed. (rs232 - v.24) - es legt ein 8 bit breites byte mit einem takt - zeitlich auf eine leitung. das gegenteil ist der demultiplexer. ein dekodierer ist immer noch ein kodierer (DIN 44300/183) (siehe auch: DIN 40700, DIN 40900) ist ein schaltnetz, das die zeichen eines zeichenvorrats in die zeichen eines andereren zeichenvorrats umsetzt (Computersysteme 1/2). ok: blos: ein zeichen in der informatik ist binär kodiert (siehe BCD: ausser am Sichtgeraet: siehe BCD und Abakus, warum: die perlen im Abakus, sind in einer zeile das was eine ziffer, arabisch und indisch ist (oder chinesisch), und die binärkodierung am LCD HD 44780, kann eben Chinesisch, arabisch, hebraeisch, inka, maja, azteken, brahmanische schriften, latein und so weiter sein). ok: BCD, entspricht trotzdem einer möglichen anderer kodierung der Arabischen bzw. indischen ziffern.

ok: ein dekodierer wiederum ist ein kodierer. das heisst, der zielzeichenvorrat ist von der wortbreite wohl meist ein weniger kleiner oder manchmal vielleicht ein weniger groesser, oder gleich, eher gleich oder kleiner, wie die in der quelle. ausgangsmenge. wie auch immer. der dekodierer ist ein kodierer. und zwar:

einer bei dem für jedes zeichen in der ausgangsmenge immer nur eine leitung gleichzeitig an ist.

ok: man kann also hier sehen: unär kodierung entspricht jetzt eher dem, das bit wandert, wie geschiftet.

typischer befehl: SLL, SLR - shift logical left/right. wenn ein bit im quelloperanden, 1 ist dann prima. ok:

andere unterschiede:

1. würde ich von der binären kodierung ausgehen, was laut computersysteme 1/2 möglich ist, dann: gibt es einen vorteil, die bitbreite ($lg(ceil(n))$ - oder floor. da ich hier romane schreibe wie also auf dem papier, sage ich hier ehrlich was sache ist. es gibt den 2er, 10er und e logarithmus, das erkläre ich hier nicht. der 10er logarithmus ist wohl ld und der 2er logarithmus wohl: lg wobei ich nachschauen könnte, jetzt aber verzichte, eine andere schreibweise und die ist immer möglich: (log_2), also die basis unten angeben, das ist das üblichste, ich glaube, das d steht fuer dezimal. und ich glaube, bei dem Dezibel wird $ld\left(\frac{P_{in}}{P_{out}}\right)$ benutzt, eingangs und ausgangsleistung, laesst sich wohl auch umdrehen, jedenfalls, dezibel, im gegensatz, zum bel und ich glaube da steht, ld. 10er. so: ansonsten basis unten angeben. wo ceil und wo floor liegt weiss ich nie.

   jedenfalls: man muss aufrunden und den 2er logarithmus auf die anzahl der knoten anlegen, und wenn ich das habe. dann erhaelte ich die bitbreite. ok:

   etwas, das die sache nun klassisch verändern würde ist folgendes: brauche ich beim binären einen addierer? habe ich damit alles falsch gemacht?

   (a) ich sage: ja und nein, nein deswegen weil es schaltet, trotzdem krotten falsch, warum?

   (b) weil wir in computersysteme 1 und 2 gelernt haben: übergangsschaltnetz und ausgangsschaltnetz.

   (c) die zustände selber sollten einfach fluppen, ok: hier ist es alles andere als krotten falsch, warum? weil ich mit addieren durch die graphentheorie nicht weit komme, eben dann sind graphen nicht einfach durch permutation eigentlich der gleiche, sondern wenn ihre kanten sich massgeblich unterscheiden. ok: addition hilft hier nichts

   (d) die aeussere darstellung ist aufgabe des uebergangsschaltnetz. und leider trotzdem falsch.

   (e) warum? eben deswegen nehme ich hier: unär und eben deswegen: mealy und moore und ich nehme unär

   (f) denn: ein unäres schaltwerk von den zuständen einfach durch schiften. über das übergangsschaltnetz was nun kein addierer ist, einfach dann

mit der eingabe die naechsten zustand erzeugen. falsch ist: mealy erfordert: bei der ausgabe wird die eingabe mitgenommen

(g) doch damit ist die sache nicht falsch. denn erstens ist der addierer ein schaltnetz und zweitens: auch beim unären entstehen hier variationen...

ok: wo weit so gut, nur:

es gibt noch hinsichtlich der tabelle alternative darstellungen. was in der mathematik üblich wäre.

das habe ich mir im smartphone auf facebook gemerkt:

```
a war so noch ein Fehler, der ist behoben das war klar, der ist so noch behoben plus, dass i

Das schreib ich dir noch rein in das PDF, was ich mache, wenn wir eine Null oder eins zu den

Daraus ein Baum machen, aber nur bedingt, weil das ist dann ein pattern  Matching

Was ich damit endlich beschreiben kann, was nichts mit KMP zu tun hat, sondern halten Automa

Das Problem bei dem Baum ist ja ich kann jetzt von einem Knoten aus dem Baum beschreiben fr

Schreibe ich ins PDF rein wie gesagt aber auch noch ein Fehler

Den hab ich behoben blo der Unterschied ist

Am was bei Algorithmen in C steht

Robert sedgewick
Heit das der sozusagen der Compiler fr die formale Sprache an der Stelle, die sich auch in b
```

ich bin dahinter gekommen, dass: und das ist mir wichtig, dass eigentlich: ein mikroprogrammsteuerwerk genauso harter speicher wie bei der havard architektur sein könnte wie: diese selber, aber einen doppelten speicher braucht, wegen der möglichen verzweigung oder die addresse ist im wort (konstante, immidiate) im wort fuer das operationswerk vorhanden, oder andere version: weil es nicht immer nur bedingungen gibt, ein kleiner speicher hinten angehaengt.

ok. das ist die eine erkenntnis, wie ich sie in den papieren habe, die hier anhaenge. das andere ist aber: das bezog sich nicht darauf.

1.2 tabelle

z_n	x ??	$z_{n+1_{x=0}}$	$y_{x=0}$	$z_{n+1_{x=1}}$	$y_{x=1}$
4	0/1 (x)	3	1	2	0
3	x	2	0	1	0
2	x	1	0	4	1
1	x	4	1	3	1

nun zu erst, die alternative darstellung der tabelle

hier wie gerade eben.

ich weise noch darauf hin, noch mal, ..., ich mache um papier zu sparen, aber um mit TikZ von LaTeXimmer gleich schaltpläne ieee-91/1984, automatendiagramme, zu tun... immer gleich alles mit LaTeX. durchaus mit kaneda. wobei LaTeXund TikZ ist zwang. es geht darum, dass ich einen grossrechner baue, (a) der form uniac, eniac usw. dies müsste mit transistoren machbar sein. über legierung lassen sich vielleicht, ich sage vielleicht, in einem hochofen der marke vevor oder alternative von denen ich destilliergeräte habe, für einen relativ erschwinglichen preiss, etwa 700 Euro aufwärts vielleicht legierungen herstellen: verhältnis z.B. 80:20. dies liesse sich eher mit gallium, als germanium tun z.B. einige haben komplizierte kühlverfahren mit bor oder brohm, sagt sich schwierig, schon der begriff dunkelkammer ist in der chemie oft keine schwierigkeit, legierungen lassen sich, vermutlich relativ einfach machen, mein onkel mütterlicherseits war goldschmied, silber in den ringen kaufen 825. 925 z.B. und legierung beschrieben in elektrotechnik gelb, autor jetzt nicht auswendig im gegensatz zum transistor, sage ich ja: müssen wir machen und sind wie wir sind. ok: köennte machbar sein, dann mit P und N und halbleiter auf halbleiter wie elektrotechnik kontakt an kontakt, will bild ersten transistors auf wikipedia gesehen haben, aber: jetzt ALU bestellt, mükra, spott günstig, damit lässt sich mit speicher und regsiter einfach cpu machen, dann transistor, habe teile eines univacs auf ebay gefunden, platine mit transistor. man könnte einfach: dann modul für modul. dafür aufbau von logikgatter mit transistor üben. univac ähnliches ding von mir: gekaufter transistor, dann später oder modul für modul, oder teilweise, eigene transitoren, wäre auch eine möglichkeit: gekaufte plus module mit eigenen... ok: deswegen auch mit kaneda, schaltpläne mit transistor üben.

jetzt die tabelle anders gestellt. was sind noch aufgaben? die transformation von mealy und moore

ich weise noch darauf hin, dass ich nicht vorhabe das univac ähnliche ding, dann als endziel erreicht zu haben und weiter nichts zu tun, die übungen sind standard.

ok: und die transformation ist wichtig und dann noch folgendes: ich konstruiere das als schaltnetz und so weiter, als plan, aber wenn erreicht, noch: wie gesagt: und zwar unär dies mal, und dazu noch: dass man das wort des automaten nimmt, mit ausgabe $y = 0$ oder 1 und, dass man: dadurch auch Pattern Matching erreicht. ok.

wenn das so weit ist, muss ich noch vorstellen, was ich LaTeXüber TikZ weiss, das ist hier auch noch thema, damit die zeichnung so schön tut, das ist mit teil der übung

erst trotzdem die tabelle:

eine andere tabelle hat doppelt so viele zeilen: weil man ein mal von $x = 0$ und ein mal von $x = 1$ ausgeht.

z_n	x ??	$z_{n+1_{x=0}}$	$y_{x=0}$	$z_{n+1_{x=1}}$	$y_{x=1}$
4	0/1 (x)	3	1	2	0
3	x	2	0	1	0
2	x	1	0	4	1
1	x	4	1	3	1

hier:

z_n	x	z_{n+1}	y
4	0	3	1
4	1	2	0
3	0	2	0
3	1	1	0
2	0	1	0
2	1	4	1
1	0	4	1
1	1	3	1

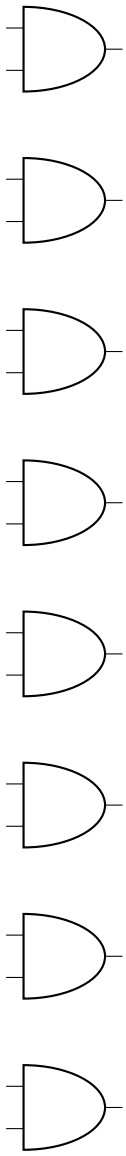
hier noch mal, weil es so schön ist, und papier mit schönheit zusammenhängt.

ok, dann geht es um die kodierung der zustände. während, $4, 3, 2, 1$ ja eine möglichkeit wäre als menge V schreiben, V Vertices (Knoten) und: müsste man hier bei 0 beginnen und die übersetzung wäre unär und binär kodiert:

das ist mealy:

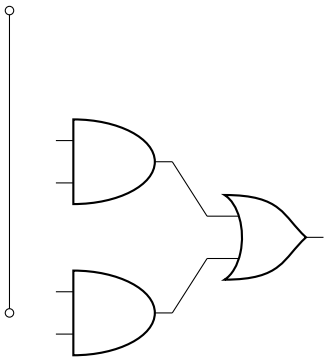
z_n	x	z_{n+1}	y
4	0	3	1
4	1	2	0
3	0	2	0
3	1	1	0
2	0	1	0
2	1	4	1
1	0	4	1
1	1	3	1

jetzt kommen wir zum thema: tikz. der unterschied, das hier ist ein übungsskript. das heisst, ich mache jetzt, das was ich sonst tue, ich lerne es nicht laut auswendig aber schreibe es wiederholt hin, warum? damit es im kopf sitzen bleibt. ich probiere zwar, lange rede kurzer sinn, hier die übung für automaten zu machen, die übung beinhaltet TikZ. und TikZ enthält Knoten für AND OR und Knoten. im Graphen. so, damit schreibe ich die wiederholt, weil das ist teil der übung. zunächst die TikZ darstellungen von denen ich ausgehe:



2 exersize

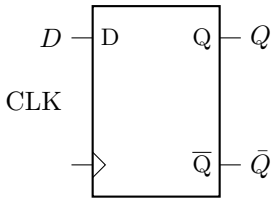
- Make a lot of and from top to bottom, make a DNF

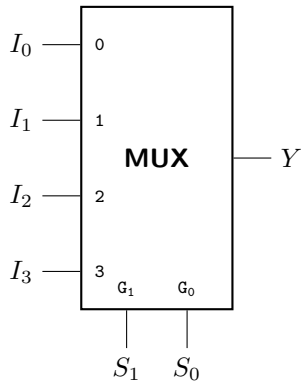


das sieht als code so aus.

```
\begin{circuitikz}
 \node [and port] (and00) at (0,0) {};
 \node [and port] (and01) at (0,2) {};
 \node [and port] (and02) at (0,4) {};
 \node [and port] (and03) at (0,6) {};
 \node [and port] (and04) at (0,8) {};
 \node [and port] (and05) at (0,10) {};
 \node [and port] (and06) at (0,12) {};
 \node [and port] (and07) at (0,14) {};
\end{circuitikz}
\section{exersize}
\begin{itemize}
\item Make a lot of and from top to bottom, make a DNF
\end{itemize}
\begin{circuitikz}
 \draw (0,0) to[short, o-o] (0,4);
 \node [and port] (and00) at (2,0) {};
 \node [and port] (and01) at (2,2) {};
 \node [or port] (or00) at (4,1) {};
 \draw (and00.out) -- (or00.in 2);
 \draw (and01.out) -- (or00.in 1);
\end{circuitikz}
```







das sieht als code so aus:

```
\begin{circuitikz}
% Configure global block options for a rectangular look
    \ctikzset{blocks/scale=1.5, blocks/fill=gray!10}

    % Draw the main multiplexer body as a rectangular block
    \node [draw, minimum width=2cm, minimum height=4cm, thick] (mux) at (0,0) {};

    % IEEE 91-1984 Center functional label
    \node [anchor=center, font=\sffamily\bfseries] at (mux.center) {MUX};

    % Inputs (Left Side) with IEEE dependency tagging (G for control, numbers for channels)
    \draw (mux.west) ++(0, 1.5)  coordinate (in0) -- ++(-0.5,0) node[left] {$I_0$};
    \draw (mux.west) ++(0, 0.5)  coordinate (in1) -- ++(-0.5,0) node[left] {$I_1$};
    \draw (mux.west) ++(0, -0.5) coordinate (in2) -- ++(-0.5,0) node[left] {$I_2$};
    \draw (mux.west) ++(0, -1.5) coordinate (in3) -- ++(-0.5,0) node[left] {$I_3$};

    % Interior labels for internal dependency paths
    \node [anchor=west, font=\scriptsize\ttfamily] at (mux.west) [yshift=1.5cm, xshift=2pt]
    \node [anchor=west, font=\scriptsize\ttfamily] at (mux.west) [yshift=0.5cm, xshift=2pt]
    \node [anchor=west, font=\scriptsize\ttfamily] at (mux.west) [yshift=-0.5cm, xshift=2pt]
    \node [anchor=west, font=\scriptsize\ttfamily] at (mux.west) [yshift=-1.5cm, xshift=2pt]

    % Control Select Lines (Bottom Side) acting as G-dependency inputs
    \draw (mux.south) ++(-0.4, 0) coordinate (s1) -- ++(0,-0.5) node[below] {$S_1$};
    \draw (mux.south) ++(0.4, 0)  coordinate (s0) -- ++(0,-0.5) node[below] {$S_0$};

    % Internal control mapping text (G stands for AND-dependency under IEEE-91)
    \node [anchor=south, font=\scriptsize\ttfamily] at (mux.south) [xshift=-0.4cm, yshift=2p
    \node [anchor=south, font=\scriptsize\ttfamily] at (mux.south) [xshift=0.4cm, yshift=2pt

    % Output (Right Side)
    \draw (mux.east) -- ++(0.5,0) node[right] {$Y$};
    \end{circuitikz}
```

und jetzt vieles üben wird durch das ständige hinschreiben von AND PORT usw. getan. und diese beispiele hier sind übrigens, mit google AI/KI entstanden. das heisst, es sind klassische beispiele für TikZ, aber CircuiTikZ. und: sie sind mit google ki entstanden, ich übe dann ein paar mal das aufschreiben von Port And und so weiter, dann kommt der automat, dann mealy wie hier, dann moore, dann als schaltplan. und dann:

die ausgabewörter. wenn es so weit kommt, sage ich mal: dann vielleicht noch mit gattern in der schaltung realisieren, das wort auf dem oszilloskop angucken.

halt falsch! mir kam dann wieder in den sinn es auch mit abel zu machen! ich sage falsch! falsch! falsch!

ich habe ja gesagt: ich wollte es nur noch schriftlich machen, ohne das bemühen der bauteile. weil die bauteile kosten geld, sollen da sein, sollen nicht so sehr benutzt werden.

dasselbe gilt für abel und den automat. ABEL - HDL - Hardware description language. genauso wie auswendig wissen, wie geht GAL - Generic Array Logic, kein Gate Array Logic. OK: trotzdem: das sollte auf dem papier geschehen. auf dem papier habe ich festgestellt, dass das papier knapp wird und dass ich eigentlich TikZ brauche, das heisst, ich nehme immer noch keine schaltung. ich habe festgestellt, dass ich codes genauso auf papier schreiben kann, was atmega8 betrifft, aber auch galasm, wie ABEL, ABEL und galasm wie palasm, aber bei linux galasm, selbe syntax: eine sprache HDL - jedenfalls: auch das kann ich auf papier machen, da ich auf papier verzichte am computer, aber das muss ja nicht in die hardware rein. schon hier stellt sich die frage nach einem simulator. ich nehme einen simulator. ich benutze auch hardware, nur jetzt nicht, wichtig ist, den quelltext der auch schon geübt wurde, über den keine unklarheit besteht, immer wieder aufschreiben, der simulator ist für mich keine pflicht, es stellt sich an der stelle die frage, ob man das benutzt.

also: auf jedenfall: keine hardware ständig, dann: papier, aber eben nicht papier, doch computer und vielleicht simulator.

ok: jetzt das üben des TikZ. und wenn ich das habe, mealy und moore. und wenn das, das wort und dann noch als schaltung in IEEE-91/1984. ok, mit gattern und so weiter.. und dann kommen hier noch die papiere vom komplexen schaltwerk rein, die einzige übung mit LaTeXist dann maximal das ASM-Diagramm, das muss geübt werden und: sonst papiere rein und: später noch: das Quine, dazu schaltplan.... und ein balkendiagramm, aber ohne: oszilloskop benutzen.

3 Erwähnung zwischendrin

Das schreibe ich nachher ins Skript rein. Ich geb dann noch mal zu. Ich hab von meiner Schwester malen Modellbaudetektor Empfänger bekommen.

Sag dann aber auch die ganze Wahrheit. Ein Klassenkamerad von mir. Damals hat über meinen Computer hergezogen das ist der IBM 5140 der jetzt zum Glück zu mir kommt vielleicht zurecht, weil die Firma sammelt die wohl alle ein und es kann sein, dass sie das dann bei mir ist ja gut aufgehoben.

Klassenkamerad fing damals mit 32 Bit an und 486 und 386 Intel daraufhin habe ich Stress geschoben und das schöne Ding kaputt gemacht. Jetzt kommt es zum Glück wieder. Bei mir ist es gut aufgehoben.

Dann gebe ich zu, dass ich einen Modellbaukasten von meiner Schwester mit einem Diodenempfänger hatte. Natürlich ist die Diode eine einfache AA Diode und gerade das macht auch der Schöne aus. Es war kein alter Drehkondensator. Ich schreib das hier nachher noch ins Skript rein. ich kann es ja genauso übernehmen.

Stellt sich die Frage, muss ich das noch mal schreiben? Nein ich hab vorher ja schon was rein kopiert

Auf die Frage, ob ich Steckbretter nehme ja ich löte nicht

Das ist ja sowieso nachher in das Script ver wurschtelt wird ist es egal

Ich löte nicht ich hab zwar ein Lötkolben ich hatte auch ein besseres ich kann als Wiedergutmachung für die zwei Transistoren bei meinem Kollegen sagen. Ich hab ja mein Zeug an die Strae gestellt es wurde benutzt, aber es hat 100 prozent funktioniert. Besonders das Oszilloskop.

Zwei stk500 jedenfalls mein Kollege, der Johannes fast hatte damals. Ich hoffe nicht, dass der dies ist, der mich hier auf Mikrocontroller. Netz angemacht hat. Es war trotzdem bisschen seltsam, und ich glaube, dass Frau Baerbock, die hat die Gruppe hier so im Griff, aber im Blick glaube ich, und es ist keine Durcheinander. Theorie von Psychologen, Herr Psychologie. Sie müssen sich da nicht vor allen Dingen sie runterschrauben und auch die Theorien von manchen

Leuten das technische informatiktechnische Informatik ist. Es stimmt so nicht.

Und wie gesagt, wenn es bis nach zu Frau Thunberg reicht wegen zwei Transistoren aus einem Modellbau, der Johannes fast und sag ich mal ich bin trotzdem auf das tg

Die hass Mentalität dieser Gruppe dieser Gruppen, teilweise ist bezeichnet sich auch in klügeren Schritt überlegen können wir kommen wir den Ant Transistor. Stattdessen kommt ein Akte. überheblichkeit wie auch Frau Baerbock, die immer irgendwas hin will, guckt er wohl hierhin ich hab ein Graffiti hier an der Wand gefunden. Da steht Baerbock drauf, das stand wohl schon länger drauf. Was auch immer ist ich wei es nicht. Ich sag jetzt folgendes.

Also ob die damit wei ich nicht aber diese ganzen Gruppierung gucken auf jedenfalls meine Schwester hatte mir ein Detektor Empfänger mit AA Diode gekauft. Glas

Und ich wusste, wie es geht. Leider hab ich den auch nicht mehr zum Thema Steckbrett, sag ich folgendes

Ich wei nicht, was der Johannes meint und ob der das auf Mikrocontroller.net ist. Ich sag nur mal folgendes folgendes also tatsächlich hatte der damals auch einen Baukasten und dafür entschuldige ich mich. Ich habe dafür stk 500 an die Strae gestellt, die ich übrigens von dem Geld meiner Mutter gekauft hatte, die ich von dem Geld meiner Mutter gekauft hatte, als ich bei ihr gewohnt hat deshalb wahrscheinlich ein Nachbar an sich genommen und er hat es zurecht. Bitte nicht kaputt machen weiter benutzen das ist ihr Zeug. Ich verfüge darüber nicht. Wahrscheinlich waren sie das das ist ihr Zeug. Ich verfüge darüber nicht. Meine meisten Computer wurden als Geburtstagsgeschenke von meinen Eltern gemacht. Plus übrigens, diesen Dingern, die hab ich bei meiner Mutter gekauft und Oszilloskop damals günstiger an damit denke ich hoffe ich mal und das sage ich jetzt mal unkompliziert. Es gibt Leute, die machen immer komplizierter. Er ist wieder Gutmachung für die zwei Transistoren. Das war auch ein Baukasten. Dabei waren Plan. Wie funktioniert ein Transistor mit Schleuse das hatte der als Plan. Das hatte der als Plan und zwei Transistoren aus dem Baukasten. Diese zwei Transistoren hab ich wahrscheinlich. Entweder ich hab sie ihm sogar wiedergegeben. Den Plan hab ich ihm auf jeden Fall wiedergegeben. Transistoren konnte man schon damals viel kriegen, deswegen hab ich daran nicht gedacht. Tut mir trotzdem leid man sieht wie wichtig die einzelnen Bauteile sind. Man sieht, wie wichtig die einzelnen Bauteile sind und tut mir trotzdem leid. Im Plan hat er wieder gekriegt. Das andere Problem war dann trotzdem

Wie gesagt also jetzt zu den Steckbrettern zu den Steckbrettern sag ich folgendes wie man sieht kann man das ohne löten machen wenn ich jetzt zum Beispiel ein Univac

Oder wie das Ding damals ist es ist vielleicht auch noch das Auto da kannst du auch ne doppelte Verwechslung geben kann geben es gibt Unix Reifen in Ungarn gibt es Unix Reifen die gibt es hier als Betriebssystem gut ich wei nicht ob die wirklich so Eniac ja, ich glaub schon, d.h. nicht, dass ich mich jetzt wieder von angeblich linken Mädchen Gruppierung oder wenn auch nicht anmachen lasse, dass sie mir sagen ich hätte keine Ahnung von dem Computer. Ich wei, wie tut, man kann es mir Transistoren machen aber man sieht hier ohne den zu bauen bin jetzt zum Beispiel die ALU kommt

Ich glaube übrigens, dass dieser Johannes fast nicht ganz so böse ist, sondern dass das, was mir im Microcontroller Netz ich wei es nicht. Ich entschuldige mich dafür deine Paste schon mal gut machen Sie es nicht komplizierter. Ich wei, dass man ich war trotzdem am technischen hat damit aber nichts zu tun und es ist hier nicht immer Teil Streit für ein Thema des Streitthema. Was tun, wenn man jetzt zum Beispiel diese alten Computer baut die Gro Rechenmaschine

Ja, das sind maximal bei dem Ding zwei Transistoren abhandengekommen die konnte man allerdings auch so kaufen. Ist trotzdem nicht nett plus den Plan hat er wieder gekriegt wie Transistoren tun wenn man jetzt das Ding nimmt also den Computer als Grorechner baut, dann kann man Transistor nehmen Der erste Schritt wird es mit der ALU zu machen, so wie das jetzt ist.

So wie das jetzt ist, kann man das so machen aber man kann auch einzelne Transistor nehmen. Man kann die übereinander stapeln. Jetzt kann man natürlich da kann man immer noch Teile löten. Jetzt kann ich die schlecht auf Streifenplatinenlöten schlecht auf Streifen oder ich kann dir sogar noch ätzen, wenn ich das aber modular mache und die einzelnen einstecke. Und es ist ja üblich bei Computern das war auch bei dem IBM PC. Und ist sie noch immer so. Sie können die Teile zusammenstecken erst mal, Sie können das IC ein und aussteigen. Es gibt kein Streit, dass die Dinge, die ich an die Strae gestellt hab, dass ich die nicht neu habe. Ich bin zufrieden. Ich hab alles neu und das andere Programmiergerätes übrigens ein günstigeres ich hab alles neu da muss man sich keine Sorgen machen. So, jetzt kann man, wenn sie darauf achten, die die Platinen teilweise zusammenstecken. Daneben gibt es die üblichen buskabel . Man kann aber auch die IC einzeln einstecken. Gut wenn ich jetzt ne Maschine Bau dann ist die kein wenig gegenüber im Nachteil wenn ich dann hinkriege,

Transistoren zu machen der alten gegenüber. Wenn ich dann hinkriege, Transistoren zu machen, kann ich dir auch mit der Hand machen. Also mit dem Hochofen vielleicht wenn wir das gelingt und dann kann ich die einzelnen reinstecken, dann kann ich auch Streifen Raster Platin nehmen ätzen. Ich kann aber genauso gut in anderen Experimenten wenn ich zum Beispiel so eine nehmen also ne ALU dann kann ich dir ne Steckbrett reinstecken dann kann ich dir den Steckbrett reinstecken also in Steckbrett und wenn ich das habe, dann kann ich dir so lassen. Also anders als bei so einer Gro Rechenmaschine wo ich das so reinsetzen und aber sozusagen übereinander gestapelt und dann mit Kabeln also übereinander gestapelt und dann kann ich in dem Stapel einzelne Komponenten ersetzen. In der steckt Brett kann ich das aber sitzen lassen deswegen sind auch tatsächlich diese Experimentierekästen nicht schlecht und die kann man auch heil und ganz lassen übrigens und die AA ist auch interessant. Trotzdem das geht jetzt auch so sag ich mal und das heit auch nicht. Also wie gesagt AA Diode das kriege ich noch hin die kann ich. Ich kann den auch noch mal kaufen trotzdem das ist jetzt soweit so und dann kann man die Steckbretter so lassen. Danke schön

4 graphen

die übung mit LaTeX. ich habe ja gesagt, dass das eine Übung ist, durchaus für die Darstellung. ich mache folgendes: ich übe - einfach die knoten, knoten für knoten, wie vorher gesagt.

```
\begin{center}
\begin{tikzpicture}[->, >=stealth', node distance=3cm, auto]
  % Zustnde definieren
  \node[state] (q0) {$q_0$};
  \node[state, right of=q0] (q1) {$q_1$};
  \node[state, below of=q0] (q2) {$q_2$};
  \node[state, right of=q2] (q3) {$q_3$};

   \node [and port] (and04) at (0,8) {};
 \node [and port] (and05) at (0,10) {};
 \node [and port] (and06) at (0,12) {};
 \node [and port] (and07) at (0,14) {};

 \node
\end{tikzpicture}
\end{center}
```

übung

```
jetzt uebung:
\begin{verbatim}
```

```
node node node node node node node
state state state state state
and port and port and port and port
state and port state and port state and port

node state and port node state or port
node node node node node node
\node \node \node \node \node
and port
or port and port or port and port or port
state state state state

\node[state]
\node[and port]
\node[or port]

\node[state]
\node[and port]
\node[or port]

\node \node \node \node
or port and port or port and port or port
and port and port or port
\node \node \node \node
state state state state state

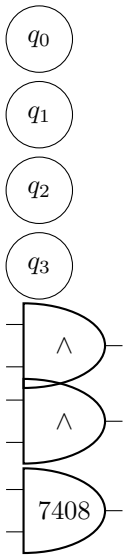
\node [state]
\node [state]
```

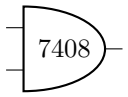
ok, mit namen

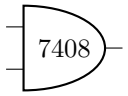
```
\node [state] (q0);
\node [state] (q1);
\node [state] (q2);
\node [state] (q3);
\node [state] (q4);
\node [state] (q5);

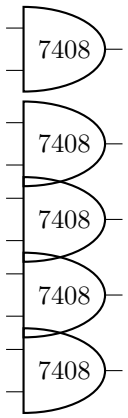
\node [or port] (or0);
\node [and port] (and0);
\node [or port] (or1);
\node [and port] (and1);
\node [or port] (or2);
\node [and port] (and2);
\node [or port] (or3);
\node [and port] (and3);
```

```
\node [state] (q0);
\node [state] (q1);
```









so, das sind und gatter:

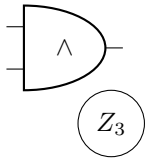
ok, das tut nicht so gut:

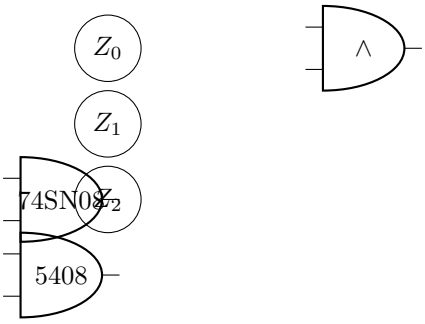
```
\node \node \node \node \node \node
and port or port and port or port and port or port
state state state state state
\node [state] (q0);
\node [state] (q1);
\node [state] (q2);

\node [and port] (and0);
\node [and port] (and1);

\node [and port] (and2) at (4,0) {$\wedge$};
\node [and port] (and3) at (4,4) {$\wedge$};

\node [state, below of=and2] (q3);
```






```
\node [state] (q0) {$Z_0$};
\node [state, below of=q0] (q1) {$Z_1$};
\node [state, below of=q1] (q2) {$Z_2$};

\node [and port, below of=q1] (and0) {74SN08};
\node [and port, below of=and0] (and1) {5408};

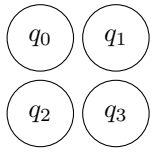
\node [and port] (and2) at (4,0) {$\wedge$};
\node [and port] (and3) at (4,4) {$\wedge$};
```

```
\node [state, below of=and3] (q3) {$Z_3$};
```

nun folgt der Automat, aus der letzteren Tabelle:

5 Automat

z_n	x	z_{n+1}	y
4	0	3	1
4	1	2	0
3	0	2	0
3	1	1	0
2	0	1	0
2	1	4	1
1	0	4	1
1	1	3	1



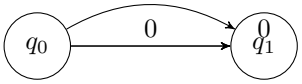

```
\begin{tikzpicture}[->, >=stealth', node distance=3cm, auto]
```

ok, es geht nicht anders als so: weil, man sieht, die knoten sind zu sehr aneinander.





z_n	x	z_{n+1}	y
4	0	3	1
4	1	2	0
3	0	2	0
3	1	1	0
2	0	1	0
2	1	4	1
1	0	4	1
1	1	3	1





diese formulierung ist, sagen wir auf das gerichtet, wo in der tabelle, die zustände nicht anders kodiert sind, als wie in der mathematik bisher üblich. wir nehmen in folge noch arabische zeichen. ok. doch wir brauchen eine binäre kodierung und: eine: mit unärer, am besten auch in der tabelle. vorher allerdings der graph so mit dem, was das betrifft.

ok, das

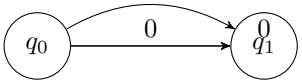
```
\begin{tikzpicture}[->, >=stealth', node distance=3cm, auto]
\node [state] (q0) {$q_0$};
\node [state, right of=q0] (q1) {$q_1$};
\node [state, below of=q0] (q2) {$q_2$};
\node [state, right of=q2] (q3) {$q_3$};

 \path (q0) edge node [midway, above] {0} (q1);

 \draw (q0) -- (q1) node [bend left, above] {0} ;

\draw (q0) to [bend left, above] (q1);
\end{tikzpicture}
```

ergab das hier:





damit nehme ich:

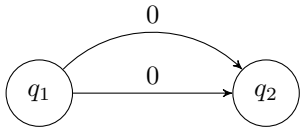




ok, dann machen wir folgendes:

z_n	x	z_{n+1}	y
4	0	3	1
4	1	2	0
3	0	2	0
3	1	1	0
2	0	1	0
2	1	4	1
1	0	4	1
1	1	3	1

ok:





ok, habe ich damit erreicht:

```
\begin{tikzpicture}[->, >=stealth', node distance=3cm, auto]
\node [state] (q1) {$q_1$};
\node [state, right of=q1] (q2) {$q_2$};
\node [state, below of=q1] (q3) {$q_3$};
\node [state, right of=q3] (q4) {$q_4$};

\draw[->] (q1) -- node[bend left] {0} (q2);

\draw (q1) to[bend left=45] node[above] {0} (q2);
\end{tikzpicture}
```

die einzige bei mir bisher funktionierte version, ist somit die

```
\draw (q1) to[bend left=45] node[above] {0} (q2);
```

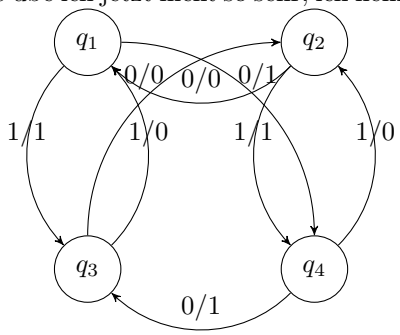
ok, ich sage gleich:

1. ich weiss backslsh node
2. ich weiss port and
3. ich weiss state
4. ... und zwar 100 pro. das heisst: ich kann
   - mit koordinaten
   - mit below und right und so weiter (of)
   - mit name in runden klammern und beschriftung in geschweiften...

ok, wenn wir das haben.

- path
- draw

das funktioniert ordentlich, das übe ich jetzt nicht so sehr, ich nehme, das gerade



beschriebene. ok, mit tabelle.

z_n	x	z_{n+1}	y
4	0	3	1
4	1	2	0
3	0	2	0
3	1	1	0
2	0	1	0
2	1	4	1
1	0	4	1
1	1	3	1

wir machen daraus mealy vorher eine andere kodierung.

eine andere kodierung, zunächst mit der richtigen nummerierung:

z_n	x	z_{n+1}	y
3	0	2	1
3	1	1	0
2	0	1	0
2	1	0	0
1	0	0	0
1	1	3	1
0	0	3	1
0	1	2	1

z_n	x	z_{n+1}	y
0b11	0	0b10	1
0b11	1	0b01	0
0b10	0	0b01	0
0b10	1	0b00	0
0b01	0	0b00	0
0b01	1	0b11	1
0b00	0	0b11	1
0b00	1	0b10	1

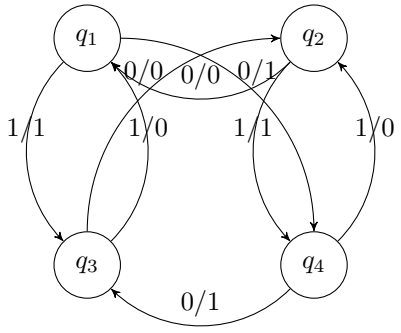
nachdem wir das nun erreicht haben, kommt die binärkodierung...

z_n	x	z_{n+1}^+	y
0b1000	0	0b0100	1
0b1000	1	0b0010	0
0b0100	0	0b0010	0
0b0100	1	0b0000	0
0b0010	0	0b0000	0
0b0010	1	0b1000	1
0b0001	0	0b1000	1
0b0001	1	0b0100	1

jetzt die unäre kodierung. jetzt stimmt die kodierung. als nächstes der moore automat. alle knoten müssen gespalten werden, die von der kante, als eingang, also nachbar, von dem es ausgeht. sowohl in der ausgabe, 1 haben, als auch 0. kurz sagen wir XOR

z_n	x	z_{n+1}^+	y	XOR
0b1000	0	0b0100	1	1
0b1000	1	0b0010	0	1
0b0100	0	0b0010	0	0
0b0100	1	0b0000	0	0
0b0010	0	0b0000	0	0
0b0010	1	0b1000	1	0
0b0001	0	0b1000	1	1
0b0001	1	0b0100	1	1

dazu gehören, ich mache da sternchen dran:



z_n	x	z_{n+1}^+	y	XOR
0b1000	0	0b0100	1	1
0b1000	1	0b0010	0	1
0b0100	0	0b0010	0	0
0b0100	1	0b0000	0	0
0b0010	0	0b0000	0	0
0b0010	1	0b1000	1	0
0b0001	0	0b1000	1	1
0b0001	1	0b0100	1	1

also muss aus knoten 0b0100, muss 0b0010

z_n	x	z_{n+1}^+	y	XOR
0b1000	0	0b0100	1	1
0b1000	1	0b0010	0	1
0b0100	0	0b0010	0	0
0b0100	1	0b0000	0	0
0b0010	0	0b0000	0	1
0b0010	1	0b1000	1	1
0b0001	0	0b1000	1	0
0b0001	1	0b0100	1	0

jetzt wird das verdoppelt: für moore:

z_n	x	z_{n+1}^+	y	X
0b1000/0/1	0	0b0100/1	1	
0b1000/0/1	1	0b0010/0	0	
0b0100/0	0	0b0010/0	0	
0b0100/0	1	0b0000/0	0	
0b0010/0/1	0	0b0000/0	0	
0b0010/0/1	1	0b1000/1	1	
0b0001/1	0	0b1000/1	1	
0b0001/1	1	0b0100/1	1	

jetzt wird das verdoppelt: für mealy. ok: jetzt machen wir folgendes:

wir haben jetzt einen moore automaten, und wenn wir den haben, ...

dann habe ich beim ausgang, ein mal mit /0 und ein mal /1

ok: und, er bekommt, so zu sagen, das doppelte an zeilen. nur, weil aber zwei zustaende von 4 verdoppelt werden, 8.

z_n	x	z_{n+1}^+	y	XOR
0b1000/0	0	0b0100/1	1	1
0b1000/1	0	0b0100/1	1	1
0b1000/0	1	0b0010/0	0	1
0b1000/1	1	0b0010/0	0	1
0b0100/0	0	0b0010/0	0	0
0b0100/0	1	0b0000/0	0	0
0b0010/0	0	0b0000/0	0	1
0b0010/1	0	0b0000/0	0	1
0b0010/0	1	0b1000/1	1	1
0b0010/1	1	0b1000/1	1	1
0b0001/1	0	0b1000/1	1	0
0b0001/1	1	0b0100/1	1	0

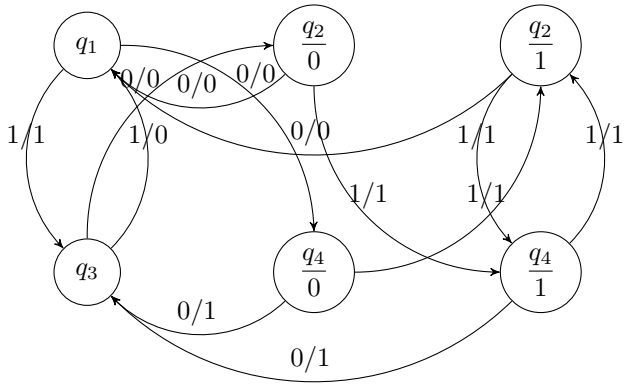
man kann das noch anders begreifen:

z_n	x	z_{n+1}^+	y	XOR
$\frac{0b1000}{0}$	0	$\frac{0b0100}{1}$	1	1
$\frac{0b1000}{1}$	0	$\frac{0b0100}{1}$	1	1
$\frac{0b1000}{0}$	1	$\frac{0b0010}{0}$	0	1
$\frac{0b1000}{1}$	1	$\frac{0b0010}{0}$	0	1
$\frac{0b0100}{0}$	0	$\frac{0b0010}{0}$	0	0
$\frac{0b0100}{0}$	1	$\frac{0b0000}{0}$	0	0
$\frac{0b0010}{0}$	0	$\frac{0b0000}{0}$	0	1
$\frac{0b0010}{1}$	0	$\frac{0b0000}{0}$	0	1
$\frac{0b0010}{0}$	1	$\frac{0b1000}{1}$	1	1
$\frac{0b0010}{1}$	1	$\frac{0b1000}{1}$	1	1
$\frac{0b0001}{1}$	0	$\frac{0b1000}{1}$	1	0
$\frac{0b0001}{1}$	1	$\frac{0b0100}{1}$	1	0

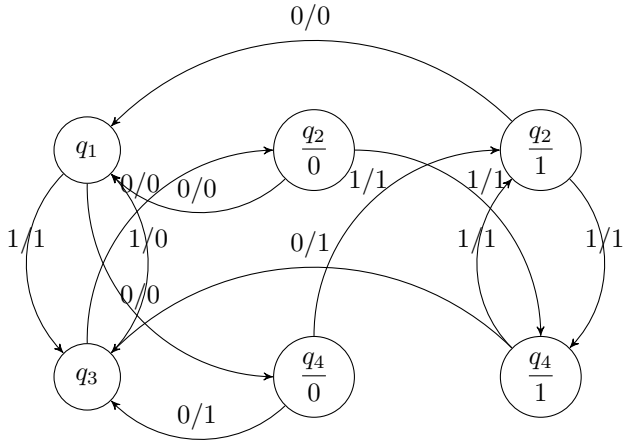
und jetzt der automat:

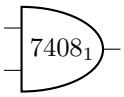
das mit dem verdoppeln war quatsch, das sehe ich gleich.

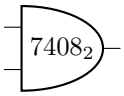
z_n	x	z_{n+1}	y
4	0	3	1
4	1	2	1
3	0	2	0
3	1	1	0
2	0	1	0
2	1	4	1
1	0	4	0
1	1	3	1

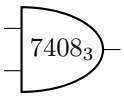


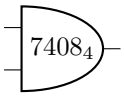
hier eine andere kanten:











da mache ich morgen weiter, das könnte jetzt zu quine mc cluskey führen, aber ich mache es nicht, das könnte auch hier rein. das ist das eine. und ich könnte das schaltwerk weiter machen, mache ich nicht, für heute ist pause. ich nehme jetzt noch das eingescante rein und dann kommt das mikroprogrammsteuerwerk rein und dann passt die sache. dann kommt auf die homepage. als pdf mit LaTeX. und mit latex2html

das mikroprogrammsteuerwerk:

Daniel Vagaa(x), 08/24/2026
Schaltwerke, Seite 3

Eine andere Idee ist daß ich die Schaltwerke auch mit Würfeln tue.
Also generell dienen die Würfel dazu
a) Wahrscheinlichkeitsrechnung
b) Die Reihen, seien es:
- √ Crash
- √ CRC
- √ Verschlüsselung
- √ Fletcher's checksum

oder Codes:
- √ Airen
- √ Excess-3

c) aber auch die Umrechnung von Zahlensystemen.
Rechnen wir mit der Zuordnung
$\{f(1)=0,\ f(x=2)=1,\ f(x=3)=2, \ldots\}$
haben wir es mit einem 6er System zu tun.
a) Das lässt sich ins Dreier umrechnen
b) bis 10er...

Jamil Vojcla (c)
08/24/2026
Schaltwerk mit Latex, Seite 1

also:
1.) Die erste Idee, die ich habe
reutet sich an den papier-
Stapel, den ich habe
a) Da die Unruhe hier
merklich/spürbar (deutsch)
bin ich noch ein raus
gegangen vor die Stadt
und dort habe ich die
Idee verbessert
b) Es handelt mehr oder
weniger um zwei Ideen..
2.) Der Stapel papier ist
hinzuziehend bezeichnend
3.) ich habe xx oder so, ja längst
ein anderes verfahren für mich
eingeführt. nämlich bei tabellen
und graphen würfel real zu
benutzen:
√ bei der Umrechnung vom
binärsystem kann der
nur nützlich sein.
√ Die Zufallszahlen so wie
alle weiteren Reihen...

David Vajda (c)
08/24/2026
Schaltwerke, Seite 2

lassen sich besser auf papier oder
jetzt am Computer das ist jetzt
der neue teil der idee...
also wie früher alle
ein laut auswendig gelerntes.
a) Regn gehört minix
b) astm cortex
c) vieles weitere
am Computer.
Das ist durchaus den Code auswendig
schreiben, z. B von minix.
Binäres dividieren, oder über-
haupt schriftlich geht am
Besten mit Excel.
Ich benutze meinen großen pc meist
aber mein ibook mit Linux
kann diese auf einem externen
Datenträger auch benutzt. Erleichtert
Zahlen
Netzwerk...
Das muß allerdings eingerichtet werden...
Der papier Stempel ist bezeichnend.
von daher jetzt alles am Computer

Was man damit auch machen kann sind graphen.

a) Ziel ist es alles zu verwalten

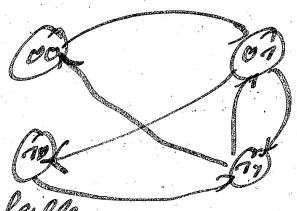
b) neben den graphen, das sind Zwei: Kommen Wahrheits-tabellen in frage.
nebenbei das Schlechte:
y - Wort
- Byte
- Doppelwat
und so weiter
a) Das lässt sich kodieren
b) Den ganzen als Matrix berechnen...
c) oder: crc... anwenden.

Doch die graphen sind
a) ein feine graphen
b) com-drogramm

Die graphen dienen zum Schaltwerken.
Das heißt Automaten.
für einen einfachen automaten:
soll auch ein Schaltplan dazu
Die erste Idee hatte nun, das gleich mit tikz in Latex zu machen

(c) Jamil Vajda
08/24/2026
Schaltwerke, Seite 5

Wenn man nun den einfachen graphen
anguckt:
für "einfache" Automaten.
Denn ist das ein graph aus 4
Zuständen in dem fall:
bei dem (zwischen zwei Zuständen)
pro Zustand zwei Kanten existieren
so z.B.



in dem falle
weniger Kanten...
Das lässt sich einfach mit
Würfeln realisieren. Schon
auf eine andere art und
weise.
Man würfelt nun Kante für
Kante sondern z.B ein langes wort:
5463126653...
Daraus lassen sich die entsprechenden...
entnehmen...
Bei diesem einfachen graph für
ein Schaltwerk..

Daniel Vajda(c)
08/24/2026
Seite 6, Schaltwerk..

fehlt noch das übergangsschaltnetz
und das ausgangsschaltnetz...
Das ist entsprechend als Schaltplan
zu machen.
auch am Computer.
mit (E) Kaneda...
allerdings lässt sich das
alles mit Tikz in Latex
machen.
Ein wenig schwierig.
hier brauche ich 4 zustände, muss
die entsprechend verbinden.
Tikz bietet viel
Vielleicht muss die Maße
für alles mit Worten einzeln:
wenn man den oft macht
geht es anders.
Die zweiten graphen betreffen komplexe
automaten.
Dabei müssen ganz konkret unterscheiden.
a) der begriff Hardware-
Algorithmus

Daud Vajda (c) 08/24/2026
Schaltwerke, Seite 11

1.) man muß den Schaltplan
des Steuerwerks machen
2.) den Schaltplan mit allgemeinen
Registern des Operationswerks
3.) man muß den Hardware
algo über würfeln:
a) erzeugen
b) aufzeichnen.

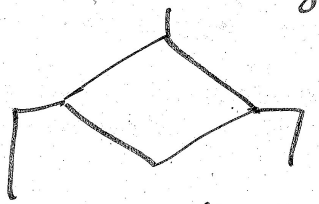
Nach ihm teilen sich
1.) Steuerwerk
2.) operationswerk.
was im asm Diagramm neben
Zuständen was Knoten sind enthalten
ist (Knoten mit 3 und 2 Wahlbaum
und Kanten)...
ist Ahl notation.
Diese definiert operanden. 6 Register z.B.
r6, r5, r4, r3, r2, r1
... bzw r6, ..., r0
so wie operationen: 6
add (+) r7 ← r5 + r4
sub (-) r5 ← r5 - r4
aber auch vergleiche. normal gibt
es drei >, =, < trichonomerie
malen ...

Samvid Vyasa (c) 08/24/2026
Schaltwerke, Seite 10

Es lässt sich meinerseits über das
Wort Modus trefflich streiten. Aber

Zustand ⇔ Modus

erhält sein Steuerwort von dem
Operationswerk über verschiedene.



gleichzeitig ist Steuerwerk = automat.
um das Verhalten zu beschreiben.
braucht es einfach ein Diagramm
Das ist wiederum ein graph
mit Knoten mit
zwei Kanten:
Herein und Heraus
vierech waagerecht

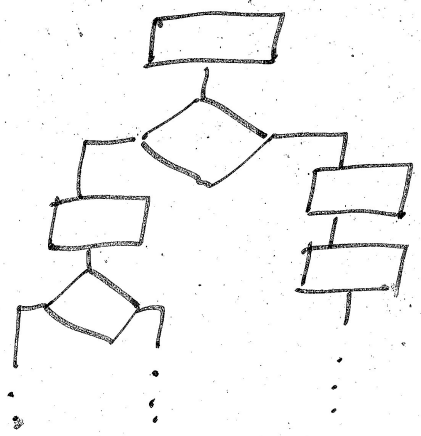
drei + Kanten: Raute..
Es steuert das Operationswerk
besteht aus befehlen. Zu tun ist
aufgabe

Jawid Vojdan (c)
08/24/2026
Schaltwerke, Seite 8

Doch sind auch in ihrer fülle
mathematisch Menge * oft
unterschiedlich und verschieden zahlreich
Calteströmlich deutsche Begriffe (etwas
wie Comic halt beschriebenes Begriff -
interpretierbar? aber so nicht gemeint))
Also:
Entscheidend neben den Operationen
sind aber auch Regisks.
Die Operanden sind
a) Quelloperand(en) ten?
b) Zieloperant...
Diese Regisks können z.B.
r5.. r0
sein.
Die operationen wiederum *
1.) Das operationswerk
a) führt operationen aus
b) hat operanten (Regisks)
Jedweiteren über das Steuerwerk
gesteuert.
2.) Das verhalten des Steuerwerkes -
ein eigentliches - das eigentliche
Unterwort - der sein Steuerwort
man kann es auch Steuermodus
nennen.

(c) David Vyčla
08/24/2026
Schaltwerke, Seite 7

Hardware algorithmen werden
generell mit asm Diagrammen
beschrieben:



Dabei müssen wir aber Streng-
unterscheiden:

1.) zwischen Operationswerk
2.) Dem Steuerwerk.

Das Operationswerk ist in der Lage
a) vergleiche auszuführen
b) rechenoperationen zu:
grundrechenarten:
vor allem: addition + subtraktion
inkrement/dekrement auszuführen
c) so wie links und Rechtsschieben.

David Vejda (c)
08/24/2026
Schaltwerke, Seite 9

Zu unterscheiden sind Zustand (a) und (b) modus
in einem Schaltnetz hängt der Zustand wohl nur vom Modus, der wahl das Eingabewort ist.
Das Eingabewort... regelt den Modus

In einem Schaltwerk... wir benutzen einen Graphen in der Mathematik so genannt...
Dieser stellt im Automat mit den Knoten zustände dar.
Also: Das wort einerseits eingabewort aussen + den Modus den ein Zustand selbst erzeugt entstehen der Modus für den Folgezustand

Anmerkung:
Es ist bei den übungen für "simple" Automaten auch für übungen für umwandlung mealy → moore zu sorgen.

So: Das Steuerwerk wiederum erhält unabhängig - Länge rede kurzer Sinn gerade eben- ...

Jund Vayda (c)
08/24/2026
Graph. Seite 12 (Schaltwerk)

Diese + der 6 Register lassen sich kodieren wie beim Computer üblich

add: 0
sub: 1
shl: 2
shr: 3
inc: 4
dec: 5 ...

so wie die operanden:

r5: 5
r4: 4
r3: 3
⋮
r0: 0

aber auch die vergleiche:

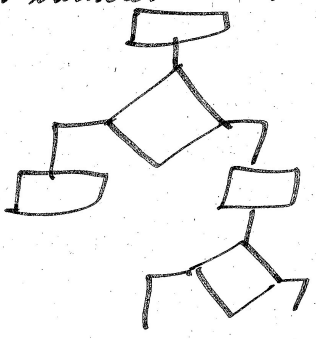
ne: 0
eq: 1
le: 2 ...
ge: 3
lt: 4
gt: 5

So: Dann haben wir zustände die sind nummeriert:

z0
z1
z2
⋮
zn

David Vajda (c) 08/24/2026
Schaltwerk, Seite 13

Dabei handelt es sich um graphen



und die nachbarn (knoten kanten)
sich auswirfeln.

bei einem gesamten wirfelt
man ein großes wort
6534122...

und zerlegt es entsprechend.
auch das läßt sich mit tilz
vollziehen...

Doch: ich möchte
dies mal ausprobieren
und deswegen heute nach
normal mit papier...
avg. wirfel...

David Vajda (c) 08/25/2026
Seite 1 asm
So: Diagramme...

also)
man nimmt 11 Zustände
oh: 12 oder 13
oder 16. Sonst hat man Mikro-
programmsteuerwerk mit 8
Leitungen...
Ein komplexes Schaltwerk
lässt sich nicht mit einem
Übergangsschaltnetz lösen.
Das... einfache ein
überg... Schaltnetz ist
anschaltnetz ist das auch...

ww: Das muß ein Komplexes
mit Eprom oder fuses sein...
und das hat mit 8 zu wenig...
also 16 und 4 bit Eingang...
So: Dann nehmen wir
für das asm Diagramm:
16 Zustände maximal
Da wir Ranten haben:
wir brauchen 3 Leitungen...
mindestens: ...
also: Dann würfeln wir:
halt! 1. Mal!

David Vajda (c)
08/25/2026
Diagramm, Seite 2

also: die erste Bedingung ist:
Dass: man einfach würfelt:
also man würfelt eine ganz
lange (Kette) Zahl:

3452162216...

sehr lange:

und mit umrechnung!
z.B über umrechnung
transformation und so
weiter:
a) kann man nach
binär
b) zehnersystem
c) <- grob c: also vor allem!
über den Rest - Rest!

über den Rest können wir
jeweils für die entsprechenden
Aufgaben:

1.) Zustände
2.) Verlinkte Zustände
3.) Nummern der Befehle..
4.) Register und operanten..

die für die entsprechende Sache
für das entsprechende 11...16 Zustände
2 Register die Zahlen ausrechnen

Jamd Vayda (c)
08/25/2026
Seite 3, asm Diagramme.

So: wir würfeln eine entsprechend
lange Zahl...
- Zustände (11... 16)
- Regiotr
- Operation
wird im entsprechenden einzeln aus
der Lade gezogen...
ganz einfach...
nun müßen wir erst mal:
16 Zustände festlegen.
Sagen wir: wir haben das...
und weil wir das haben:
würfeln wir nun ob
Zustand Z0, Z1, Z2, ... Z15
Die Zustände sind durchnummeriert..
müßen wir nun auswürfeln:
wenn auch bei Oyotem..
Dann können wir von 65 z.B.
Rest 16 rechnen ohne besondere
Vorstellung.
3 Zustände werden dann
benötigt.
ok: wenn das (gasdrehen ist:) falsch.
bin gerade wo anders im Zimmer herumgelaufen
1.) das ganze findet dies mal

Daniel Vogdacci
08/25/2026
Seite 4, asm Diagramme

mit:

papier statt...

ich sage ja: es muß in Zukunft
automatisch mit Latex gehen!
automatisch mit tikz...
automatisch!

Das man das auch so hinschreibt!
trotzdem!

Es ist jetzt erst aufgabe:
Wir sollen herausfinden!
+ asm Diagramm
+ Hardware Algorithmus
+ Operationswerk
+...

Dawid Vajda (c)
08(24)5(2026
Seite 5, dem Diagramme..

abo: habe ich gesagt



daß man jetzt nur würfelt?
und es auf papier macht?
Neinnnn!
ich habe gesagt!!
a) man würfelt ein mal
ema macht papier
b) habe ich gesagt!!!
ich habe gesagt dann
auf papier
und das dürft nicht!!!
Dass man es dann nicht
auch noch in Latex
einsetzt!!
auch nach als zweiten Schritt!
Aufgy! für alle opôkcu!

Dumd Vaya (c)
08/25/2026
Seite 6, axm Stagaraum!

Kaffee (1.)
Dumm anfangen!
Dumm anfangen!

30, 31, ... und los!

So. 1015 mesz cest
Tübingen (don't know
what: coordinates
longitude, latitude?
know: long: west/east
latitude: north south
Druck ...
würfeln? dying.. temperatur
L?
zu viel!?

fertig mit den neuen..

$\Omega = (6454262216556115431 5)$

ok: jetzt nehmen wir
1.) 1cm rede zu viel...
(Ereign) Ereignis
weise ni ..-bla bla to
much?
Ereignis nur ergebnis
Elementarmenge $\Sigma = \{1, 2, \ldots, 63$
ist die "mögliche" menge
1 bleibt als Ergebnis? ! Es bleibt das
Ob sind das Elementarereignisse!

Daniil Vajda (c)
08/25/2026
Seite 7, asm Diagramme!

Elementarereignisse...
Zweitens: sorry: zu viel da Rede!
nur
6464...
ist eine lange Zahl klaro sieht aber als
1.) elementarereignisse adressen
2.) tupel: paare: trippel
(a, b): paar
(a, b, c): tripel
$(a_n, \dots, a_0)$: n-tupel
und:
wenn man:
blau und gelben
würfel hat:
(blau, gelb): paar
aber (6,4, 3,4, ...)
gemultiplexed! ok...
^sry... nicht zu laut?
(:)??
Ja ob wg. ??!
ok: jetzt haben wir die Zahl:
Ω 08262026daniavaj0001

Dawid Vajda (c)
08/25/2026
Seite 8, vom Diagramm...

Dann nehmen wir die Zustände

$Z = \{z_0, z_1, \dots, z_{15}\}$

ab und schneiden von der Zahl:

0x08252026dawidvajda0001

or what ever →

Stücke ab...

ab erste drei für 3 der 16 Zustände

bei 16 Zuständen ist max:

16

mal 3. aber wir nehmen

den anfang

$64 \% 16 = x \stackrel{?}{=} ??$

irgendwas

wir könnten den anderen liegen

lassen. bei modulo zwangsweise

nötig?

auch nicht!

wir addieren es zum nächsten-

kurz weiter verarbeitung

bis so etwas gegen null

noch?

damit nicht nächste

Zahl: jetzt. vorher:

raucherpause..

Dana Vojáa (c)
08/25/2026
Seite 9, vom Diagramme...

ok: jetzt bleibt ein problem:
1.) die würfelfolge besteht aus
$S = [1,2,3,4,5,6]$
in das: 6er system umgesetzt
bedeutet das
$[(x()]\ f(x=1) = 0$
falsch $f(x=2) = 1$ ...

ok: bei 16 zuständen ist das
umgeschickt. Das ist zwar hex
System! 16
0,1,2,.. f (mit 0x)!
trotzdem 16 existiert im dezimal-
system!
und es gilt zwei möglichkeiten
von bin → dec
a) wir rechnen die potenzen
zusammen

$0b\overset{2}{1}00\overset{0}{1}.0011$
$\quad\ 3\quad\ 3\,2\,1\,0$

$2^2 + 2^4 + 2^1 + 2^0$

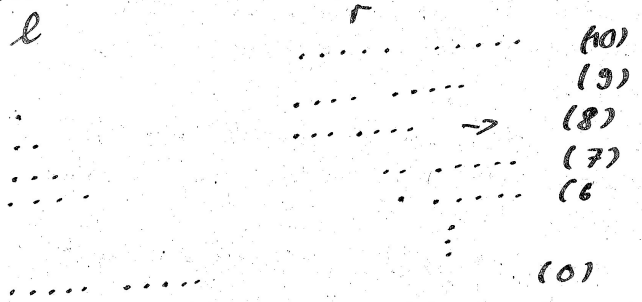
oder wir machen das was wir
verwirrender weise "anders herum"
machen würden
wir teilen durch 10_{dec} =
1010_{bin} aber im bin system. nehmen
den Rest..

Daniel Voyda (c)
08/25/2026
arm Programme, Seite 10

Vorwort!!
Jetzt lange rede!

a) Das ist was man üblicher bei bcd tut...
Zahlenumrechnung für darstellung am Bildschirm
↑ Dies ist aber geringe argumentation!

1.) wenn man abakus anguckt hat man



Das sind aber die indischen Ziffern:
0, 1, 2, ... , 10
und nicht einfach bis ebenso wie arabische



und maya 

dabei:  = 7 also ist eine Ziffer:



David Vojda (c)
08/25/2026
Vom Diagramm: Seite II

Zwar!
ist bcd ein typisches Fall
von Darstellung auch ohne
jedes teilen!

0b0000 : 0d ... 0b1010 : #^{10}
fehlt

Trotzdem entsteht es bei division
durch 10_{dec}
ist aber binär 1010_{bin}!

und : ob zwar :
0b0000 : binär $\hat{=}$ 0_{dec} : dezimal ?!
ist in bcd trotzdem :
0 b 1000 : als dezimalziffer 8
zu sehen!

Ah : bcd :
nicht nur unser klassisches
phänomen zur darstellung
am Bildschirm!

man rechnet um :
$dec(0b1001.0011) = 2^7 + 2^4 + 2^1 + 2^0$
$= 128 + 16 + 2 + 1$

aber auch :
II.) $0b1001.0011 / 1010$
$\swarrow 0d$

David Vojda(c)
08/25/2026
Seite 12, asm diagramm
Dieses zweite verfahren ist
wirrig!
und!
wenn wir mit würfel
1...6 (0...5)
würfeln aber 16 zustände haben.
müssen wir vorher in dec umrechnen.
Sonst:
Zufallszahl gestört!
ok:
Das könnten wir so machen.
teilen durch
10 im 6er system!
aber halt!
an der stelle tut sich ein
problem auf!
warum ist es eigentlich einfach!
im binärsystem zu testen?!
warum geht integer division!?
ganz einfach!
1.) schriftliches Dividieren!!!
2.) aber: noch mehr:
wenn ich schriftlich dividiere!!!

Daniel Vojda (c)
08/25/2026
asm diagramm! Seite 13

Das macht alles anderes!
verfehrisch! aber zu rein
linär dividieren ist einfach!

weil schriftlich dividieren einfach
ist!
aber auch binär!
warum

1010 kann:
beim schriftlichen dividieren
subtrahieren nicht verwechseln

1111 ← hier weil das da
− 1010 steht brauche ich:
11111 nicht
aber in
~~41 ok~~ 10000
− 1010 } 1 Stelle
drüber

maximal 1 oder 0!
nur bei 0 wg. Subtraktion
4/5 Stellen
verlor: beweis!
Beweis 1010 nein
+ 1010 geteilt
2 durch!

Dumm Vogel (c)
08/25/2026
aus Diagramm, Seite 14

$$\begin{array}{r} 1111 \\ +\,1111 \\ \hline 11110 \end{array}$$

also:

1111 gelt in 11110
nein
wird aber neu
zu
10 0000
↑
6 Stellen!
und wenn dann gelt auch!
10000
- 1111
oder 4 Stellen!
1111
- 1111
also: binär: Das beinhaltet
binär: gelt ~~2~~ 0 · 1 0 . 2
mal ist möglich!
Subtrahieren! Schriftliche
Division!
Dumm: einfach: Blöp
der System!

David Vogda (c)
08/25/2026
asm Diagramm: Seite 15

also:
es gibt ja zwei möglichkeiten:
1.) teilen!
2.) wie vorher
dec(1010)$_b$ = $2^3 + 2^1 = 8 + 10$
teile? ↑ ok $2^1 = 2$
eigentlich 10
ok: wenn ich im bin system teile:
nichts blicken!
blop

521
$= 5 \cdot 6^2 + 2 \cdot 6^1 + 1 \cdot 6^0$
$= 5 \cdot 36 + 2 \cdot 6 + 1$
im dec system!
ist:
150 + 30 + 12 + 1
193 im dec system!
und jetzt mod 16
warum eigentlich?
also computer zufallszahlen!
wenn:
dann teile (wie hash??)
mod? ja und am pc
ja: blop. hier pyg? / zufall
python3 + l
16
21 oder

David Leyda (c)
08/25/2020
aom.... Seite 16

also:
Zufall:
2 rest
in prog!
wg. max: wer
die Reihe
jetzt: Diagramm
+ Darbe Computerspiele I/II
Bildung, Schiffman,....
!! kommt liegt
Idlige, Baum?
↑ suchen!!!

Jawid Vajda (1)
08/25/2026
Seite 17
asm diagramm:

also die Zahl:

64 im Dezimalsystem ist

$6 \cdot 6^1 + 4 \cdot 6^0$

$= 36 + 4 = 40$

oh 40 mod 16 = (32, 2, 8)

↑ hä?

mod paar(

(der quotient, rest)

aust rest

(32, 2, 8) @ mod 3 ...

also 8

38 ist Doule

Lecht.

pause machen weiter...

David Vajda (c)
08/25/2026
asm Diagramm, Seite 17

Q = 64542522155561154315

gut:
64, 54, 26, 22

16 Zustände $Z = \{z_0, z_1, \dots z_{15}\}$

welche 3 Zustände sind ◇ ?

welche Nachbarzustände ? (Kanten?)

$dec(64_{6er}) = 6 \cdot 6^1 + 4 \cdot 6^0$
$= 36 + 4 = 40$

und 40 mod (rest) 16 = 8

ok... bleibt 32 als: quotient · divisor
2 · 16

$32 + 54 = 86_1$

$dec(86_{6er}) = 8 \cdot 6^1 + 6 \cdot 6^0$
$= 46 + 6 = 52$

52 rest 16 = (32 + 16 + 272) rest 16
= (48 + 4) rest 16 = 4 $\quad z_{m_2} = 7$

ok: $z_{m_0} = z_4$ $\quad$ (25 + 48) rest 16
$z_{m_1} = z_9$ $\quad$ = 73 rest 16 =
= (64 + 7) rest 16 = 7

Jamd Vajda, Seite 18
08/25/2026
asm Diagramme...

nun kommen die
kante$_{2\,out}$
vertices:
3 vertices: 3 nachbarn:
wie bei normalen...
Knoten für anweisungen
1 in
1 out
2 nachbarn zwei kanten
bei Kante beziehung:
3
1 in, 2 out...
3 kanten, 3 Nachbarn...

64+22 = 86
86 rest 16 sind:
64+16 = 80 ✗ falsch
5·(8)16 = 50+30 ✓ ok
also rest 6
$v_{m_1} = (4,6)$
$v_{m_2} = (15\ 8) = (8,15)$
80+15 = 95
95 rest 16 = 15
$v_{m_3} = (7,6)$...
15+55 = 70
rest 16
= 6

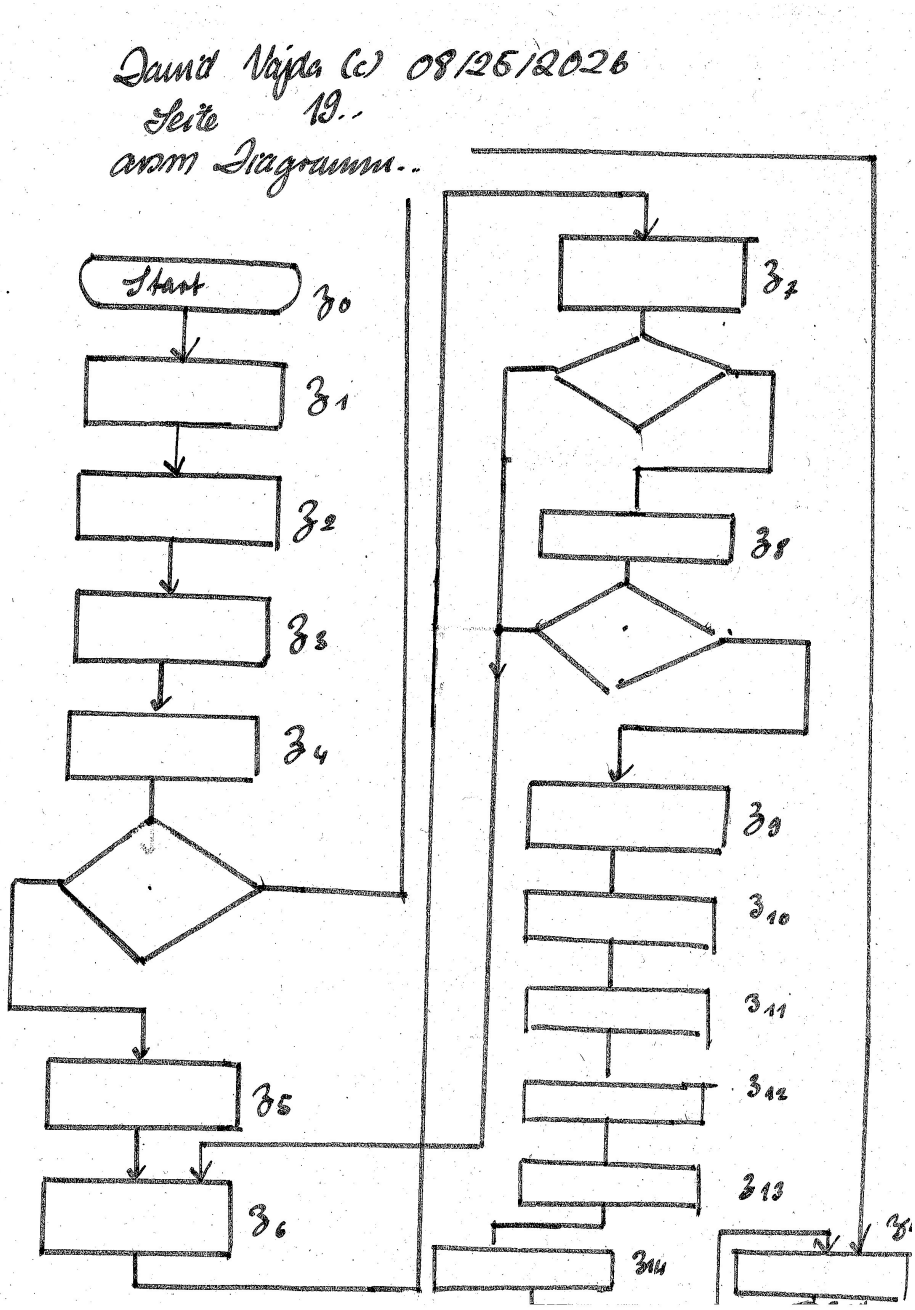
David Vajda (c)
08/25/2026
Seite 20
Asm Diagramm...
nun folgen

Register
operanden + operatoren...

r0: 1 x=1, 0
r1: 2 x=2, 1
r2: 3 x=3, 2
r3: 4 x=4, 3
r4: 5 x=6, 4
5
r5: 6 x=6, 5

oh: 3 operatoren für anweisungen
kein inkrement
dekrement
r1 ← r1 + r2
wobei vorher
r2 ← ?
oh: hier gelt es nicht
braucht man aber nicht..

add:	+	0 (1)	> :	0 (1)
sub:	-	1 (2)	= :	1 (2)
mul:	*	2 (3)	< :	2 (3)
div:		3 (4)	!= :	3 (4)


David Vajda (c) 08/26/2026
Seite 19..
vom Diagramm..
Start
Z0
Z1
Z2
Z3
Z4
Z5
Z6
Z7
Z8
Z9
Z10
Z11
Z12
Z13
Z14

Daniel Vajda (c) 08/25/2026

Seite 21

vom Diagramm...

wir haben vergessen den 6er

selbst 6er System dann zu

machen. Mach mir's.

Dann war es halt in dem

Falle so.

dazu: wir müssen

(f(x=1)=0, f(x=2)=1, ...,

f(x=6))

und dann kommen Register +

Befehle +... (+) indem wir oft

↑ Fehlerquelle

↑ merkt ob?

eine Folge von Zahlen erzeugen...

6er Folge r6, ..., r0

4er Folge: für opc:

add/sub/mul/div +, -, *, /

>, =, <, !=

0, 1, 2, 3...

David Vayda (c)
08/25/2026
Seite 22, aom diagramm...

64 54 25 22 15 55 61 15

$6 \cdot 6^1 + 4 \cdot 6^0 = 36 + 4 = 40$
$5 \cdot 6^1 + 4 \cdot 6^0 = 30 + 4 = 34$
$2 \cdot 6^1 + 5 \cdot 6^0 = 12 + 5 = 17$
$2 \cdot 6^1 + 2 \cdot 6^0 = 12 + 2 = 14$
$1 \cdot 6^1 + 5 \cdot 6^0 = 6 + 5 = 11$
$5 \cdot 6^1 + 5 \cdot 6^0 = 30 + 5 = 35$
$6 \cdot 6^1 + 1 \cdot 6^0 = 36 + 1 = 37$
$1 \cdot 6^1 + 5 \cdot 6^0 = 6 + 5 = 11$

ok nun: wir brauchen:

16 Zustände: ____ keine notation
= + 3 + 13 (2+3) keine distributivität...

3·16 Register à 6 Werten
(0..5)

40 mod 6 =

falsch wir hätten eben
0..5 drauf machen
sollen...

David Vajda (c) 08/25/2026

Seite 23, zum diagrammieren:

64 54 23 22 15 55 61 15

$\sim$

53 43 14 11 04 44 50 05

ok: so besser...

$5 \cdot 6^1$... ok wir nehmen wir den falls den Taschenrechner zur Hilfe

$$a_3 \cdot 6^3 + a_2 \cdot 6^2 + a_1 \cdot 6^1 + a_0 \cdot 6^0 + \dots$$

$6^{10} =$.
$6^9 =$:
$6^8 =$
$6^7 =$
6^6 :
:

besser anders herum

$(a_n) = 6^n$

$(a_n) = (a_{n-1}) \cdot 6$

$(a_0) = 1$

ok: wirst (a_n) machen anderen (b_n)... basis jedenfalls

$(a_0) = 1$
$(a_1) = (a_0) \cdot 6 = 1 \cdot 6 = 6$
$(a_2) = (a_1) \cdot 6 = 6 \cdot 6 =$
$(a_3) = (a_2) \cdot 6 =$

David Vajda (c)
08/25/2026
Seite 22, asm diagramm...

64 54 25 22 15 55 61 15

$6 \cdot 6^1 + 4 \cdot 6^0 = 36 + 4 = 40$
$5 \cdot 6^1 + 4 \cdot 6^0 = 30 + 4 = 34$
$2 \cdot 6^1 + 5 \cdot 6^0 = 12 + 5 = 17$
$2 \cdot 6^1 + 2 \cdot 6^0 = 12 + 2 = 14$
$1 \cdot 6^1 + 5 \cdot 6^0 = 6 + 5 = 11$
$5 \cdot 6^1 + 5 \cdot 6^0 = 30 + 5 = 35$
$6 \cdot 6^1 + 1 \cdot 6^0 = 36 + 1 = 37$
$1 \cdot 6^1 + 5 \cdot 6^0 = 6 + 5 = 11$

ok nun: wir brauchen:

16 Zustände: ____ Lebode notation
$= +3 + 13$ (2+3) keine distributivität...

3·16 Register à 6 Werten
(0..5)

40 mod 6 =

~~Falsch~~ wir hätten eben
0..5 draus machen
sollen...

Daniel Vajda (a)

08/25/2026

Seite 23, zum Diagramm..

40341714 nehmen wir jetzt
als Zahl und den Taschenrechner
zur Hilfe: Das heißt:
wir teilen durch 16
wenn wir das tun der Taschenrechner..
teilt und macht nicht wie
intel div befehl
div bx
wobei dx:ax
geteilt wird
oder
div bh ? bl geht dl ?
je zwar wäre operand nicht
möglich
div dl
dann ax...
ok: dann würde der Rest...
dx würde der Rest
ax enthält den ganzzahl
quotient..
wenn ich taschenrechner nehme,
bleibt eine Nachkommazahl...
wie den Rest herausfinden...
ich ziehe das vor dem Komma
ab und multipliziere mit dem Divisor?

David Vejda (c) 08/25/2026
Seite 24, asm diagramm...

ok: Dann habe ich den Rest.
Den teil vorne merken. also los
geht es...
(Rest (zu) mal 16x durch)
Das ist wichtig
16x durch 4 teilen
Rest : operation
16x3 x durch 6 teilen
operanden
ist asm Diagramm..

David Vejda (c) 08/25/2026
Seite 24, asm diagramm...

ok: dann haben wir den Rest.
den teil vorne merken. also los
gehts es...

(Rest (zu) mal 16x durch)
das ist wichtig
16x durch 4 teilen
Rest: operation
16x3 x durch 6 teilen
operanden
im asm Diagramm..

gut halt: nun haben wir
eine andere Zahl.
wir haben:

$6^1 \cdot a_1 + 6^0 \cdot a_0$

ok. doch wir müßten haben:

$6^n \cdot a_n + 6^{n-1} \cdot a_{n-1} + \dots + 6^0 a_0$

ok...
und doch ist das dasselbe: Man
denke an binär:

09 1001. 1100

dann haben wir

09 1001. 1100 (3 2 1 0 / 3 2 1 0)
3 2 1 0
1001 . 1100
wirft dasselbe?! Derlei ausprobieren

Daniel Vojda CO 08/25/2026

Seite 26, asm diagramm...

also:

$(a_0) = 1$
$(a_1) = (a_0) \cdot 6 = 1 \cdot 6 = 6$
$(a_2) = (a_1) \cdot 6 = 6 \cdot 6 = 36$
$(a_3) = (a_2) \cdot 6 = 36 \cdot 6 = 216$
$(a_4) = (a_3) \cdot 6 = 216 \cdot 6 = 1296$
$(a_5) = (a_4) \cdot 6 = 6 \cdot 1296 = 7776$
$(a_6) = (a_5) \cdot 6 = 6 \cdot 7776 = 46656$
$(a_7) = (a_6) \cdot 6 = 6 \cdot 46656 = 279936$
$(a_8) = (a_7) \cdot 6 = 6 \cdot 279936 \cdot 6 = 1.679.616$
$(a_9) = (a_8) \cdot 6 = 6 \cdot 1679616 = 10.077.696$

jetzt mit taschenrechner:

$15 \cdot 1 + 61 \cdot 36 + 55 \cdot 216 + 15 \cdot 1296 +$
$22 \cdot 7776 + 25 \cdot 46656 + 54 \cdot 279936 +$
$64 \cdot 1.679.616$

$=$

$2211 + 31320 + 1337472 +$
122611968

$=$

122611968
$+ \quad 1337472$
$+ \quad 31320$
$+ \quad 2211$

1

ah: taschenrechner:
123982971

ah
das ist die Zahl sieht echt aus!

sieht immer mit rest teilen...

Daniel Vajda (c) 08/25/2026

Seite 27 arm diagramm...

Das sind ~~regiole~~ operationen

123982971 / 4 = 30995742.75 ◆ 3 Rest Zahl : op3

30995742/4 = 7748935.5 ◆ Rest 0 × 4 = 0.5 = 2

7748935/4 = 1937233.75 Rest: 0.75·4 = 3

1937233.75/4 = 484308.25 Rest: 4·0.25 = 1

484308/4 = 121077 Rest 0. 0·4 = 0

121077/4 = 30269.25 ⇒ 30269 Rest 1

30269/4 = 7567.25 ⇒ 7567 Rest 1

7567/4 = 1891 Rest 3

1891/4 = 472 Rest 3

472/4 = 118 Rest 0

118/4 = 29 Rest 2

29/4 = 7 Rest ◆ 1

7/4 = 1 Rest 3

3/4 = 0 Rest 3

ok: (3, 2, 3, 1, 0, 1, 1, 3, 3, 0, 2, 1, 3, 3)
13 12 11 10 9 8 7 6 5 4 3 2 1 0

Das sind ⌊ ok fehlen 2

!!

machen wir noch

die Zahl 4×4 3×16 /16 wg. reg

oder 4 reg besser zu brauchen dann 14 ⇒ Redeyld ???

Daud Vajda (c) 08/25/2026
Seite 28 aom diagramm...

nun wandeln wir die inzwischen
existierende Zahlenfolge entsprechend
der Knoten (im graph)
bedingter
"unbedingter" Zustand...
ausdruck!

in Befehle um:

3: z_1: div /
2: z_2: mul *
3: z_4: ne not equal !=
1: z_5: sub -
0: z_6: add +
1: z_7: equal eq ==
1: z_8: eq equal ==
3: z_8: div /
3: z_{10}: div /
0: z_{11}: add +
2: z_{12}: mul *
1: z_{13}: sub -
3: z_{14}: mul *
3: z_{15}: mul

kommodore weiss hat es hier gesehen
sonst hätte ich mir bei 18 einen
schelz erlaubt von vorne anzufangen

Diese Kombi benutze ich jetzt auch für
Zielregister (operant)...
und als ersten Quelloperant 0 0 41 1
d o t dest source t:02

Daud Vajda (c) 08/25/2026
Seite 28 aom diagramm...

um wandeln wir die inzwischen
existierende Zahlenfolge entsprechend
der Knoten (im Graph)
bedingter
"unbedingter" Zustand...
ausdruck!

in Befehle um:

3: z_1: div /
2: z_2: mul *
3: z_4: ne not equal !=
1: z_5: sub -
0: z_6: add +
1: z_7: equal eq ==
1: z_8: eq equal ==
3: z_9: div /
3: z_{10}: div /
0: z_{11}: add +
2: z_{12}: mul *
1: z_{13}: sub -
3: z_{14}: mul *
3: z_{15}: mul

komischer weise hat es hier gereun
sonst hätte ich mir bei 18 einen
schelz erlaubt von vorne anzufangen
Diese Kombi benutze ich jetzt auch für
Zielregister (operant)...
und als ersten Quelloperant

Jamil Vajda (c) 09/25/2026

Seite 29 asm diagramm...

um den zweiten Quelloperanden
A zu nehmen nehme ich das
ganze noch mal anders herum.

Damit ich natürlich fest einen
Computer unter der Voraussetzung:

1.) arbeitsdatenspeicher
zu Register
2.) die befehle wirklich
nachgeladen ← pc/ip
3.) das nachladen aus ...
dem programmspeicher
ermöglicht spröünge ...

$r_3 \leftarrow r_3 / r_3$
$z_2 \leftarrow z_2 \cdot z_3$
$z_3 \leftarrow$

auch z_3 vergessen...

pause!!!

19:04

mükra electronic

SN74LS181 782820

Marke: No

€1,53

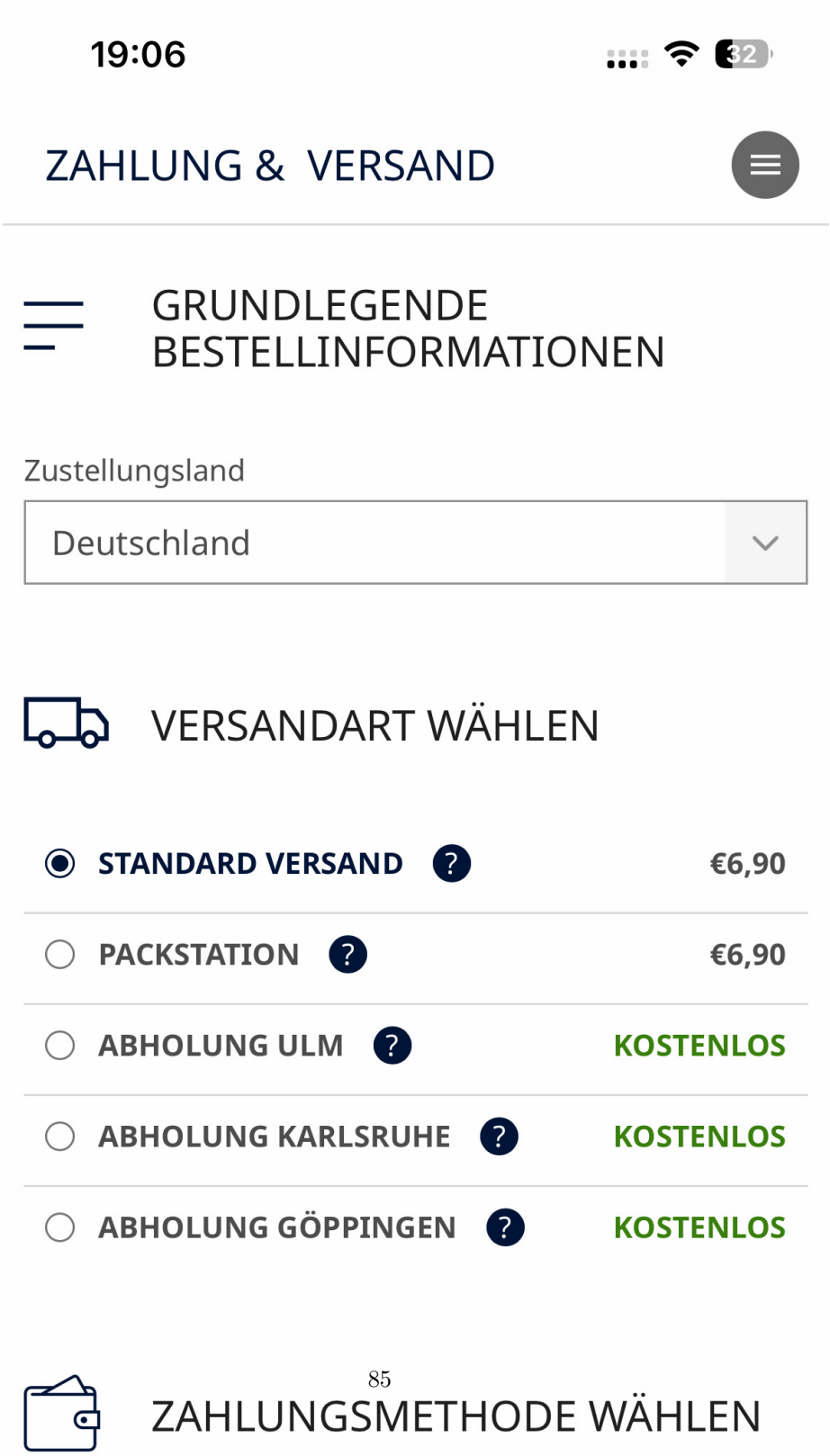
Lieferung bis: 31.08.2026

1

In den Warenkorb

TTL-Schaltung, Gehäuse: DIL24 4-Bit arithmetische / logische Einheit

Detaillierte Informationen


19:06
ZAHLUNG & VERSAND
GRUNDLEGENDE BESTELLINFORMATIONEN
Zustellungsland
Deutschland
VERSANDART WÄHLEN
STANDARD VERSAND €6,90
PACKSTATION €6,90
ABHOLUNG ULM KOSTENLOS
ABHOLUNG KARLSRUHE KOSTENLOS
ABHOLUNG GÖPPINGEN KOSTENLOS
ZAHLUNGSMETHODE WÄHLEN
BANKÜBERWEISUNG KOSTENLOS

19:25

mükra electronic

SN74LS181 782820

Marke: No

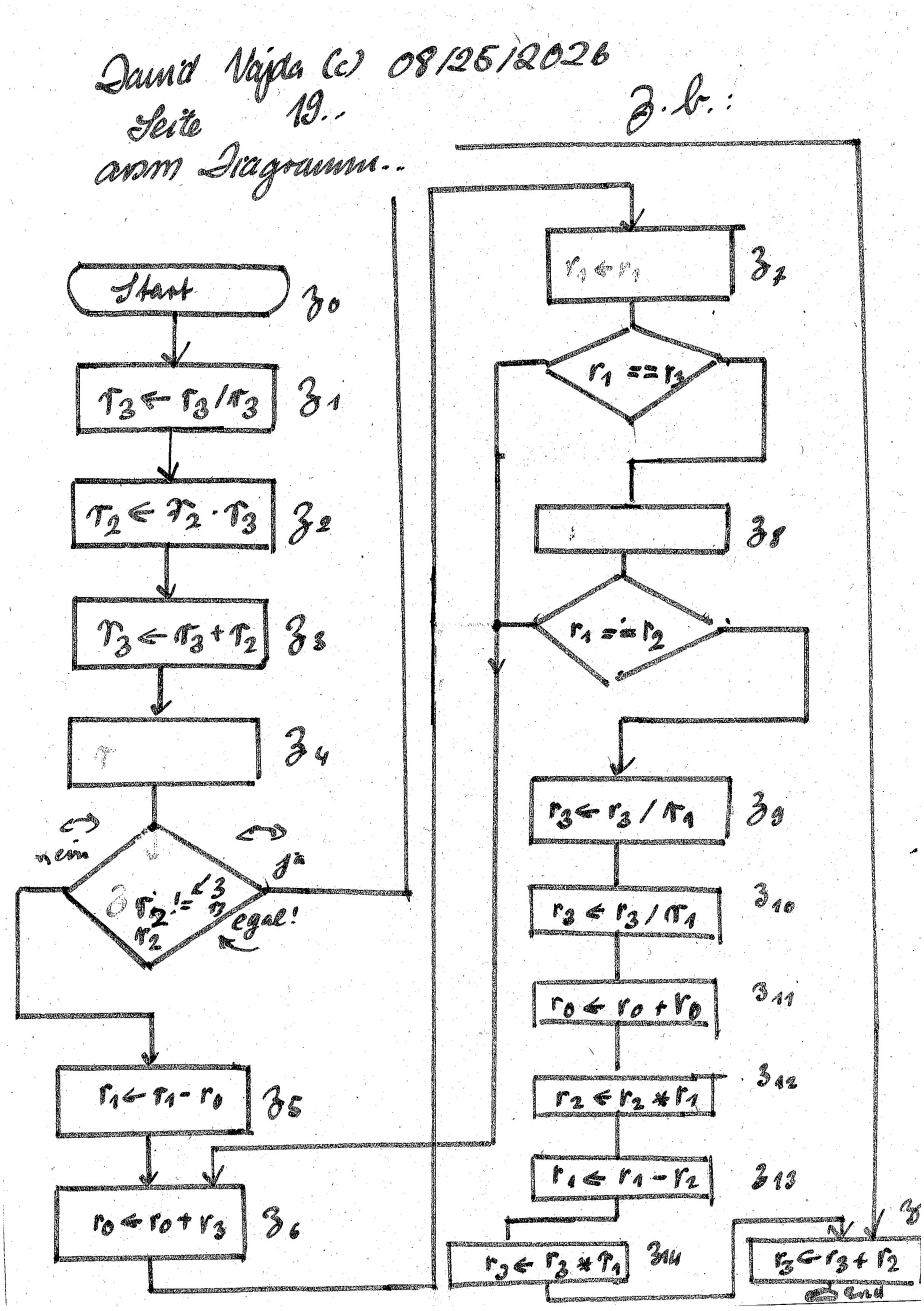
€1,53

Lieferung bis: 31.08.2026

1 In den Warenkorb

TTL-Schaltung, Gehäuse: DIL24 4-Bit arithmetische / logische Einheit

Detaillierte Informationen


David Vajda (c) 08/25/2026
Seite 19..
zum Diagramm..
Z.b.:
Start
Z0
r3 ← r3 / r3
Z1
r2 ← r2 · r3
Z2
r3 ← r3 + r2
Z3
Z4
nein
ja
r2 != r3
r2
equal!
r1 ← r1 - r0
Z5
r0 ← r0 + r3
Z6
r1 ← r1
Z7
r1 == r3
Z8
r1 != r2
r3 ← r3 / r1
Z9
r3 ← r3 / r1
Z10
r0 ← r0 + r0
Z11
r2 ← r2 * r1
Z12
r1 ← r1 - r2
Z13
r3 ← r3 * r1
Z14
r3 ← r3 + r2
Ende

Daniel Vajda (c) 08/25/2026
Seite 29 asm diagramm..
um den zweiten Quelloperanden
A zu nehmen nehmae ich da
ganz noch mal anders herum.

Damit ich natürlich fest einen
Kompute unter der Voraussetzung:

1.) arbeitsdatenoperator
zu Register
2.) die befehle wirklich
nachgeladen ← pc/ip
3.) das nachladen aus ...
dem programmspeicher
ermöglicht spring ...

$r_3 \leftarrow r_3 / r_3$
$z_2 \leftarrow z_2 \cdot z_3$
$z_3 \leftarrow$
auch z_3 vergessen... Das darf nicht
bei pc z sondern
r!
pause!!!
$z_3 \leftarrow z_3 \cdot z_3$ ← tbl?
entscheidung! rtl
rtl wie
in asm Diagramm fernsehen
abreibweise
bzw. $R \subseteq rtl$
R untermenge..

6 und nun. . . jetzt kommt das schaltnetz

z_n	x	z_{n+1}^+	y	XOR
$\frac{0b1000}{0}$	0	$\frac{0b0100}{1}$	1	1
$\frac{0b1000}{1}$	0	$\frac{0b0100}{1}$	1	1
$\frac{0b1000}{0}$	1	$\frac{0b0010}{0}$	0	1
$\frac{0b1000}{1}$	1	$\frac{0b0010}{0}$	0	1
$\frac{0b0100}{0}$	0	$\frac{0b0010}{0}$	0	0
$\frac{0b0100}{0}$	1	$\frac{0b0000}{0}$	0	0
$\frac{0b0010}{0}$	0	$\frac{0b0000}{0}$	0	1
$\frac{0b0010}{1}$	0	$\frac{0b0000}{0}$	0	1
$\frac{0b0010}{0}$	1	$\frac{0b1000}{1}$	1	1
$\frac{0b0010}{1}$	1	$\frac{0b1000}{1}$	1	1
$\frac{0b0001}{1}$	0	$\frac{0b1000}{1}$	1	0
$\frac{0b0001}{1}$	1	$\frac{0b0100}{1}$	1	0

```
z3+ := (z0 and not x) or (z1 and x)
z2+ := (z3 and not x) or (z0 and x)
z1+ := (z3 and x) or (z2 and not x)
z0+ := (z2 and x) or (z1 and not x)

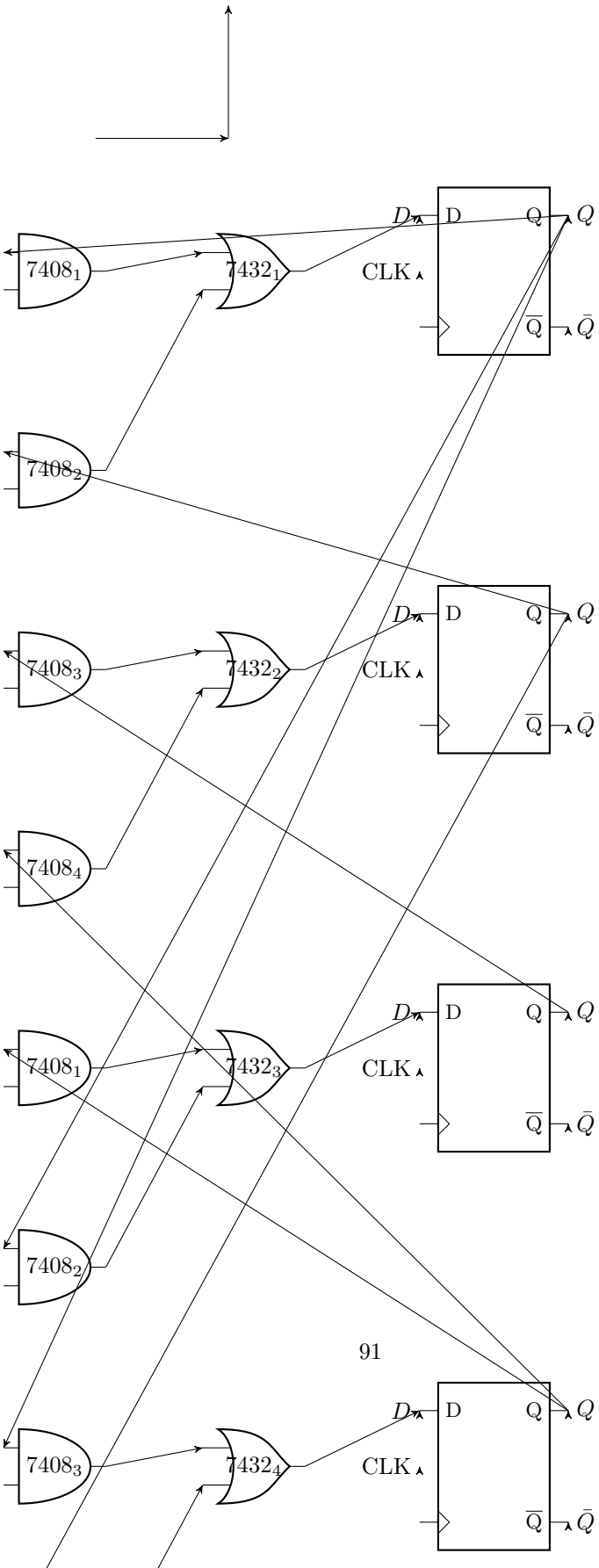
y := (z3 and not x) or (z1 and not x) or \\
 (z0 and not x) or (z0 and x)
```

in der tabelle war ein fehler: der zustand 1 muss nicht als 0 kodiert sein. ist es auch, blos leider nur im zielzustand, im ausgangszustand, ist es eins. im

z_n	x	z_{n+1}^+	y
0b1000	0	0b0100	1
0b1000	1	0b0010	0
0b0100	0	0b0010	0
0b0100	1	0b0001	0
0b0010	0	0b0001	0
0b0010	1	0b1000	1
0b0001	0	0b1000	1
0b0001	1	0b0100	1

zielzustand war es null

jetzt haben wir das schaltnetz, dann gucke ich, dass ich das als schaltnetz tue, und zwar: soll ich mit kaneda, ja oder nein? ich denke so oder so, ja, aber ich mache es trotzdem mit LaTeX. und zwar indem ich erst mal 8 UND Gatter hinmache 7408, oder wie, also nach IEEE-91/1984. ok: und: jetzt kommt: vier ODER Gatter, wir verbinden das, auch, dann kommt noch Kaneda. und dann 4 FF. wie von vorher übernommen.


7408$_1$
7432$_1$
7408$_2$
7408$_3$
7432$_2$
7408$_4$
7408$_1$
7432$_3$
7408$_2$
7408$_3$
7432$_4$
D
Q
Q
CLK
$\overline{\text{Q}}$
$\bar{Q}$

kurze zeichenübung - ist teil der sache:

```
\begin{tikzpicture}[->, >=stealth', node distance=3cm, auto]
\node [and port] (and00) {$7408_1$};
\node [and port, below of=and00] (and01) {$7408_2$};
\node [and port, below of=and01] (and02) {$7408_3$};
\node [and port, below of=and02] (and03) {$7408_4$};
\node [and port, below of=and03] (and04) {$7408_1$};
\node [and port, below of=and04] (and05) {$7408_2$};
\node [and port, below of=and05] (and06) {$7408_3$};
\node [and port, below of=and06] (and07) {$7408_4$};

\node [or port, right of=and00] (or00) {$7432_1$};
\node [or port, right of=and02] (or01) {$7432_2$};
\node [or port, right of=and04] (or02) {$7432_3$};
\node [or port, right of=and06] (or03) {$7432_4$};

\draw (and00.out) -- (or00.in 1);
\draw (and01.out) -- (or00.in 2);
\draw (and02.out) -- (or01.in 1);
\draw (and03.out) -- (or01.in 2);
\draw (and04.out) -- (or02.in 1);
\draw (and05.out) -- (or02.in 2);
\draw (and06.out) -- (or03.in 1);
\draw (and07.out) -- (or03.in 2);


    \node[flipflop D, right of=or00] (D00) {};
    \draw (D00.pin 1) node[left] {$D$} ;
    \draw (D00.pin 2) node[left] {CLK};
    \draw (D00.pin 6) node[right] {$Q$};
    \draw (D00.pin 4) node[right] {$\bar{Q}$};


        \node[flipflop D, right of=or01] (D01) {};
    \draw (D01.pin 1) node[left] {$D$} ;
    \draw (D01.pin 2) node[left] {CLK};
    \draw (D01.pin 6) node[right] {$Q$};
    \draw (D01.pin 4) node[right] {$\bar{Q}$} {};


        \node[flipflop D, right of=or02] (D02) {};
    \draw (D02.pin 1) node[left] {$D$} ;
    \draw (D02.pin 2) node[left] {CLK};
    \draw (D02.pin 6) node[right] {$Q$};
    \draw (D02.pin 4) node[right] {$\bar{Q}$};
```

```
        \node[flipflop D, right of=or03] (D03) {};
    \draw (D03.pin 1) node[left] {$D$} ;
    \draw (D03.pin 2) node[left] {CLK};
    \draw (D03.pin 6) node[right] {$Q$};
    \draw (D03.pin 4) node[right] {$\bar{Q}$};

\draw (or00.out) -- (D00.pin 1);
\draw (or01.out) -- (D01.pin 1);
\draw (or02.out) -- (D02.pin 1);
\draw (or03.out) -- (D03.pin 1);


\draw (D00.pin 6) -- (and06.in 1);
\draw (D01.pin 6) -- (and07.in 1);

\draw (D03.pin 6) -- (and04.in 1);
\draw (D00.pin 6) -- (and05.in 1);

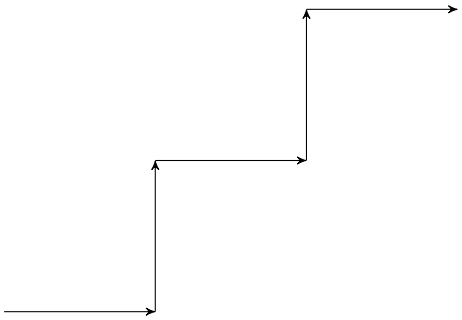
\draw (D03.pin 6) -- (and03.in 1);
\draw (D02.pin 6) -- (and02.in 1);

\draw (D01.pin 6) -- (and01.in 1);
\draw (D00.pin 6) -- (and00.in 1);

\draw (0,2) -- (2,2);
\draw (2,2) -- (2,4);

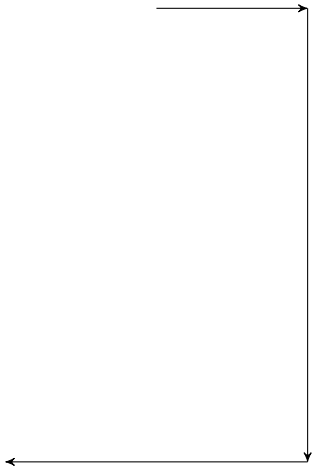


\end{tikzpicture}
```



Schnecke:

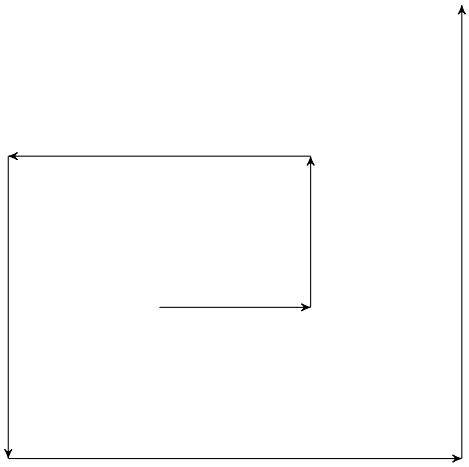
$$((n,m),(n+2,m),(n+2,m-4),(n-4,m-4),(n-4,m+8),\dots))$$



nein:

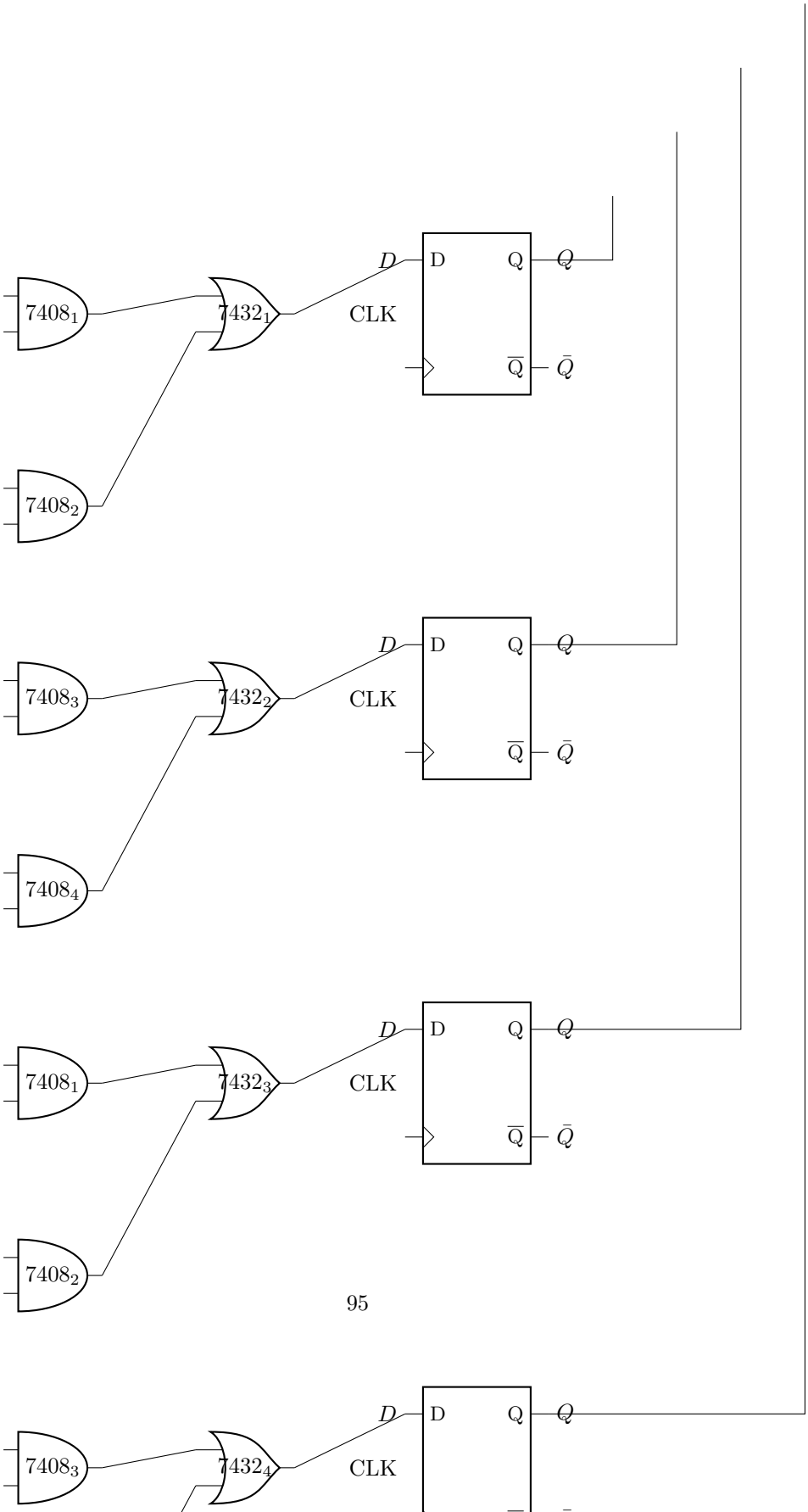
Schnecke:

$$((n,m),(n+2,m),(n+2,m+2),(n-4,m+2),\dots))$$



Es wäre interessant eine (vielleicht)

1. Schnecke mit metrisch (in anführungsstrichen) gleich bleibenden längen
2. grösser werden


7408$_1$
7408$_2$
7432$_1$
D
CLK
D
Q
Q
$\overline{\text{Q}}$
$\bar{Q}$
7408$_3$
7408$_4$
7432$_2$
D
CLK
D
Q
Q
$\overline{\text{Q}}$
$\bar{Q}$
7408$_1$
7408$_2$
7432$_3$
D
CLK
D
Q
Q
$\overline{\text{Q}}$
$\bar{Q}$
7408$_3$
7408$_4$
7432$_4$
D
CLK
D
Q
Q
$\overline{\text{Q}}$
$\bar{Q}$

```
\begin{tikzpicture}[node distance=3cm, auto]
\node [and port] (and00) {$7408_1$};
\node [and port, below of=and00] (and01) {$7408_2$};
\node [and port, below of=and01] (and02) {$7408_3$};
\node [and port, below of=and02] (and03) {$7408_4$};
\node [and port, below of=and03] (and04) {$7408_1$};
\node [and port, below of=and04] (and05) {$7408_2$};
\node [and port, below of=and05] (and06) {$7408_3$};
\node [and port, below of=and06] (and07) {$7408_4$};

\node [or port, right of=and00] (or00) {$7432_1$};
\node [or port, right of=and02] (or01) {$7432_2$};
\node [or port, right of=and04] (or02) {$7432_3$};
\node [or port, right of=and06] (or03) {$7432_4$};

\draw (and00.out) -- (or00.in 1);
\draw (and01.out) -- (or00.in 2);
\draw (and02.out) -- (or01.in 1);
\draw (and03.out) -- (or01.in 2);
\draw (and04.out) -- (or02.in 1);
\draw (and05.out) -- (or02.in 2);
\draw (and06.out) -- (or03.in 1);
\draw (and07.out) -- (or03.in 2);


    \node[flipflop D, right of=or00] (D00) {};
    \draw (D00.pin 1) node[left] {$D$} ;
    \draw (D00.pin 2) node[left] {CLK};
    \draw (D00.pin 6) node[right] {$Q$};
    \draw (D00.pin 4) node[right] {$\bar{Q}$};


        \node[flipflop D, right of=or01] (D01) {};
    \draw (D01.pin 1) node[left] {$D$} ;
    \draw (D01.pin 2) node[left] {CLK};
    \draw (D01.pin 6) node[right] {$Q$};
    \draw (D01.pin 4) node[right] {$\bar{Q}$} {};

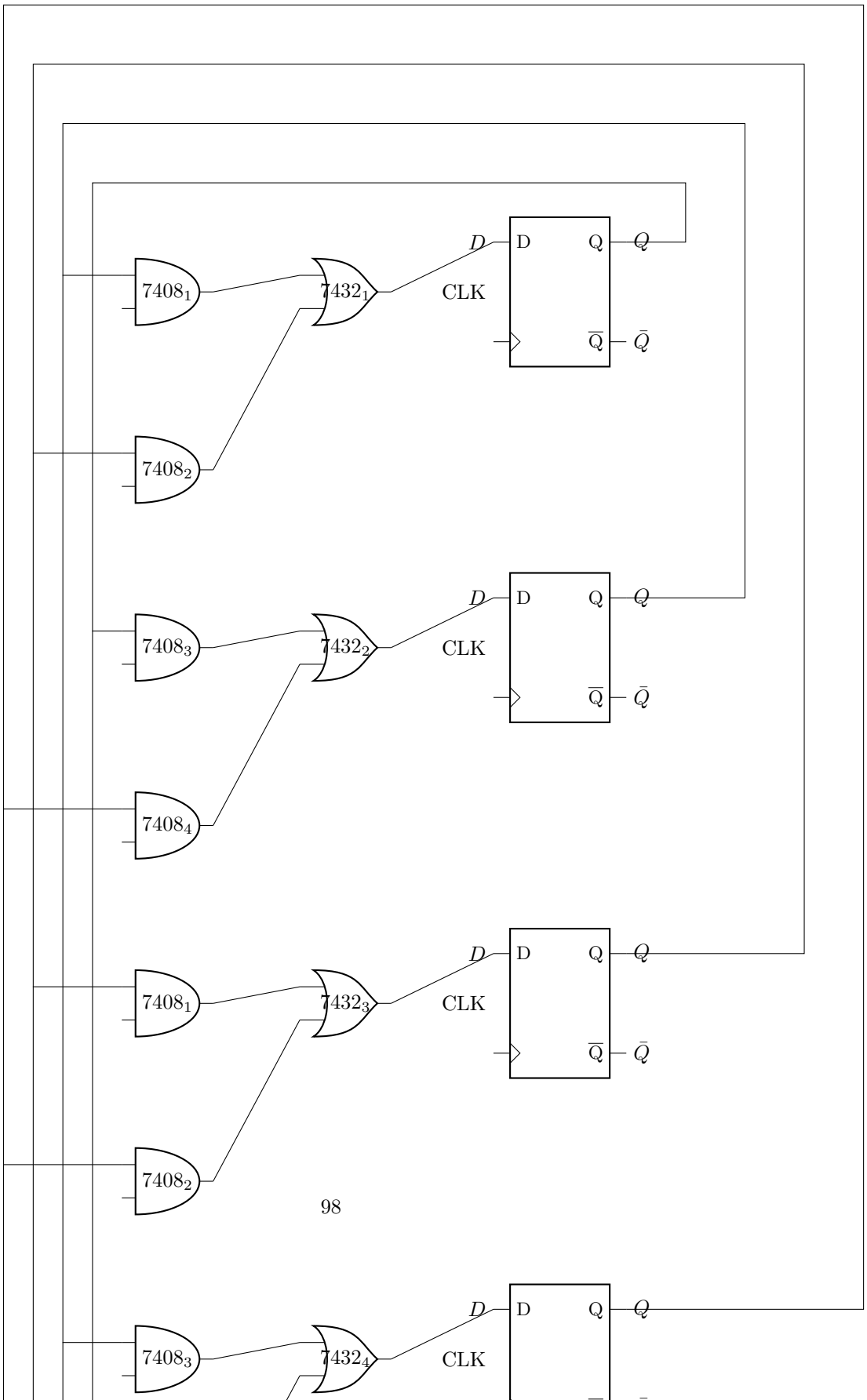

        \node[flipflop D, right of=or02] (D02) {};
    \draw (D02.pin 1) node[left] {$D$} ;
    \draw (D02.pin 2) node[left] {CLK};
    \draw (D02.pin 6) node[right] {$Q$};
    \draw (D02.pin 4) node[right] {$\bar{Q}$};
```

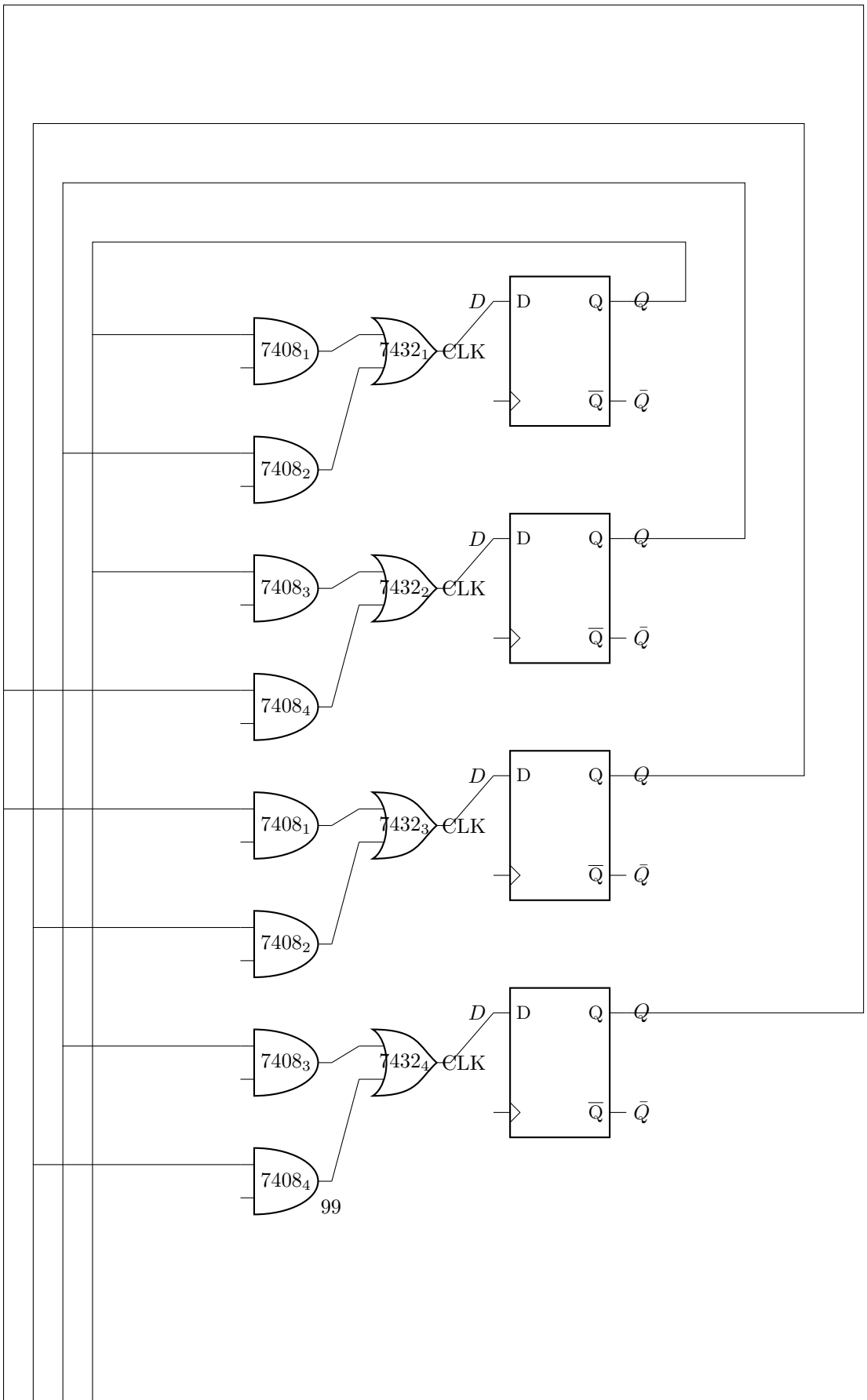
```
        \node[flipflop D, right of=or03] (D03) {};
    \draw (D03.pin 1) node[left] {$D$} ;
    \draw (D03.pin 2) node[left] {CLK};
    \draw (D03.pin 6) node[right] {$Q$};
    \draw (D03.pin 4) node[right] {$\bar{Q}$};

\draw (or00.out) -- (D00.pin 1);
\draw (or01.out) -- (D01.pin 1);
\draw (or02.out) -- (D02.pin 1);
\draw (or03.out) -- (D03.pin 1);

\draw (D00.pin 6) -- +(1,0) -- +(1,1);
\draw (D01.pin 6) -- +(2,0) -- +(2,8);
\draw (D02.pin 6) -- +(3,0) -- +(3,15);
\draw (D03.pin 6) -- +(4,0) -- +(4,22);

\end{tikzpicture}
```


7408$_1$
7432$_1$
7408$_2$
7408$_3$
7432$_2$
7408$_4$
7408$_1$
7432$_3$
7408$_2$
7408$_3$
7432$_4$
D
CLK
Q
$\overline{\text{Q}}$
Q
$\bar{Q}$


7408$_1$
7432$_1$
CLK
D
D
Q
Q
$\overline{\text{Q}}$
$\bar{Q}$
7408$_2$
7408$_3$
7432$_2$
CLK
D
D
Q
Q
$\overline{\text{Q}}$
$\bar{Q}$
7408$_4$
7408$_1$
7432$_3$
CLK
D
D
Q
Q
$\overline{\text{Q}}$
$\bar{Q}$
7408$_2$
7408$_3$
7432$_4$
CLK
D
D
Q
Q
$\overline{\text{Q}}$
$\bar{Q}$
7408$_4$

```
z3+ := (z0 and not x) or (z1 and x)
z2+ := (z3 and not x) or (z0 and x)
z1+ := (z3 and x) or (z2 and not x)
z0+ := (z2 and x) or (z1 and not x)

y := (z3 and not x) or (z1 and not x) or \\
 (z0 and not x) or (z0 and x)

\begin{tikzpicture}[node distance=2cm, auto]
\node [and port] (and00) {$7408_1$};
\node [and port, below of=and00] (and01) {$7408_2$};
\node [and port, below of=and01] (and02) {$7408_3$};
\node [and port, below of=and02] (and03) {$7408_4$};
\node [and port, below of=and03] (and04) {$7408_1$};
\node [and port, below of=and04] (and05) {$7408_2$};
\node [and port, below of=and05] (and06) {$7408_3$};
\node [and port, below of=and06] (and07) {$7408_4$};

\node [or port, right of=and00] (or00) {$7432_1$};
\node [or port, right of=and02] (or01) {$7432_2$};
\node [or port, right of=and04] (or02) {$7432_3$};
\node [or port, right of=and06] (or03) {$7432_4$};

\draw (and00.out) -- (or00.in 1);
\draw (and01.out) -- (or00.in 2);
\draw (and02.out) -- (or01.in 1);
\draw (and03.out) -- (or01.in 2);
\draw (and04.out) -- (or02.in 1);
\draw (and05.out) -- (or02.in 2);
\draw (and06.out) -- (or03.in 1);
\draw (and07.out) -- (or03.in 2);


    \node[flipflop D, right of=or00] (D00) {};
    \draw (D00.pin 1) node[left] {$D$} ;
    \draw (D00.pin 2) node[left] {CLK};
    \draw (D00.pin 6) node[right] {$Q$};
    \draw (D00.pin 4) node[right] {$\bar{Q}$};


        \node[flipflop D, right of=or01] (D01) {};
    \draw (D01.pin 1) node[left] {$D$} ;
    \draw (D01.pin 2) node[left] {CLK};
    \draw (D01.pin 6) node[right] {$Q$};
    \draw (D01.pin 4) node[right] {$\bar{Q}$} {};
```

```
        \node[flipflop D, right of=or02] (D02) {};
    \draw (D02.pin 1) node[left] {$D$} ;
    \draw (D02.pin 2) node[left] {CLK};
    \draw (D02.pin 6) node[right] {$Q$};
    \draw (D02.pin 4) node[right] {$\bar{Q}$};


        \node[flipflop D, right of=or03] (D03) {};
    \draw (D03.pin 1) node[left] {$D$} ;
    \draw (D03.pin 2) node[left] {CLK};
    \draw (D03.pin 6) node[right] {$Q$};
    \draw (D03.pin 4) node[right] {$\bar{Q}$};

\draw (or00.out) -- (D00.pin 1);
\draw (or01.out) -- (D01.pin 1);
\draw (or02.out) -- (D02.pin 1);
\draw (or03.out) -- (D03.pin 1);

\draw (D00.pin 6) -- +(1,0) -- +(1,1) -- +(-9,1) -- +(-9,-22);
\draw (D01.pin 6) -- +(2,0) -- +(2,6) -- +(-9.5,6) -- +(-9.5,-20);
\draw (D02.pin 6) -- +(3,0) -- +(3,11) -- +(-10,11) -- +(-10,-18);
\draw (D03.pin 6) -- +(4,0) -- +(4,17) -- +(-10.5,17) -- +(-10.5,-17);

\draw (and07.in 1) -- +(-3.5,0);
\draw (and06.in 1) -- +(-3,0);

\draw (and05.in 1) -- +(-3.5,0);
\draw (and04.in 1) -- +(-4,0);

\draw (and03.in 1) -- +(-4,0);
\draw (and02.in 1) -- +(-2.5,0);

\draw (and01.in 1) -- +(-3.0,0);
\draw (and00.in 1) -- +(-2.5,0);
\end{tikzpicture}
```

7 quine mc cluskey

```
(C) David Vajda
Sun Aug 30 19:25:20 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    0
  5 0 1 0 1    0
  6 0 1 1 0    0
  7 0 1 1 1    1
  8 1 0 0 0    1
  9 1 0 0 1    1
10 1 0 1 0    1
11 1 0 1 1    1
12 1 1 0 0    1
13 1 1 0 1    1
14 1 1 1 0    0
15 1 1 1 1    0
```

n	x_3	x_2	x_1	x_0	y
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	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

```
(C) David Vajda
Sun Aug 30 19:25:20 2026
4 Network - TTL - Disjunktive Normalform

  1 0 0 0 1    1
  3 0 0 1 1    1
  7 0 1 1 1    1
  8 1 0 0 0    1
  9 1 0 0 1    1
10 1 0 1 0    1
11 1 0 1 1    1
12 1 1 0 0    1
13 1 1 0 1    1
```

```
Gruppe 1:
  1 0 0 0 1    1
  8 1 0 0 0    1
Gruppe 2:
  3 0 0 1 1    1
  9 1 0 0 1    1
10 1 0 1 0    1
12 1 1 0 0    1
Gruppe 3:
  7 0 1 1 1    1
11 1 0 1 1    1
13 1 1 0 1    1

1:3  0 0 - 1
1:8  - 0 0 1
8:9  1 0 0 -
8:10  1 0 - 0
8:12  1 - 0 0

3:7  0 - 1 1
3:11  - 0 1 1
9:11  1 0 - 1
9:13  1 - 0 1
10:7  1 0 1 -
12:13 1 1 0 -


1:3  0 0 x 1
1:8  x 0 0 1
8:9  1 0 0 x
8:10  1 0 x 0
8:12  1 x 0 0

3:7  0 x 1 1
3:11  x 0 1 1
9:11  1 0 x 1
9:13  1 x 0 1
10:7  1 0 1 x
12:13 1 1 0 x



1:8  x 0 0 1
3:11  x 0 1 1
8:12  1 x 0 0
3:7  0 x 1 1
```

```
9:13  1 x 0 1
9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1
10:7  1 0 1 x
12:13 1 1 0 x
8:9  1 0 0 x



1:8  x 0 0 1
3:11  x 0 1 1

8:12  1 x 0 0
3:7  0 x 1 1
9:13  1 x 0 1

9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1

10:7  1 0 1 x
12:13 1 1 0 x
8:9  1 0 0 x



1:8  x 0 0 1
3:11  x 0 1 1
1:8:3:11  x 0 x 1

Gruppe 1:
8:12  1 x 0 0
Gruppe 2:
9:13  1 x 0 1
3:7  0 x 1 1

8:12   1 x 1 1
8:12:9:13  1 x 0 x


9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1

9:11:8:10  1 0 x x
```

```
9:13:1:3  - 0 x 1

8:9  1 0 0 x
10:7  1 0 1 x
12:13 1 1 0 x

8:9:10:7  1 0 x x
8:9:12:13  1 x 0 x



1:8:3:11  x 0 x 1
8:12   1 x 1 1
8:12:9:13  1 x 0 x
9:11:8:10  1 0 x x
9:13:1:3  x 0 x 1??
8:9:10:7  1 0 x x
8:9:12:13  1 x 0 x
```

gut ein wenig weiter, standard übung

```
punkt, linie, zoll = finger, handbreit, fuss
punkt, linie, zoll = finger, handbreit, fuss
punkt, linie, zoll = finger, handbreit, fuss
punkt, linie, zoll = finger, handbreit, fuss
elle, klafter, lachter, meile
elle, klafter, lachter, meile
elle, klafter, lachter, meile
elle, klafter, lachter, meile

punkt = 0.35mm
punkt = 0.35mm
punkt = 0.35mm

1pt = 1 dot = 0.35mm
1pt = 1 dot (??) = 0.35mm

# sorry nur ein wenig uebung

1 pt = 0.35mm
1 pt = 0.35mm

punkt, linie, zoll = finger, handbreit, fuss
```

```
1 zoll = 1 inch, ein zeigefinger spitze = 2.5cm bei david vajda (c)
1 zoll = 1 inch

punkt, linie, zoll = finger, handbreit, fuss
1 fuss = 3 zoll
1 fuss = 3 zoll
1 elle = 2 fuss
1 elle = 2 fuss
... hier herrscht unklarheit, gleich lernen.

cent, gran, karat, skrupel, heller, pfennig, mark, pfund, unze, centner
cent, gran, karat, skrupel, heller, pfennig, mark, pfund, unze, centner
cent, gran, karat, skrupel, heller, pfennig, mark, pfund, unze, centner

1 pfund = 2 mark
1 pfund = 2 mark

1pfd = 1 pfund
stearling silber: 785, 885, ...

cent, gran, karat, skrupel, heller, pfennig, mark, pfund, unze, centner
```

8 das wurde geübt

nächster teil. es ging um die zufallszahlen in der y-Reihe. in der Senkrechten.
bisher habe ich das wuerde sich von der wahrscheinlichkeit eines zufallsexperiments, genau gleich verhalten, wobei hier kommen vielleicht besonderheiten der Zufallszahlen zustande.

so: ich habe jede 0 und 1 in der senkrechten y einzeln zufällig erzeugt.

trotzdem kann man ja 16 Bit werte nehmen, worte. so, das kann man erzeugen, und als y nehmen.

wenn wir das haben, dann: haben wir die senkrechte als wert. aber, auch wir können vielleicht matrizen,
als: mit der Waagerechten und Senkrechten machen, aber verschiedene.
ok: deswegen lohnt sich umrechnung, hier auch die codes.

trotzdem bevorzuge ich: jetzt erst quine mc cluskey standard übung zu machen, dann auch

```
abel ... galasm
```

und dann geht es drum speicherbausteine usw. zu machen. so jetzt weiter. trotzdem erst diese zufallszahlen und das ganze ding.

```
(C) David Vajda
Sun Aug 30 19:25:20 2026
4 Network - TTL - Disjunktive Normalform

  1 0 0 0 1    1
  3 0 0 1 1    1
  7 0 1 1 1    1
  8 1 0 0 0    1
  9 1 0 0 1    1
10 1 0 1 0    1
11 1 0 1 1    1
12 1 1 0 0    1
13 1 1 0 1    1

Gruppe 1:
  1 0 0 0 1    1
  8 1 0 0 0    1
Gruppe 2:
  3 0 0 1 1    1
  9 1 0 0 1    1
10 1 0 1 0    1
12 1 1 0 0    1
Gruppe 3:
  7 0 1 1 1    1
11 1 0 1 1    1
13 1 1 0 1    1

1:3  0 0 - 1
1:8  - 0 0 1
8:9  1 0 0 -
8:10  1 0 - 0
8:12  1 - 0 0

3:7  0 - 1 1
3:11  - 0 1 1
9:11  1 0 - 1
9:13  1 - 0 1
10:11 1 0 1 -
12:13 1 1 0 -

1:3  0 0 x 1
1:8  x 0 0 1
8:9  1 0 0 x
8:10  1 0 x 0
8:12  1 x 0 0
```

```
3:7  0 x 1 1
3:11  x 0 1 1
9:11  1 0 x 1
9:13  1 x 0 1
10:11 1 0 1 x
12:13 1 1 0 x



1:8  x 0 0 1
3:11  x 0 1 1
8:12  1 x 0 0
3:7  0 x 1 1
9:13  1 x 0 1
9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1
10:11 1 0 1 x
12:13 1 1 0 x
8:9  1 0 0 x



1:8  x 0 0 1
3:11  x 0 1 1

8:12  1 x 0 0
3:7  0 x 1 1
9:13  1 x 0 1

9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1

10:7  1 0 1 x
12:13 1 1 0 x
8:9  1 0 0 x



1:8  x 0 0 1
3:11  x 0 1 1
1:8:3:11  x 0 x 1

Gruppe 1:
8:12  1 x 0 0
```

```
Gruppe 2:
9:13  1 x 0 1
3:7  0 x 1 1

8:12   1 x 1 1
8:12:9:13  1 x 0 x


9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1

9:11:8:10  1 0 x x
9:13:1:3  - 0 x 1

8:9  1 0 0 x
10:7  1 0 1 x
12:13 1 1 0 x

8:9:10:7  1 0 x x
8:9:12:13  1 x 0 x



1:8:3:11  x 0 x 1
8:12   1 x 1 1
8:12:9:13  1 x 0 x
9:11:8:10  1 0 x x
9:13:1:3  x 0 x 1
8:9:10:7  1 0 x x
8:9:12:13  1 x 0 x
```

zwischenübung: minix

```
kernel, mm, fs, (h)

./kernel/proc.c
./kernel/main.c
./kernel/mpx88.s
./kernel/klib88.s
./kernel/memory.c
./kernel/stty.c
./kernel/floppy.c
./kernel/clock.c
./kernel/disk ?? wini...c
```

```
./h/glo.h
./h/const.h
./h/type.h

.. (proc.??nein)

ok:

virtual
physisch/physical/physikalisch

clicks
bytes

vir
phys
clicks
bytes

vir, phys, clicks, bytes

map
vir
phys
len

vir phys len

mem_map
mem_vir
mem_phys
mem_len

struct mem_map {
 vir_clicks mem_vir;
 phys_clicks mem_phys;
 vir_clicks mem_len;
};

NR_REGS
NR_SEGS

11 REGS: ax bx dx cx sp bp si di cs ds ss

T: Text (Code): cs
```

```
D: Data: ds
S: Stack: ss

PIC 8259A: Programmable Interrupt Controller 8259A

Port: 0x20, 0x21, 0xA0, 0xA1 (0xa0, 0xa1):

0x20: Port
0x21: Mask
0xA0: Port
0xA1: Mask

X: Gift: 13: alt: wg. untere Nummer
X passt zu 13
XT: alter IBM PC XT

AT: neuer
AT: advanced...

#define XT_WINI_INT ?? 13
CLOCK: 8
KEYBOARD: 9

SYS_INT, ja

XT_WINI_INT
KEYBOARD_INT
CLOCK_INT
TTY_INT??
FLOPPY_INT...

trap divide interrupt

divide: abbildung auf den 0 vektor, eintrag in der interrupt vektortabelle, mit index 0 in d
trap: alle interrupts > 15, bedeutet: alle: die nicht am PIC 8259A sind
interrupt <= 15

suprise, trap, divide:
suprise: interrupt nicht erwartet

ok: botschaften:

sys_call: die einzigen system_calls die ausgetauscht werden sind:
SENDEN und EMPFANGEN, nachher in der
./kernel/proc.c?? nachher nachdenken
```

```
ok:
mini_send und mini_rec (interrupt und: sys_call)

in der
kernel.h?? oder proc.h <- zweiteres ist die

proc structure:

struct proc {

} procs [NR_TASKS+NR_PROCS];


darein kommt:

int p_regs [NR_REGS];
extern struct mem_map p_map [NR_SEGS];

usertime und systime
realtime time clocks ticks

jedenfalls:
usertime, system time vom prozess
vom aufgerufenen kind

stack pointer, anfang, grenze
int pid

process id;

..

also:

extern struct proc {
 extern int p_regs [NR_REGS];
 extern struct mem_map [NR_SEGS];
 int pid;
 // stackpointer, times...
} procs [NR_TASKS+NR_PROCS];

gut:
./kernel/main.c

grosse routine: main:
MM, FS, INIT
```

```
Der Kernel selber wird durch INIT eher repraesentiert, was weitere prozesse startet

TASKS: Betriebssystem
PROCS: User

schleife:

und: einsprungsaddressen:

keyboard_int, clock_int, floppy_int, disk_int
sys_call divide trap ....

erst: in der schleife die segmente:
T, D, S: anfang und laenge
und weitere...

dann interrupts initialisieren

set_vec (

set_vec (keyboard_int, ...)

ok:

mpx88.s:
klib88.s

mpx88.s

MINIX:
TTY_INT:
CLOCK_INT:
...

klib88.s:
hilfsroutinen:

dma_read
dma_write
lock
unlock
getchrome
video buffer copy...

proc.c:
```

mini_rec, mini_send, sys_call, interrupt

bei dem einen, werden processe gleich mit aufgebaut
bei dem anderen, nur die botschaften gesendet

diese sind caller, empfaenger und botschaft

dann kommen:
ready, unready, schedule...

ok... ich lerne heute mal auswendig, dann lerne ich das
spaeter weiter...
egal, wie vollstaendig, noch der ARM Cortex

ARM: Advanced Risc Machine

Advanced Risc Machine ldt.

ldt.: gmbh:

Linien:
A: Operating System, ... mit allem
R: Embedded System
M: Microcontroller

ARM Cortex-M7
ARM Cortex-A7
ARM Cortex-R7

ok: Dann Versionen:

zunaechst: ARM7TDMI ... what ever:

ARM: Architecture Application Level Reference Manual
TRM: technical Reference Manual

ARM, ARM, ARM

ISA: Immer: Instruction Set Architecture

ARM: Advanced Risc Machine
ARM: Architecture Application Level Reference Manual
ARM: Operating System, Embbeded, Microcontroller

Das sind Profiles

ok: Waynes?

ok:
Apple Computer, Arcorn und: VLSI Technology

VLSI: Very Large Scale Integration

VUSE: Very Ultra Super Extra

Small Medium Large

Memory Modell: x86: Tiny, Small, Medium, Compact, Large, Huge, Flat ...

Frequency: Low, Medium, High Frequency
Very High, Ultra High, Super High, Extra High

Very Large, Ultra Large...

ok: Very Large Scal Intergration Technology

Apple Computer, Arcorn und: VLSI Technology: ARM

STMicroelectronics: Atmel (heute)

dazu: PHILLIPS (PX...?)
OKI, ...

ARM Cortex-M7
ok: v4, v5, v6, v7

Thumb: daumen

bei dem prozessor:
Thumb-2: Erweiterter Befehlssatz: 32 Bit
Thumb-Instructions: 16 Bit...

ok:

Graph:
(auf Gut Deutsch: User und System)

User-Thread, Kernel...
ok: wie rein und raus:... gerade nicht praesent...

ok:
16 Register

```
13 GPR: General Purpose Register

r13 ... r0
Stack: Main Stack Pointer, Process Stack Pointer
Link Register: Ruecksprungaddresse
... noch was...

es gibt befehle, wie
ADD, ADC,
B: Branch, BC?? wie carry

flags sind generell: wie x86
sign, carry, overflow... (aehnlich)

atmega8 nebenbei gibt es ja noch neben
SIGN auch NEGATIVE ...
ok:

dann gibt es:
3 Formen der... Interrupts, oder wie:

jedenfalls generell:

NMI: Non Maskable Interrupts
Siehe: PIC 8259A, 0x21, 0xa1

Nested Vectored Interrupt

Verschachtetele Interrupt Routinen, oder der Vektor??

dann gibt es:

ABI: Application Binary Interface
EABI: Enhanced ABI

Advanced Periphal Bus
Advanced High Perphormance Bus
Advanced

Port, Bus:

Serial Wire: Seriell (V.24, RS232... z.B.)

JTAG: Joint Test Action Group

Debug Schnittstelle: JTAG...
```

```
Debug Port
JTAG Debug Port...



Es gibt mehrere Einheiten:

MMU: Memory Management Unit
MPU: Memory Protection Unit
LSU: Load and Store Unit
ALU: Arithmetic and Logic Unit
Power Management Unit

Neben, Neumann/harward...:
I/O, CPU und Mem:

Peripherals, auf dem IC, Debug Interface, what ever...

ok: pause, ich mache aufgaben weiter...
```

aufgaben:

- was muss man noch lernen: z.B. die TTL Gatter, mit Kaneda, am Besten, das standard so ungefaehr drin sitzt
- dann jetzt codes auch: CRC und so weiter... und: vorher, das fertig.
- ja: und binaeres dividieren...: das von neulich aus excel auf den anderen pc, neue aufgabe, 16 Bit Wort ausgeben und ausrechnen, erneut

```
(C) David Vajda
Sun Aug 30 19:25:20 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    0
 5 0 1 0 1    0
 6 0 1 1 0    0
 7 0 1 1 1    1
 8 1 0 0 0    1
 9 1 0 0 1    1
10 1 0 1 0    1
```

```
11 1 0 1 1    1
12 1 1 0 0    1
13 1 1 0 1    1
14 1 1 1 0    0
15 1 1 1 1    0

(C) David Vajda
Sun Aug 30 19:25:20 2026
4 Network - TTL - Disjunktive Normalform

  1 0 0 0 1    1
  3 0 0 1 1    1
  7 0 1 1 1    1
  8 1 0 0 0    1
  9 1 0 0 1    1
10 1 0 1 0    1
11 1 0 1 1    1
12 1 1 0 0    1
13 1 1 0 1    1

Gruppe 1:
  1 0 0 0 1    1
  8 1 0 0 0    1
Gruppe 2:
  3 0 0 1 1    1
  9 1 0 0 1    1
10 1 0 1 0    1
12 1 1 0 0    1
Gruppe 3:
  7 0 1 1 1    1
11 1 0 1 1    1
13 1 1 0 1    1

1:3  0 0 - 1
1:8  - 0 0 1
8:9  1 0 0 -
8:10  1 0 - 0
8:12  1 - 0 0

3:7  0 - 1 1
3:11  - 0 1 1
9:11  1 0 - 1
9:13  1 - 0 1
10:11 1 0 1 -
12:13 1 1 0 -
```

```
1:3  0 0 x 1
1:8  x 0 0 1
8:9  1 0 0 x
8:10  1 0 x 0
8:12  1 x 0 0

3:7  0 x 1 1
3:11  x 0 1 1
9:11  1 0 x 1
9:13  1 x 0 1
10:11 1 0 1 x
12:13 1 1 0 x



1:8  x 0 0 1
3:11  x 0 1 1
8:12  1 x 0 0
3:7  0 x 1 1
9:13  1 x 0 1
9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1
10:11 1 0 1 x
12:13 1 1 0 x
8:9  1 0 0 x



1:8  x 0 0 1
3:11  x 0 1 1

8:12  1 x 0 0
3:7  0 x 1 1
9:13  1 x 0 1

9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1

10:7  1 0 1 x
12:13 1 1 0 x
8:9  1 0 0 x
```

```
1:8  x 0 0 1
3:11  x 0 1 1
1:8:3:11  x 0 x 1

Gruppe 1:
8:12  1 x 0 0
Gruppe 2:
9:13  1 x 0 1
3:7  0 x 1 1

8:12   1 x 1 1
8:12:9:13  1 x 0 x


9:11  1 0 x 1
8:10  1 0 x 0
1:3  0 0 x 1

9:11:8:10  1 0 x x
9:13:1:3  - 0 x 1

8:9  1 0 0 x
10:7  1 0 1 x
12:13 1 1 0 x

8:9:10:7  1 0 x x
8:9:12:13  1 x 0 x




1:8:3:11  x 0 x 1 (hier ist was komisch)
8:12   1 x 1 1
8:12:9:13  1 x 0 x
9:11:8:10  1 0 x x
9:13:1:3  x 0 x 1 (hier ist was komisch), ok, es war eher das:
8:9:10:7  1 0 x x
8:9:12:13  1 x 0 x

9:13  1 x 0 1, ok, war rechtschreibfehler, gruppe (2 und 1)
9:13 und: 8:12, dann: selber ausdruck


1:8:3:11  x 0 x 1
8:12   1 x 1 1
8:12:9:13  1 x 0 x
```

```
9:11:8:10  1 0 x x
8:9:10:7  1 0 x x
8:9:12:13  1 x 0 x


  1 0 0 0 1    1
  3 0 0 1 1    1
  7 0 1 1 1    1
  8 1 0 0 0    1
  9 1 0 0 1    1
10 1 0 1 0    1
11 1 0 1 1    1
12 1 1 0 0    1
13 1 1 0 1    1

   1 3 7 8 9 10 11 12 13
1:8:3:11  1 1  1   1
8:12      1    1
8:12:9:13     1 1   1 1
9:11:8:10     1 1 1 1
8:9:10:7    1 1 1 1
8:9:12:13     1 1   1 1
```

ok, damit sind so oder so alle terme bedient und primimplikantentafel:

```
   1 3 7 8 9 10 11 12 13
1:8:3:11  1 1  1   1    p
8:12      1    1
8:12:9:13     1 1   1 1
9:11:8:10     1 1 1 1
8:9:10:7    1 1 1 1     p
8:9:12:13     1 1   1 1


   1 3 7 8 9 10 11 12 13
1:8:3:11  1 1  1   1    p
8:9:10:7    1 1 1 1     p
8:9:12:13     1 1   1 1
```

ok:

```
1:8:3:11  x 0 x 1
8:9:10:7  1 0 x x
8:9:12:13  1 x 0 x
```

```
 y = not x2 and not x0 or
   x3 and not x2 or
   x3 and not x1;


in galasm aus zu druecken, uebung:


device description module
device description module
device description module
device description module

GAL16V8
Module GAL16v808302026

... pins - gleich uebung

y = /x2 * /x0
  x3 * /x2
  x3 * /x1

Descriptions

oder vergessen...

ok: pins realitaet:

VCC, GND, /OE, CLK

Auf der einen Seite Input
Auf der anderen Seite Output

Im Westen: Input
Im Osten: Output

GND: Sued-West, wie immer
VCC: Pin 20: Nord-Ost

GND: Seite: Input
VCC Seite Output

CLK: ist auf Seite: Input
/OE: auf Seite Output
```

als

```
CLK X3 X2 X1 X0 NC NC NC NC GND
/OE Y NC NC NC NC NC NC NC VCC
```

so zum beispiel und ich glaube das ist richtig...

```
GAL16V8
Module GAL16v808302026

CLK X3 X2 X1 X0 NC NC NC NC GND
/OE Y NC NC NC NC NC NC NC VCC

y = /x2 * /x0
  x3 * /x2
  x3 * /x1

Description
```

kann sein # ist oder oder so:

```
GAL16V8
Module GAL16v808302026

CLK X3 X2 X1 X0 NC NC NC NC GND
/OE Y NC NC NC NC NC NC NC VCC

y = /x2 * /x0 # x3 * /x2 # x3 * /x1

Description
```

ok, wenn wir das haben, geht es weiter...

wir testen zunaechst mit eigenen mitteln, ist der ausdruck richtig, wir machen die tabelle i
auch ein timing diagramm.

n	x_3	x_2	x_1	x_0	y
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	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

so,

```
 y = not x2 and not x0 or
   x3 and not x2 or
   x3 and not x1;

testen auf not x2 and not x0...

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

ok:

not x2 and not x0
```

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


  0 0 0 0 0 0 1 1 1
  2 0 0 1 0 0 1 1 1
  8 1 0 0 0 1 1 1 1
10 1 0 1 0 1 1 1 1

x3 and not x2 or

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


  0 0 0 0 0 0  1
  1 0 0 0 1 1  1
  2 0 0 1 0 0  1
```

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

x3 and not x1:

```
  8 1 0 0 0 1 1 1
  9 1 0 0 1 1 1 1
12 1 1 0 0 1 1 1
13 1 1 0 1 1 1 1
```

9 minix

SIGHUP, SIGINT, and SIGKILL are standard POSIX signals used in Unix-like operating systems
Here is the direct breakdown of their differences, behaviors, and standard use cases:

Overview Comparison

Signal Name Signal Number Trigger Method Catchable / Ignorable? Primary Modern Purpose
SIGHUP(Hangup) 1 Closing terminal, kill -1 Yes Reload configuration files or notify terminal
SIGINT(Interrupt) 2 Pressing Ctrl+C, kill -2 Yes Interrupt foreground processes gracefully.
SIGKILL (Kill) 9 kill -9 No (Handled by OS kernel) Forcefully and instantly destroy an unres

eigenbemerkung:

1. Errors
2. Call Nr
3. Signals

Daneben gibt es

1. Errors

2. Call Nr
3. Signals
4. Interrupts (Interrupt Vektor)

typische Fehler

```
34 Fehler? NERROR 34
#define NERROR 34
#define OK 0
#define ERROR -1

// Fehler Allgemein: ERROR (-1) immer, NERROR, 34 Betriebssystem Minix
// OK: 0

/*
ENOMEM
EIO
ENOENT (Oft...)

E NO ENVIRONMENT (No Environment - keine Umgebung)
ENOMEM (Kein Speicher)
EIO (IO Fehler)
EDIV (Device Fehler)

Geraet, IO, Memory

...

SIGNALS sind klassischer weise:
SIGHUB
SIGKILL
SIGINT
SIGILL
SIG...

alle lernen

/h/signal.h
/h/callnr.h
/h/error.h

SIGHUB?? wusste ich nicht

Closing Terminal
```

Also: Konsole: Bash Schliessen
Ctrl+C (das beruehmte, womit man immer sein programm beendet ist:
SIGINT

also, das Ctrl+C, was man immer drueckt, ist SIGINT

Die Konsole selber schliessen ist SIGHUB

Dieses Ctrl+C ist -2,
und dann sagt man in Zukunft mehr (ich - sie vielleicht schon - zum lernen)
Ctrl+C sondern SIGINT

und beim Beenden der Konsole SIGHUB ...

ok: SIGKILL
ist KILL prozess beenden und zwar von aussen... also wenn ich

pkill Processname/Prog Name..

oder

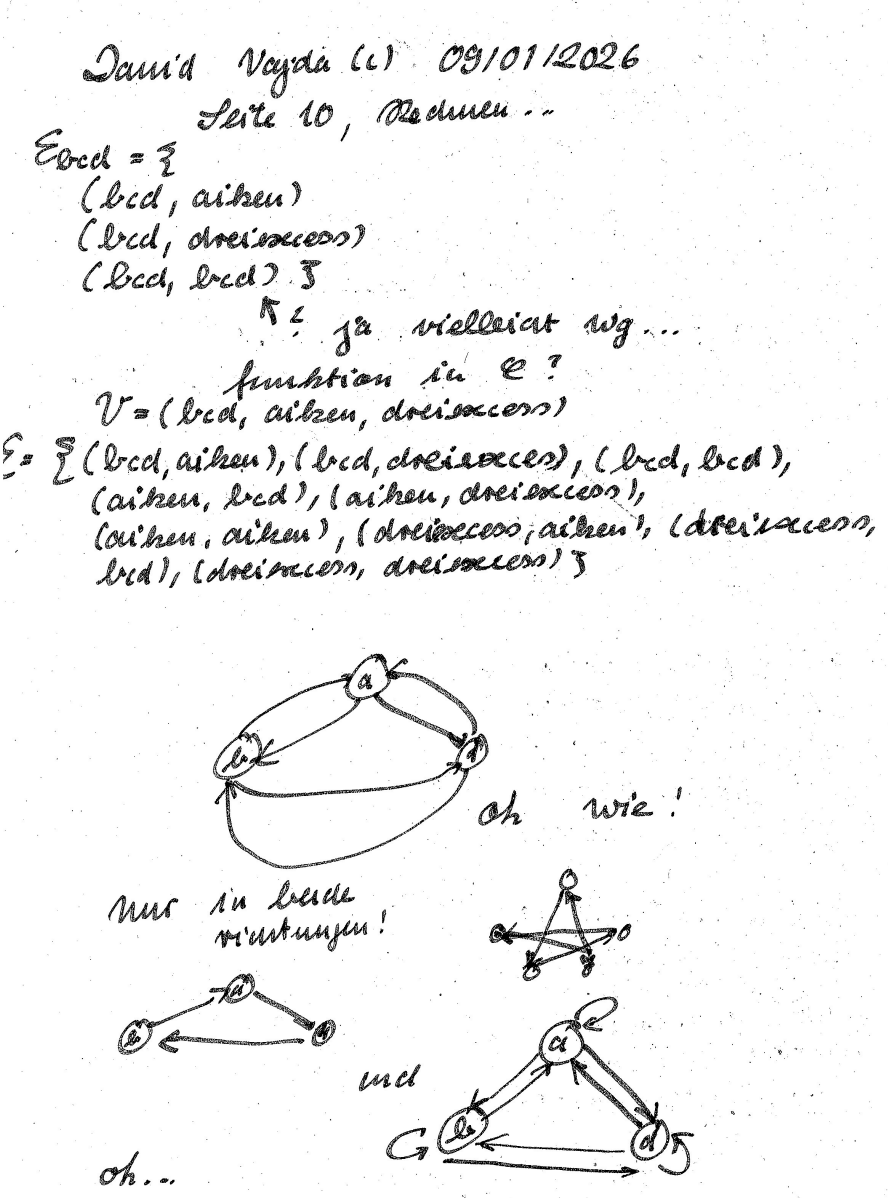
kill -KILL ProcessNr

Dann ist das

SIGKILL

Sighub - HangUp

10 hash table, codes, ...



David Vajda (c) 09/01/2026
Seite 8, Rechnen ...

1. bcd ziffer: 0b0010
2. bcd ziffer: 0b1000
3. bcd ziffer: 0b1001
4. bcd ziffer: 0b0101
5. bcd ziffer: 0b1001

oh

binär	bcd	achten_gleiaccess	
0b0000	0b0000(0)	0b0000 (0)	pt
0b0001	0b0001(1)	0b0001 (1)	pt
0b0010	0b0010(2)	0b0010 (2)	pt
0b0011	0b0011(3)	0b0011 (3)	0b0011 (0)
0b0100	0b0100(4)	0b0100 (4)	0b0100 (1)
0b0101	0b0101(5)	pt	0b0101 (2)
0b0110	0b0110(6)	pt	0b0110 (3)
0b0111	0b0111(7)	pt	0b0111 (4)
0b1000	0b1000(8)	pt	0b1000 (5)
0b1001	0b1001(9)	pt	0b1001 (6)
0b1010	pt	pt	0b1010 (7)
0b1011	pt	0b1011 (5)	0b1011 (8)
0b1100	pt	0b1100 (6)	0b1100 (9)
0b1101	pt	0b1101 (7)	pt
0b1110	pt	0b1110 (8)	pt
0b1111	pt	0b1111 (9)	pt

(c) Jamid Vojta 09/01/2026

Seite 3, Rechnen...

probe:

$bin(28959_{dec}) \overset{?}{=} 111000100011111_b$

ok von links.

... .1111b

0111.0001.0001.1111b

28959 ist < 64 ok ?

blöp: ... ? 1... ← msb

probe!

$0111^{14}.0001^{12}.0001^{10}.1111_b$

$7^{13}\ 6\ 5\ 4\ 3\ 2\ 1\ 0$

$2^{10} = 1024$

... 2^{12}

$2^{12} = 4096$

$2^{13} = 8192$

$2^{14} = 8192 \cdot 2 =$

16384

16384
+ 8192
+ 4096
+ 256
+ 16
+ 8
+ 4
+ 1 3 3 2 ← ok

28959 !

8+9 = 17 ✓ 21

9+5 = 14 22+14 = 35

1+4 = 5, 3 ok

glück!!! ok!

4+2 = 6

6+6 = 12

6+8 = 14

4+2+1 = 7 ✓

6+14 = 20

12+7 = 19+ = 39 nicht gut

ok vermutet ungerade ? ? ? aber das ist doch eine ungerade zahl in jedem falle also..

(c) David Vajda 09/01/2026

Seite 5, Rechnen...

noch mal hinschreiben...

0b-0111.0001.0001.1111/1010b = ... 010010...

(c) a b g h i j k

...111

00110
- 0000
1100(e)
- 1010
00100
- 0000(f)
1001
- 0000 (g)
10011 (h)
- 1010
1001
- 0000
10010 (i)
- 1010
1000 (j)
10001
- 1010
001111(k)
- 1010
101
- 0000
...

← 2 rückw. (Chaos! panique? panic)

← vom Eindruck zu schnell man schießt über das Ziel hinaus...

0b01... von hinten!

0b 0001.0110.1001.0111 Rest

probe! 0b0101

0b dann ist stop!

← vielst gut aus 101 ist 5!

(c) Jamid Vayda
(08) 09/01/2026
Seite 2, Rechnen ...

28959
ich habe eine Lösung für das Problem gefunden:

echo $((RANDOM % (8192 - 4)))
↑ grob

28959 zunächst in eine Binärzahl umrechnen: 1110001000111111

28959/2 = 14479 Rest 1
14479/2 = 07239 Rest 1
7239/2 = 3619 Rest 1
3619/2 = 1809 Rest 1
1809/2 = 904 Rest 1
904/2 = 452 Rest 0
452/2 = 226 Rest 0
226/2 = 113 Rest 0
113/2 = 56 Rest 1
56/2 = 28 Rest 0
28/2 = 14 Rest 0
14/2 = 7 Rest 0
7/2 = 3 Rest 1
3/2 = 1 Rest 1
1/2 = 0 Rest 1

(c) David Vajda
(08/) 09/01/2026
Seite 1 Rechenübungen
+ Codes...

Was machen wir heute?
folgendes:

1.) aiken
2.) dreiexcess
3.) gray o. hamming?

... oh: wie auch immer...

Dazu kommt:

1.) Zufallszahlen
2.) fletchers Checksum
(wie adler-32)
3.) hash table...
4.) vor allem: crc...

oh: fangen wir an:

* Dazu matrizenmultiplikation indem wir das senkrecht y Wort mit der $x_3\ x_2\ x_1\ x_0$ Matrix der input variablen multiplizieren...

oh:

wir fangen an: Eine Zufallszahl im pc:
echo $((RANDOM % MAX_INT))

(c) Daniel Vajda 09/01/2026

Seite 6, Rechnen...

0b 0001. 0110. 1001. 0111 Rest 0b0101
12 11 9 7 6 5 4 3 2 1 0

probe:

4096
+ 1024
+ 512
+ 128
+ 16
+ 4
+ 2
+ 1 3 1
5783

6+4 = 10
2+8 = 10
6+4 = 10
2+1 = 3
33

9+1 = 10
2+2+1 = 5
...+3 = 18

5783 · 10 + 5

↳ Da existiert fehler!

weil:

28959 ich dachte 5 am Ende...

ok: projekt ablreden hier an der Stelle: nächstes ...

pause!

(c) Jamil Vayda 09/01/2026
Seite 4, Rechnen...

im nächsten Schritt teilen wir binär durch 10_{dec} (dec)? bin(10_{dec}) = 1010b

ohne ins - zu fallen * Ob 1010

Ob 0111.0001.0001.1111 / 1010b = 01011

- 1010 if... siehe * ← ausdruck?

(0000) falsch

0111

0111.0

- 1010

01000 (a)

- 0000 ← angst fehler hat sich eingeschlichen (klaube nicht) wird sich zeigen!

10000 (b)

- 1010

001101 (c)

- 1010

0011 ... ok nächste Seite ↙ Zeilen vorschub

* soll bleiben <cr><lf>?

was heißt Seiten vorschub?

if ((0111 - 1010) ≥ 0) in Latex:

then vfill

... pagebreak

else

return 0111 pagebreak ↙ = parse return u. carriage return

?

hinweisgebend!

(c) Jamal Vajda 09/01/2026

Seite 7, Rechnen...

als nächstes rechnen wir die Zahl trotzdem als Dezimal Zahl einfach in bcd...

und in die anderen Codes...

$bcd(28959_{dec})$ = 0b0010.1000.1001.0101.1001

↑ fehlt!

0b0000.0000.0000.0010.1000.1001.0101.1001

gepackt: 5 bcd ziffern in einem:

32 bit doppelwort!

ungepackt wäre es:

3 x 0b0000 bei 5 heißt 2 byte extra notwendig

0b0000.0010
0b0000.1000
0b0000.1001
0b0000.0101
0b0000.1001

Das sind 5 byte davor 4 byte!

oh: noch:

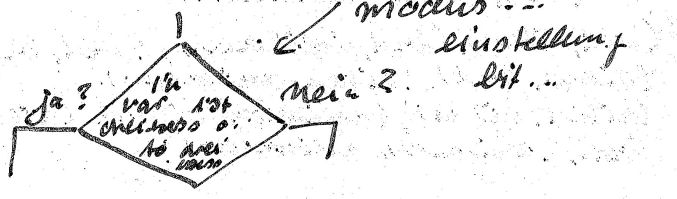
zehen + dreizehen!

Jamid Vajda (c) 09/01/2026
Seite 9, Rechnen...

ok:

1.) 0b0010 ist ein aiken/
dreiexcess... code

2.) 0b 0010 in bcd in aiken/
dreiexcess ... code ausdrücken



aiken(0bcd 0010) =
aiken(0bcd 1000) =
aiken(0bcd 1001) =
aiken(0bcd 0101) =
aiken(0bcd 1001) =

dreiexcess(0aik 0010) =
dreiex(0aik 1000) =
dreiex(1001 aik) =
dreiex(0101 aik) =
dreiex(0b 1001 aik) =

dreiexcess(0bcd 0010) =

möglichkeiten
paare im graph ?
?! ? ok

Damir Vajda (c)
09/01/2026, Seite 11, Rechnen -

dann am besten 3 tabellen!

bcd Ausgangsmenge -
← hat als int zu interpretieren zu werden..

n	bcd	aiken	dreiexcess
1	0bcd 0010 (2)	$0b\,0010_{aiken}$ (2)	pt
2	0bcd 1000 (8)	pt	$0b\,1000_{drei}$ (5)
3	0bcd 1001 (9)	pt	$0b\,1001_{drei}$ (6)
4	0bcd 0101 (5)	pt	$0b\,0101_{drei}$ (2)
5	0bcd 1001 (9)	pt	$0b\,1001_{drei}$ (6)

↑ wenn mehr fehlen..

n	aiken	bcd	dreiexcess
1	$0b\,0010_{aiken}$: pt	x	a
2	$0b\,1000_{aik}$: 5	0bcd 0101	$0b\,1011_{drei}$
3	$0b\,1001_{aik}$: 6	0bcd 0110	0b 10
4	↖ kaputt!	falsch!!:	
5			

ah abbrechen: beim nächsten mal wieder?
nein: aber ect fletchers checksum!
knacke und what we aber dec + bcd

David Vajda (c) 09/01/2026
Seite 12, Rechnen--

Später:

- Matrizenmul!
- ähnen, streichen, bed führt zu verschiedenen matrizen + ausserhalb innerhalb?
- crc! nicht vergessen
- Integral
- Logarithmus
  und!

was mir fehlt ist:

fft: fast fourier transformation?
vielleicht!

lieber: polynom multiplikation!
gleich!

Daniel Vajda (c) 09/01/2026

Seite 13, Rechnen...

1.) fletcher's checksum auf:

$\mathcal{X} = (2, 8, 9, 5, 9)$

```
sum1 := 0
sum2 := 0
for i := 0 to length(X) do
        sum1 := (sum1 + X[i]) modulo 7
        sum2 := (sum2 + sum1) modulo 7
endfor
checksum := (sum1, sum2)
```

ergänzen mit Flecken!

als Folge ausdrücken?

1 Vorschrift

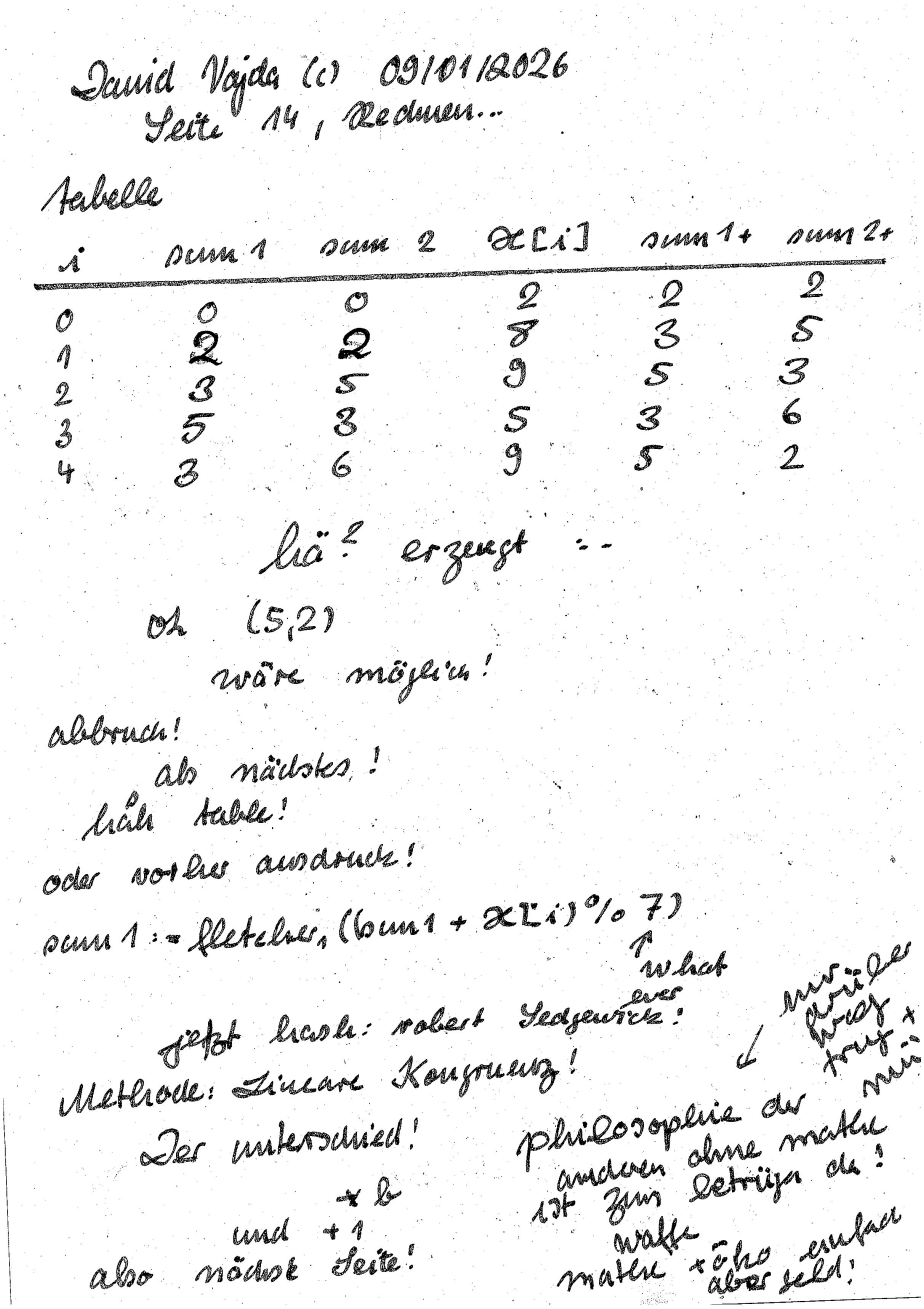
-später...



Tabelle

i	sum1	sum2	sum1+	sum2+	X[i]
0	0	0	2	2	2
1	2	2	(8+2) mit 7 = 3	5	8
2	3	5	0	5	9
3	0	5	5	3	5
4	5	3	0		

Verwendung Spalten oder Zeilen organisiert!



im nächsten schritt folgt die hash-table, die zufallszahlen, die fletchers checksum mit latex

```
X = (2,8,9,5,9)
```

```
fletcher's checksum
i sum1 sum2 X [i] sum1+ sum2+
0 0  0  2 2  2
1 2  2  8 3  5
2 3  5  9 5  3
3 5  3  5 3  6
4 3  6  9 5  4

sum1_(i+1) := fl1 (sum1_i + X [i]) % 7
sum2_(i+1) := fl2 (sum1_i + sum2) % 7

eigentlich eher:  sum1_(i) := fl1 (sum1_(i-1) + X [i]) % 7
ja!!! nein!, weil negatives i-1, fuer i=0!
oder mathematik folge faengt mit 1 an zu zaehlen, waehrend
sonst 0. ???
halt wiederlegt. y_(i+1) ist fuer i = 0, 1. damit stimmt die sache!

ernsthaft ist die aussage:
sum1_(i+1) := fl1 (sum1_i + X [i]) % 7
sum2_(i+1) := fl2 (sum1_i + sum2) % 7
y_(i+1) = hash (12*y_(i)+X[i]) % 7
weil: i = 0 ist i+1 = 1 und damit fangen in mathe folgen an, es gibt keinen
negativen index

(5,4)

moeglich:


 i X [i] sum1 sum2 sum1+ sum2+
0 2 0  0  2  2
1 8 2  2  3  5
2 9 3  5  5  3
3 5 5  3  3  6
4 9 3  6  5  4

hash-table

y_i = hash (12*y_(i-1)+X[i]) % 7
y_0 = 0

oder?? auch zu oben!!!

y_(i+1) = hash (12*y_(i)+X[i]) % 7
y = 0
```

```
i X [i] y_(i) y_(i+1)
0 2 0 2
1 8 2 (12*2 + 8) % 7 = (24+8) % 7= 32 % 7 = 4
2 9 4 (12*4 + 9) % 7 = (48+9) % 7 = 1
3 5 1 (12*1 + 5) % 7 = 17%7 = 3
4 9 2 (12*3 + 9) % 7 = (36+9) % 7 = (45 % 7) = 3

 ok:

 i X [i] y_(i) y_(i+1)
0 2 0 2
1 8 2 4
2 9 4 1
3 5 1 3
4 9 2 3

 random (lineare Kongruenz)

 a [i] = (a [i-1] * b + 1) % m

eine idee, die ich jetzt haette, waere b entsprechend der folge
X = (2,8,9,5,9) zu waehlen.
einfach als beispiel noch.
b = 12
m = 7
seed = a [0] = 2

i a [i]
0 2
1 (12*2+1) % 7 = 25%7 = (21+4)%7 = 4
2 (12*4+1) % 7 = 49%7 = 0
3 (12*0+1) % 7 = 1
4 (12*1+1) % 7 = (13 % 7) = 5
5 (12*5+1) % 7 = (60+1) % 7 = (56 + 5) % 7 = 5

 zwei forderungen:
 1.) nicht zwei zahlen duerfen zur selben zufallszahl fuehren
 2.) m und b muessen so gewaehlt sein, dass sie bei der zufallzsahl nicht auf dieselbe abbil

 kann man die gleichung loesen, ist die frage:

 y = (12*y+1) % 7

 ok, aber das spaeter.

 nun, erst in LaTeX Tabellen und dann: CRC Zyklische Redundanzpruefung.
```

i	X_i	$sum1$	$sum2$	$sum1+$	$sum2+$
0	2	0	0	2	2
1	8	2	2	3	5
2	9	3	5	5	3
3	5	5	3	3	6
4	9	3	6	5	4

ok: CRC-Check

```
0b0111.0001.0001.1111b

CRC mit 0b1010 und vorher mit... 11001

Rahmen  0b0111.0001.0001.1111
Pruefwert  0b1010
   ---------
      1101
      11010001

was habe ich vergessen?
1.) es wird ein pruefwert bestimmt
2.) die laenge - oft wohl 5 bit (also n bit)
3.) der Rahmen (nicht wie beim Paging) wird um n-1 0en bit geclearte bit erweitert.

Pruefwert: 1101b
Rahmen: 0b0111.0001.0001.1111.000

1101b

polynomdivision: ok
polynommultiplikation wollten wir machen, mit...
dann kommt langrage ... das brauchen wir ja bei FFT...


Rahmen:    0b011.1000.1000.1111.1000
Pruefwert:   0b101.1
Division:     / (bzw. XOR)
Zwischenergebnis: 0b110.0 ...
die frage ist, soll ich das hintere erhalten...

jetzt geht es tatsaechlich um: relevante stellen!

das wort wurde bisher nicht geklaert!
relevante stellen, naemlich die 1 im msb - most significant bit.

ok:
```

```
1.) frage: zwischenergebnis
2.) frage: von anfang relevante stellen

a) theoretisch: entspricht das der division die ich bisher binaer ausgefuehrt habe
b) theoretisch: ist die subtraktion dabei immer ein XOR
c) theoretisch: die relevanz von stellen.


Rahmen:    0b011.1000.1000.1111.1000
Pruefwert:   0b101.1
Division:     / (bzw. XOR)
Zwischenergebnis: 0b110.0 ...
Zwischenergebnis: 0b110.0000.1000.1111.1000

ok: wir brechen das ab und nehmen gleich von anfang an relevante stellen:

Rahmen:   0b11.1000.1000.1111.1000
Pruefwert:  0b1011

ok. wir verschieben den punkt

Rahmen:    0b1110.0010.0011.1110.00
Pruefwert:   0b1011
Division:   / (XOR)
Zwischenergebnis: 0b0110.0010.0011.1110.00

relevante stellen:

Rahmen:   0b1100.0100.0111.1100.0
Pruefwert:   0b1011
Divsion:   / (XOR)
Zwischenergebnis: 0b0111.0100.0111.1100.0

relevante stellen:   0b1110.1000.1111.1000

Rahmen:   0b1110.1000.1111.1000
Pruefwert:   0b1011
Division:   / (XOR)
Zwischenergebnis: 0b0101.1000.1111.1000

relevante stellen: 0b101.1000.1111.1000
Rahmen:   0b1011.0001.1111.000
Pruefwert:   0b1011
Division   / (XOR)
Zwischenergebnis: 0b0000.0001.1111.000
```

```
relevante stellen: 0b1111.1000
Rahmen:   0b1111.1000
Pruefwert:   0b1011
Divsion:   / (XOR)
Zwischenergebnis: 0b0100.1000

Relevante stellen:
Rahmen:   0b1001.000
Pruefwert:   0b1011
Division:   / (XOR)
Zwischenergebnis: 0b0010.000

ok, ich muss jetzt abbrechen, es waere moeglich da sind fehler drin
```

jetzt machen wir langrage!

also, wir haben die punkte: y-Werte.

$$Y = (2, 8, 9, 5, 9)$$

und das auf den x-Stellen, Abszissen-Werten.

$$X = (0, 1, 2, 3, 4)$$

und nun kommt langrage.
also:

1. ordinatenwerte: $Y = (2, 8, 9, 5, 9)$
2. abszissenwerte: $X = (0, 1, 2, 3, 4)$

langrage:

$$l_i(x) = \prod_{\substack{i=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}$$

ich stelle wie ich das gesetzt habe, in LaTeXvor

```
$$l_i(x) = \displaystyle\prod_{\stackrel{i=0}{j\neq i}}^n\dfrac{x-x_j}{x_i-x_j}$$
```

$$l_i(x) = \prod_{\substack{i=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j} = \frac{x - x_0}{x_i - x_0} \cdots \frac{x - x_{i-1}}{x_i - x_{i-1}} \cdot \frac{x - x_{i+1}}{x_i - x_{i+1}} \cdots \frac{x - x_n}{x_i - x_n}$$

so, die formel die Formel in LaTeX

```
$$l_i(x) = \displaystyle\prod_{\stackrel{i=0}{j\neq i}}^n\dfrac{x-x_j}{x_i-x_j} = \dfrac{x-x
```

nun die formel, fuer $l_i(x)$ jeweils.
damit sind die absizssenwerte nicht optimal, aber so:
$X = (1, 2, 3, 4, 5)$

```
i x[i] x[j=0] x[j=1] x[j=2] x[j=3] x[j=4] (x-x[j=0])/(x[0]-x[j=0])  (x-x[j=1])/(x[0]-x[j=1])
0 2 x  8  9  5  9  x     (x-8)/(2-8)
1 8 2  x  9  5  9
2 9 2  8  x  5  9
3 5 2  8  9  x  9
4 9 2  8  9  5  x
```

ich mache pause

ok, ich probiere jetzt atmega8 programme inzwischen zu schreiben

```
;; (c) david vajda
;; 09/01/2026
;; m8/led count/exersize

.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
l1:
out PORTB, r16
rcall r1
dec r16
rjmp l1

r1:
push r16
push r17
push r18
ldi r16, 0xff
r1l1:
ldi r17, 0xff
r1l2:
```

```
ldi r18, 0x04
r1l3:
dec r18
brne r1l3
dec r17
brne r1l2
dec r16
brne rl1
pop r18
pop r17
pop r16
ret
```

ok, das dürfte funktionieren

```
;; (c) david vajda
;; 09/01/2026
;; m8/INT0/INT1

.include "m8def.inc"

ldi r16, HIGH (RAMEND)
out SPH, r16
ldi r16, LOW (RAMEND)
out SPL, r16

ldi r16, (1 << INT0) | (1 << INT1)
out MCUCR, r16
ldi r16,
out GICR, r16
```

so, ich fasse die Beitrge zusammen, ich mache noch langrage vielleicht interpolation von Polynomen???

ich brauche dafuer eine richtige tabelle, vielleicht, fuer die einzelne, wie muss die aussehen auf papier???

spaeter ein wenig pause, trotzdem gut!

```
i x[i] x[j=0] x[j=1] x[j=2] x[j=3] x[j=4] (x-x[j=0])/(x[0]-x[j=0])  (x-x[j=1])/(x[0]-x[j=1])
0 2 x  8  9  5  9  x     (x-8)/(2-8)
1 8 2  x  9  5  9
2 9 2  8  x  5  9
3 5 2  8  9  x  9
4 9 2  8  9  5  x
```

dass diese tabelle drauf passt? nur wie?

etwa eine 3-D so zu sagen in zwei D. ok, machen wir mal, ist ja nicht ungewoehnlich

```
mit Robert Sedgewick oder Wikipedia.

vielleicht diese version ohne vollstaendiges ausfuellen.
i x[i] j x[j] (x-x[j])/(x[i]-x[j])
0 2 0 2 ---
  1 8 (x-8)/(2-8)
  2 9 (x-9)/(2-9)
  3 5 (x-5)/(2-5)
  4 9 (x-9)/(2-9)
1 8 0 2 (x-2)/(8-2)
  1 8 ---
  2 9 (x-9)/(8-9)
  3 5 (x-5)/(8-5)
  4 9 (x-9)/(8-9)
2 9 0 2 (x-2)/(9)
  1 8 (x-8)/()
  2 9 ---
  3 5 (x-5)/
  4 9 (x-9)/
3 5 0 2
  1 8
  2 9
  3 5 x
  4 9
4 9 0 2
  1 8
  2 9
  3 5
  4 9 x

ok: hier stellt sich was interessantes heraus:

wenn: bijektivitaet: surjektivitaet, injektivitaet...
also:

achso!

ich weiss, was der fehler war! ich habe die falschen werte genommen
ich habe y genommen nicht x. das ist das problem.
y und nicht x!


i x[i] j x[j] (x-x[j])/(x[i]-x[j])
0 0 0 0 ---
  1 1 (x-1)/(0-1)
  2 2 (x-2)/(0-2)
```

```
  3 3 (x-3)/(0-3)
  4 4 (x-4)/(0-4)
1 1 0 0 (x-0)/(1-0)
  1 1 ---
  2 2 (x-2)/(1-2)
  3 3 (x-3)/(1-3)
  4 4 (x-4)/(1-4)
2 2 0 0 (x-0)/(2-0)
  1 1 (x-1)/(2-1)
  2 2 ---
  3 3 (x-3)/(2-3)
  4 4 (x-4)/(2-4)
3 3 0 0 (x-0)/(3-0)
  1 1 (x-1)/(3-1)
  2 2 (x-2)/(3-2)
  3 3 ---
  4 4 (x-4)/(3-4)
4 4 0 0 (x-0)/(4-0)
  1 1 (x-1)/(4-1)
  2 2 (x-2)/(4-2)
  3 3 (x-3)/(4-3)
  4 4 ---
```

ok. das wäre besser, als schöne tabelle:

i	x_i	j	x_j	$\frac{x - x_j}{x_i - x_j}$
0	0	0	0	EDIVBYZERO
		1	1	$\frac{x-1}{0-1}$
		2	2	$\frac{x-2}{0-2}$
		3	3	$\frac{x-3}{0-3}$
		4	4	$\frac{x-4}{0-4}$
1	1	0	0	$\frac{x-0}{1-0}$
		1	1	EDIVBYZERO
		2	2	$\frac{x-2}{1-2}$
		3	3	$\frac{x-3}{1-3}$
		4	4	$\frac{x-4}{1-4}$
2	2	0	0	$\frac{x-0}{2-0}$
		1	1	$\frac{x-1}{2-1}$
		2	2	EDIVBYZERO
		3	3	$\frac{x-3}{2-3}$
		4	4	$\frac{x-4}{2-4}$
3	3	0	0	$\frac{x-0}{3-0}$
		1	1	$\frac{x-1}{3-1}$
		2	2	$\frac{x-2}{3-2}$
		3	3	EDIVBYZERO
		4	4	$\frac{x-4}{3-4}$

4	4	0	0	$\frac{x-0}{4-0}$	
		1	1	$\frac{x-1}{4-1}$	
		2	2	$\frac{x-2}{4-2}$	im nächsten schritt rechnet man die aus,
		3	3	$\frac{x-3}{4-3}$	
		4	4	EDIVBYZERO	

und dann addiert man sie. einzeln. und dann kommt das einsetzen, danach das integral.

```
(x-1)/(-1)*(x-2)/(-2)*(x-3)/(-3)*(x-4)/(-4) = ((x-1)*(x-2)*(x-3)*(x-4))/((-1)*(-2)*(-3)*(-4)
((x-1)*(x-2)*(x-3)*(x-4))/((-1)*(-2)*(-3)*(-4)) = [(x-1)*(x-2)]*[(x-3)*(x-4)]/(24) =
([x^2-2*x-x+2]*[x^2-3*x-4*x+12])/24 =
(x^2-3*x+2)*(x^2-7*x+12)/24 =

ok, was wir jetzt brauchen ist polynommultiplikation.
wir haben polynomdivision und die multiplikation

ein ansatz waere - also, natuerlich ist es an und fuer sich einfach, aber wir brauchen ein
verfahren.

und zwar koennten wir eine tabelle nehmen.
wir nehmen eine tabelle und nehmen bei beiden die potenzen, die koeffizienten...

eine quadratische? probieren wir es aus:
  b_2 b_1 b_0
a_2
a_1
a_0

oder wir druecken es aus, als:
  a_{22} a_{21} a_{20}
a_{12}
a_(11}
a_{10}

dabei entspricht das den potenzen
(x^2-3*x+2)*(x^2-7*x+12)

  (2,1)  (1,-3) (0,2)
(2,1)
```

```
(1,-7)
(0

ok, noch mal:

  a_{22} a_{21} a_{20}
a_{12}
a_(11}
a_{10}

vielleicht als paare, weil: wir muessen ja auch den exponenten multiplizieren

dabei entspricht das den potenzen
(x^2-3*x+2)*(x^2-7*x+12)

  (2,1)  (1,-3) (0,2)
(2,1)  (4,1)  (3,-3) (2,1)
(1,-7) (3,-7) (2,21) (1,-7)
(0,12) (2,12) (1,-36) (0,24)

ok und jetzt alle paare addieren, die gleiches (a,...) oder (a_1,...) haben

  (2,1)  (1,-3) (0,2)
(2,1)  (4,1)  (3,-3) (2,1)
(1,-7) (3,-7) (2,21) (1,-7)
(0,12) (2,12) (1,-36) (0,24)

(4,1)
(3,-3), (3,-7)
(2,1), (2,21,) (2,12)
(1,-7), (1,-36)
(0,24)

x^4 -10 * x^3 + 34 *x^2 + -43*x+24

waere moeglich, womoeglich fehler, also noch mal mathematisch.
```

i	x_i	j	x_j	$\frac{x - x_j}{x_i - x_j}$
0	0	0	0	EDIVBYZERO
		1	1	$\frac{x-1}{0-1}$
		2	2	$\frac{x-2}{0-2}$
		3	3	$\frac{x-3}{0-3}$
		4	4	$\frac{x-4}{0-4}$
1	1	0	0	$\frac{x-0}{1-0}$
		1	1	EDIVBYZERO
		2	2	$\frac{x-2}{1-2}$
		3	3	$\frac{x-3}{1-3}$
		4	4	$\frac{x-4}{1-4}$
2	2	0	0	$\frac{x-0}{2-0}$
		1	1	$\frac{x-1}{2-1}$
		2	2	EDIVBYZERO
		3	3	$\frac{x-3}{2-3}$
		4	4	$\frac{x-4}{2-4}$
3	3	0	0	$\frac{x-0}{3-0}$
		1	1	$\frac{x-1}{3-1}$
		2	2	$\frac{x-2}{3-2}$
		3	3	EDIVBYZERO
		4	4	$\frac{x-4}{3-4}$

4	4	0	0	$\frac{x-0}{4-0}$
		1	1	$\frac{x-1}{4-1}$
		2	2	$\frac{x-2}{4-2}$
		3	3	$\frac{x-3}{4-3}$
		4	4	EDIVBYZERO

	(2,1)	(1,-3)	(0,2)
(2,1)	(4,1)	(3,-3)	(2,1)
(1,-7)	(3,-7)	(2,21)	(1,-7)
(0,12)	(2,12)	(1,-36)	(0,24)

$$f(x) = x^4 - 10 \cdot x^3 + 34 \cdot x^2 + -43 \cdot x + 24$$

$(x-1)/(-1)\cdot(x-2)/(-2)\cdot(x-3)/(-3)\cdot(x-4)/(-4) = ((x-1)\cdot(x-2)\cdot(x-3)\cdot(x-4))/((-1)\cdot(-2)\cdot(-3)\cdot(-4)) = ((x$

Jetzt ist Pause ich mach nachher einen Beitrag aus der Sache. Ich mach jetzt einen Beitrag aus der Sache. Ich räum das Zimmer auf und was mache ich noch? Ich mach aus der Sache einen Beitrag das kommt aber auch so auf die Homepage und ich sag was ich dazu denke ich geb die Prognosen und so weiter alles ab alles was dazugehört

So, damit müßten wir auskommen, ein Register müßte wiederum als buffer dienen, dazu daß am Ausgang Info erhalten bleibt... RAM rom Alu Divisionsbefehl inklusive... in Hardware algo...

Also, warten Sie, ich hab noch einen Fehler erkannt. Die Eingänge von den Datenspeichern müßen reguliert werden. Ebenso also, wir haben beim Datenspeicher.

Einen Daten, Ausgang und einen Daten Alleingang, ein Steuerwort Chips, Select, und so weiter Write und Output enable und die Adreßen, die ein einziges Eingang sind

Trotzdem haben wir bisher sozusagen die ALU mit ihrem Ausgang alleine sozusagen mit den Datenspeichern benutzt aber die sozusagen Adreß die Adreßeingänge haben keine Verbindung zu einem sozusagen Programm. Speicher was fest untergebracht ist und gleichzeitig zu einem Steuer Wort

Die Sache funktioniert trotzdem und Sie müßen keine falsche Vorstellung haben. Ich mein das ernst weil, was sie jetzt haben, sind sozusagen vier feste Speicher.

Und es macht aber keinen Unterschied, die müßten sie parallel quasi programmieren. Trotzdem ist die Sache unsinnig, sich da Gedanken zu machen. Sie können sogar Paging unterbringen und eine memory Management Unit ohne Probleme. Sie können damit alles unterbringen aus folgendem Grund.

Das, was wir hier haben, ist eine Microarchitektur eine Microarchitektur. Da müßen Sie aufpaßen da ist einbezogen. Sozusagen die Programmierung der eeprom mit oder ohne daß ihren Speicher integriert ist das kann man nachher immer noch mischen, also das erste

Erstens das, was wir jetzt haben, ist eine Mikroarchitektur und die Mikroarchitektur enthält das Mikrogramm. Es ist letztenendes vollkommen egal wie die Sache funktioniert von der Sache her. Wir müßen die Sache nur möglichst einfach schalten.

Wir nehmen die Arbeitßpeicher bringen Sie in einen Speicherbaustein einen festen mit den Adreßen Daten Ausgang. Diese gehen in die ALO und einer der festen Speicher ist gleichzeitig sozusagen mit dem Steuer Wort der ALU verbunden.

Wenn es da rausgeht, geht es allerdings nicht in den Datenspeicher, sondern es geht doch von mir aus, direkt in den Datenspeicher. Aber wie gesagt, wir können Eingänge miteinander verbinden. Es geht wieder in die festen Speicher.

Wer Kritik an der Sache übt ihr hier sind Leute die laßen sich leicht durcheinanderbringen. Die Leute, die das nicht tun gut und der Rest soll bitte den Mund halten und sich mit seinem eigenen Thema beschäftigen oder lieber nicht weil die meisten Themen auf dieser Welt sind Müll und soll nicht über andere Themen regieren. So der Witz bei der Sache ist nämlich, daß wenn wir jetzt die Adreßen haben dann müßen wir sozusagen diese festen Speicher ebenso mit der Sache verbringen, das ist das sind parallel programmierbare und zwar nicht einzeln, sondern das ist ein einziger dekodieren und der Ausgang. Hier kommt unser eigentlicher Arbeitßpeicher hin später in einem großen Modell, womit wir sehr schnell Sachen lösen. Wir können auch nicht behaupten, daß Intel das an-

ders gemacht hat, intern trotzt Pipeline.

Auf diese Art und Weise gelingt es uns zum Beispiel sehr viele Dinge auf dem Prozeßor unterzubringen und Speicherplatz zu sparen. Die Steuer Voort in den Sachen, in denen in den festen Speichern ist sozusagen Teil der Micro Architektur Teil des Prozeßor. Das ist das eine. Es ist aber unsere Aufgabe mit der Software der Hardware Algorithmus ist auch eine Software herauszufinden, wo die Grenzen der Sache sind und die Sache trotzdem zu programmieren. Wenn diese Mikroarchitektur fertig ist, kann sie von außen immer noch mit einfachen befehlen bearbeitet werden. Und es ist Aufgabe der Software sozusagen die Grenzen innerhalb dieser Sache einzuhalten und damit alle Probleme zu lösen da sind sie aber danke schön jetzt muß ich was zu eßen kaufen. Danke

Ich hab ne geile Idee. Zum Glück hab ich das Ding gestern aufgebaut.

Manchmal lohnt es, sich, die Finger schmutzig zu machen. Ich hätte mir auch auf dem Papier schmutzig machen können.

Hab ich mehr oder weniger auch weil ich wenig gemacht hab. Ich hätte mir aber

Naja, denken wir über den Prozeß nach, mache ich ein Papier draus, was man tun muß, um sich auf dem die Finger schmutzig zu machen, ich hätte eine Buchführung. Heute würde man sagen Datenbank führen müßen müßen über die Dinger die ich als Bausteine bei mir habe

Trotzdem ist sie Situation unrealistischer Schreiben für den rein wohl gar nichts. Ist ja alle Infos die wir haben.

Gut, also folgendes trotzdem: also mir ist aufgefallen, die 74HC175

Sind erst mal C, d.h. CMOS Das läßt sie eigentlich nicht mischen mit dem LS. Was TTL

Ist

Es kommt noch schlimmer die Dinger haben kein allgemeines Chip enable Write enable aber ein CLK doof trotzdem, den Prozeßor arbeitet mit einem Takt der Computer

Und wenn der Takt da ist, kann ich zum Beispiel über einen UND gleichzeitig die Eingabe übernehmen und gleichzeitig den Takt machen, dann muß ich das sozusagen ver mischen. Trotzdem ist es eigentlich keine Lösung. Weil über das den Takt wird. Eigentlich der Prozeßor arbeitet ja mit dem Takt und es ist keine gute Lösung.

Ich hab aber ne viel beßere Lösung. Mir ist noch was aufgefallen. Es gibt noch weitere Nachteile

Zwei Register sind sehr wenig

Wir hatten das Problem, der Computer hat keinen Arbeitßpeicher

Ein Prozeßor ohne Arbeitßpeicher ist eigentlich kein Computer, maximal Mikro Programm steuerwerk

Wir haben das Problem der Wort Breite vier bitte sehr wenig. Vor allen Dingen ist es deswegen sehr wenig.

Damit laßen sich keine großen Adreßen adreßieren

Wir haben keinen Stapel

Der Stapel braucht die Operationen Push und Pop Ohne diese Operationen können wir keine Sprungbefehle in einen Unterprogramm Aufruf machen.

Das klingt sehr kompliziert und nach viel Hardware, aber das Problem mit der Hardware ist in dieser Hinsicht nicht vorhanden. Aus einem einzigen Grund.

Die Hardware liegt in diesem Fall in dem eeprom/prom wie auch immer das der Speicher enthält, den Programmspeicher bei Harvard sind Daten atmega8 gehört dazu und der Code getrennt

So, der andere prom enthält allerdings die Adreße ist sozusagen unsere Eingangs vektor X in das Schaltnetz mit den sozusagen Eingangs variablen X1 und X0 und so weiter

Während also die Adreße wären, sozusagen des Daten Wort am Ausgang des Steuer Wort ist, also quasi folgendes, sozusagen unser Steuer Bus und so weiter jedenfalls

Wenn ich das habe, ist mir aufgefallen ich habe noch RAM Bausteine

Gucken Sie mal, ich hab den Registers Satz

Und es ist jetzt ein Beispiel dafür, daß er eigentlich der gesamte Hardware Aufwand eigentlich in dem prom sitzt d.h. man könnte jetzt Angst haben so natürlicherweise, daß ich bei diesen Sachen zum Beispiel dem Stapel sehr viel an der Sache herum schalten muß. Das muß ich allerdings überhaupt nicht lustigerweise bis hin zum Stapelzeiger. Ich brauche kein getrenntes Register mehr oder weniger. Also ich muß nicht jetzt noch da kommen und eine sozusagen irgendein Register neben ein 74 LS 175 das brauche ich alles nicht.

Sondern ich bringe das alles in diesem Speicher unter. Also verschiedene feste Speicher davon sind die einen der Programm Speicher aber auch der Dekodierer

Das an der Sache ist, daß ich sehr gute RAM habe

Kann die vielen ALU parallel schalten?

D.h. wenn ich eine acht Bit ALU haben möchte d.h. nicht daß ich nicht 16 Bit adreßieren kann zum Beispiel der atmega8 Kann das, ich glaube der 8085 auch

Es gibt jetzt verschiedene Optionen. Entweder ich nehme einen dieser ram. Das wird dann schwierig, weil

Ich meine, ich habe darüber ja öfter nachgedacht das ist der berühmte Registers Satz selber ist auch quasi ein ram

Weil die Register sind intern Adreßen, also die Register und ihre Nummerierung sind intern Adreßen

Gibt allerdings Speicher die haben gleichzeitig zwei Output und ein Input

Das könnte man sogar wiederum formulieren, als wir nehmen einfach zwei Speicherbausteine

Es ist durchaus üblich, daß wir einen oberen Teil an Registern haben und einen unteren Teil

R0-15 und r16 bis r31

Es ist nicht unbedingt üblich, daß man zum Beispiel R Null nicht mit R1 zusammenbringen kann und trotzdem ist es auch in diesem Fall nicht so bedeutend. Dafür haben wir sehr viel Register. Dafür können wir von verschiedenen Annahmen Gebrauch machen

Sache wird jetzt sehr, sehr praktisch. Wir sparen sehr viel Bauteile.

Wir können zum Beispiel zwei dieser Speicher nehmen und das ist ganz einfach

Wir haben jetzt sogar noch die Vorzüge, daß wir alle drei EnNable haben, die wir typischerweise haben

Output, Write und Chip Cowboy

Ohne das Boy und damit haben wir unseren Speicher vielleicht auch doppelt. Wir können problemlos einen Teil davon als Stapelzeiger gebrauchen. Wir können damit viele Probleme auf einmal erledigen und wir haben jetzt einen sehr guten Speicher. Daran müßen wir denken, damit wird die Sache problemlos funktionieren. Ich zeige ein Foto von meinen Speichern und fange dann nachher an. Das Ding. Wird noch heute programmiert werden muß allerdings das

eeprom

Ich mache jetzt nicht weiter, weil ich auch erst mal morgens werden muß und die Leute im Haus erwachen also folgendes. Ich muß die Aufgabe so machen, daß sie ohne tristate funktioniert. Und ohne multiplexer

Jetzt haben Speicherbausteine einen Vorteil. Laut dem gelben Buch heißt es bei mir sie haben den Zustand als Gesamtspeicher über den Modus stand-by damit müßten Sie ein treestate integriert haben. Das Problem ist.

Muß jetzt erst mal die Leute in Ruhe laßen, weil ich diktierdas. Ich sag das nur noch kurz

Idee ist auch langfristig nicht dumm. Wenn ich dieses Ding so baue, dann hab ich mir auf jeden Fall Bauteile erspart.

Ich gehe davon aus, daß ich eigentlich die meiste Sache in dem Programm selber reinmachen

Also das Programm, das kriegt natürlich nicht irgendein Aßembler sondern die Befehle werden entsprechend in den Speicher in den festen Speichern untergebracht

Problem ist ich kann zwar die Eingänge beliebig miteinander verbinden, aber nicht die Ausgänge

Wenn alle Ausgänge null sind, und beim anderen Bauteil auch alle Ausgänge null sind, kann ich sie trotzdem nicht miteinander verbinden. Ich kann die Ausgänge vom Bauteil niemals einfach miteinander verbinden.

Es gibt aber einen unterschied ich kann die Eingänge miteinander verbinden

Das Problem bei der Sache ist, wenn ich zum Beispiel einen Stapelzeiger habe ein Programmzähler und so weiter dann werden wir mehr Taktbefehle nicht erspart bleiben

überhaupt ist die Sache deswegen schwierig, weil ich jetzt einen Speicher habe einen festen, der als dekodieren infrage kommt für das Befehl

Gut okay das ist das. Eine des Problem ist einerseits noch ein anderes Problem mit dem L DI, d.h. Load immidiatly heißt ein Direktwert Laden. Das wird sehr schwierig in den Arbeitßpeicher, wenn wir keinen Zugriff umgekehrt haben. Außerdem brauchen wir ja sozusagen die Auswertung unser Status Bits

Nicht nur weil wir haben zum Beispiel bedingte Sprungbefehle die müßen einen Vergleich anstellen wie auch immer d.h. das Datum im Arbeitßpeicher muß auch in jedem Fall sozusagen mit dem Decoder verbunden sein in irgendeiner Form

Das ist das eine Problem dann haben wir die Steuer Wort an der ALU des Steuer Wort an der AL U wird auf jeden Fall mit dem Festwert Speicher dem Dekoda verbunden

Meine Idee ist jetzt für den Anfang ist einfach so zu machen. Ich mische das Programm also das eigentliche Programm vom Nutzer und worüber er nicht unbedingt Bescheid weißt, ist die Software technische Aufgabe mit dem dekodieren

Des geht deswegen bei der einerseits den aßembler später haben für den Nutzer

Es gibt Maschinen, die es nicht anders machen zum Beispiel VLIW very large Instruction Word

Okay, wenn wir das haben, dann müßen wir darüber nachdenken. Folgendes Problem.

Wir lösen die Sache einfach so wir nehmen wenn wir den Arbeitßpeicher redundant haben können wir auch den Festwert Speicher und dann nehmen

Wir schalten einfach an beide Eingänge sowohl einen Festwert Speicher als auch einen Arbeitßpeicher

Das Problem bei der Sache ist zum Beispiel, wenn wir von dem Register auf

Also dem Register die Im Arbeitßpeicher untergebracht sind auf dem festwert Speicher zugreifen müßen. Das ist aber kein Problem weil gucken Sie mal in ein Register bei einer 8 Bit ALU paßen maximal 256 Werte, doch der Eingang an einem Festwert Speicher sind zum Beispiel 13 Bit A0 bis A zwölf

Die unteren 8 Bit, da läßt sich das im Programm sozusagen aufeinander abbilden. Die null wird auf die Null abgebildet in der Adreße und so weiter. Das ist allerdings Aufgabe der Software das macht alles die Software in dem Festwert Speicher

Die 8 Bit werden auf die acht Bit abgebildet das klingt nach einer Riesen Platzverschwendung ist es allerdings nicht aus folgendem Grund, weil oben bleiben 3 Bit und die sind entscheidend. Mit 13 Bit kann ich

8 kB pro problemlos adreßieren davon 256. Das ist kein Problem. Jetzt kann man sehr lange darüber nachdenken, was das alles zu bedeuten hat. Das ist Aufgabe der Software und wird am Computer entwickelt und kann sogar stückweise entwickelt werden.

Ich mache einfach folgendes. Ich schalte die Sache in Reihe. Ich nehme die Arbeitßpeicher, die Arbeitßpeicher gehen in den Festwert Speicher Die Festwert Speicher gehen in die beiden Eingänge der ALU.

Der Ausgang der ALU geht wiederum in beiden der Arbeitßpeicher die wiederum in beide der Festwert Speicher gehen, so baue ich die Sache und einer der Festwert Speicher ist mit dem Steuer Wort der AL U verbunden

Mehr muß ich nicht machen und der Rest ist Software. Ich muß das entsprechende Programm auf das eeprom unterbringen

Diese Maschine ist sogar sehr leistungsfähig, sehr, sehr leistungsfähig auf große Ebene ausgebaut mit 64 Bit ALU und so weiter. Wir müßen daran denken, eine Schwierigkeit bei Prozeßoren im Allgemeinen ist

Daß wir darauf achten müßen eine Struktur auf einem großen Halbleiterbaustein, also auf einem Chip muß quasi sehr gleichmäßig sein. Durch diese Struktur, die ich jetzt erschaffen habe, schaffe ich eine extrem gleichmäßige Struktur. Den Speicher sind deswegen so voller sozusagen. Können deswegen so groß sein, weil sie eine gleichmäßige Struktur haben

Die Sache, wie sie ist, würde mich nicht daran hindern, es trotzdem mit einem externen Speicherbaustein zu verbinden, also auf moderner Ebene auf großer Ebene angebracht

Den Rest, den ich hier tun muß. Allerdings für diese Sache ist die Software selber, indem ich den sozusagen Festwert speicher. Entsprechend programmiere die sozusagen

Die sagen wir sozusagen Grenzen dieser Sache. Also, was ich beim Programmieren zu tun habe, legt die Hardware auf doch es gibt eigentlich diese Grenze nicht entsprechen. Entsprechend deßen, was die Sache ist, läßt sich alles quasi damit lösen, was bisher unsere Computer auch auch können. Alles und es ist einfach die Frage, was ich mir beim Programmieren einfallen laße. Also es ist nicht einfallen laße aber die Grenzen sind da, aber sie sind in diesem Rahmen da, daß sie komplett das komplett alles machbar ist was beim Computer machbar ist und alles was ich zu tun habe, muß ich jetzt mit der Software lösen d.h. ist die Maschine einmal aufgebaut. Ich muß zwar die Speicherbausteine immer wieder rausnehmen zum Programmieren, die festen und trotzdem es gibt diese Grenzen an sich nicht. Ich muß nur alles, was da zu tun ist, muß ich Software technisch lösen sowie beim Programmieren immer die erste Programmierstunde. Dabei mußte ich immer sozusagen gucken. Wo sind Anfang und Ende die Grenzen beim Programmieren und diese gesamte Sache ist definiert also die Grenzen liegen nicht nur in der Hardware aber wo die Grenzen bei der Hardware liegen etwa die Unterteilung der Schritte das ist das Problem. Kann ich mir jetzt beim Programmieren drüber nachdenken, wo die sind?

Muß alles mit der sozusagen Software selber gehen also dann gibt es also mit der Hardware Software Hardware Algorithmus heißt das also ich habe die Festwert Speicher7 die gehen beide direkt in die Eingänge von der AL U und die Festplattenspeicher gehen direkt an das Steuer Wort der ALO und diese gehen wiederum mit den Ausgängen an die Arbeitßpeucher da die Daten breite ein ist, kann ich drei, 8 Bit ist, kann ich folgendes machen wie gesagt ich übersetze das und noch etwas. Ich kann jetzt jederzeit sozusagen gleich fertig. Kann jederzeit auch von dem einen in das andere übernehmen. Das geht ohne Problem zum Beispiel addieren +0 heißt sozusagen, daß ich den selben Wert nehme. Der kann dann wieder im anderen Land so fertig das mache ich heute wie gesagt er ist

nicht viel zu tun Hardware technisch zum basteln mit den Händen

Schaltpläne datasheets hier... und:

Hier ist leider HC mit LS zusammen mal gucken ob es tut, elkos an den Eingang der Sache...

Das ist eine 74181

ALU einsetzen + Register Input + gucken wir mal + ...

Ich baue den Prozeßor. Jetzt weil die ALU kamen ja gestern per Post. Danke

Und ich baue den Prozeßor jetzt und es geht ohne Registers Satz und ohne multiplexer, was sogar viel einfacher ist aus eigenem Grund, weil ich einfach vier Register nehme

Ich mache einen 8 Bit Prozeßor. Also zwei nebeneinander zwei, 4 Bit Register.

Und 16 Bit Adreßen damit kann ich vielleicht sogar ein LCD ansteuern

Aus dem Grund, daß das einfache geht, weil ich vier Register nehme das ist fürn Anfang glaube ich nicht ein bißchen wenig weil dann nehme ich einfach eine unäre Kodierung

Also

0001 Register null 0010 Register eins 0100 Register zwei 1000 Register drei

Wenn ich das habe, kann ich auf multiplexer verzichten und auf Registers Satz und so weiter, ja und dann mußte das gehen

Nicht nur das

Ich fange jetzt einfach mal an

Schwieriger ist es mit sozusagen

Dem nachladen ich denke ich mach erst mal ne 4 Bit Architektur

Ich nehme 4 Bit ohne das LCD anzustellen. Ich hab auch ne gewiße Anzahl an Platinen.

Und man muß drüber nachdenken. Der Speicherbaustein hat 8 Bit Ausgabe.

Dann ist es so bequemer

Wegen der Adreßen halt, dachte ich. Bei 16 Bit ermöglicht Doppelte.

Aber man kann auch drüber nachdenken, daß ich halt entsprechend im Speicher Baustein ich weiß nicht wie viel meine jetzt am zwölf Bit Eingang Adreße und acht bitte Ausgang des reicht sozusagen

Für sozusagen für sozusagen die Befehls für die Breite des Befehls und kann man ja drüber nachdenken aber ja

Ach, da gibt es ein Trick. Ich nehme einfach ein fünftes Register und es ist das Output Output/input Datenworten am I/O Port

Das macht sing Sinn, ein Register, damit ich eben auch die Bits

An darstellen kann

Gut, ich habe keine tri state

Man kann ja die Eingänge parallel schalten, aber nicht die Ausgänge. Ich habe keine tristate

Dann genügt es mit Version mit zwei Registern, aber drei Registern

Also einem Output Port was allerdings in Register reingeht

Was keine Eingabe ermöglicht

Und an den Registern ist ja sozusagen pin übernehmen, also Input enable

Und wenn ich deshalb dann kann ich die anderen Eingänge parallel schalten

Aber dann brauche ich die Tri State und nicht und dann brauche ich nur zwei Register und kann nur zwei haben für die Eingänge an der ALU das ist auch besonders einfach zu verhalten. Damit fange ich jetzt an

Im zweiten Teil kommt ein Arbeits speicher dran an den selben Prozeßor

Das ist dann keine risc machine

So oder so nicht, weil bei den einfachen befehlen ist es für die zwei Register mit den üblichen Operationen mit einer 4 Bit ALU

Durch das sozusagen adreßieren im Programm Speicher , Harvard Architektur mindestens 8 Bit plus dem Arbeitßpeicher gibt es dann auch zwei also Befehle mit der Breite 2 Byte

Oder man kann es sogar so machen es sind alle befehle so lang nur die enthalten dann nicht so viele Information und trotzdem 4 Bit zum Beispiel dann kann man sozusagen im zweiten Teil bequem für den RAM unterbringen

Und dann hat man immer noch load und Store

Was wichtig ist und man kann ein Stapel sehr gut implementieren

Nein, für den Befehl Zähler nimmt man immer +2, indem man noch den übertrag am addierer nimmt irgendwie und kann gucken aber das überlege ich mir noch, daß man dann sozusagen ja gucken wir mal weil ich nehme nur zwei Register für den Befehl Zähler

Nein, das kommt dann in einer späteren Version. Dann laßen wir den Datenspeicher am Anfang raus.

Das sieht man das wird so kompliziert, weil auch ich brauche ja dein Register dann mache ich das anders

Also das wird dann zu kompliziert mit den Bausteinen, die ich hab wird es zu kompliziert und ich mach ne reine vier Bit Architektur ohne RAM bis auf für die Befehle selbst die sind 8 Bit

Und ich mache eine reine, solche Architektur

Zum Testen, und das andere mache ich mit einem GAL

Darauf einige ich mich, weil ich brauch ja auch sozusagen deine kompliziertere Schaltung, dann brauche ich noch tristate

Zum Glück habe ich ja GAL und ein Programmiergerät dazu, und da muß ich genug Programmierarbeit leisten daß man noch sagen kann, daß ich eigene Sache, weil eben ist es ja spannend an einem GAL und dann denk ich mir, wenn ich aber den Arbeitßpeicher integriere, dann sollte ich auf jeden Fall noch irgendwie tristate holen und vieles weiter und das, was ich nicht leisten kann, gerade wenn's um den Arbeitßpeicher so weitergeht. Dafür nehme ich dann ein GAL oder mehrere

Laße es jetzt bei den zwei sozusagen Registern im Registers Satz 4 Bit

Plus sozusagen dem Daten Port der nur Ausgabe kann und damit fange ich jetzt an, weil das ist relativ einfach geschehen

Das kann man sozusagen als Mikrogramm steuerwerk bezeichnen, wobei der Unterschied halt ist, daß es ein Programmzähler hat

Also, der Computer hat ein Speichern, Datenspeicher, Harward oder von Neumann

Das hat kein Datenspeicher, aber es ist ein damit, 1 Mikro Programm steuerwerk, ein einziger Unterschied und es ist kein Unterschied, 1 mikroprogramm steuerwerk eben mit externen Zähler also wo der Zähler nicht im Befehl ist Text übrigens kann ich auf dem Tisch auf dem auf der sieben Segmentanzeige trotzdem ausgeben

Ich kann einen Text ausgeben und es kriegt ein extra Grafik Display, nämlich ein B CD nach sieben Segmentanzeige Dekoda das hat es dann trotzdem. Also ein Grafik Display und Text kann ich ausgeben, indem ich sie sozusagen in den Befehlen implementiere. Also den Text kann ich als Festwert einbauen und dann kann ich den kann ich trotzdem eine sinnvolle Zeichenkette außer AB CDE ... kann ich trotzdem ausgeben

Also Mikroprogramm steuerwerk mit eigenständigen oder extern externen Zähler und einem sozusagen Grafik Display mit BCD zu sieben Segmentanzeige Dekoda jetzt fange ich an

Der Unterschied es gibt noch ein es gibt noch einen Unterschied. , es gibt noch einen unterschied zum Mikrogramm steuerwerk zum Rhein. Ich nehme nämlich ein befehls Dekodierer weil die Befehle aus dem Programm Speicher stecke ich da rein und es ist einfach wieder ein Programm Speicher der über die Adreße, sozusagen des Eingangs Wort des Befehl selber und dann kommt der entsprechend dekodiert Befehl also die Steuer Wort wieder raus und es macht es zum Computer. Ohne Arbeitßpeicher würde ich sagen, weil um dieses Befehl dekoder oder geht es dann trotzdem?

Und es hat dann einen Vorteil. Des ist dann einfach genauso wie der Programm Speicher Re Programmierbar. Weil ich eben überhaupt den dekodieren jedes Mal neu einrichten kann, wenn mir die Befehle nicht gefallen, ohne was an der Hardware zu ändern. Das ist jetzt Planungs Aufwand genug und damit kann ich jetzt anfangen.

so, jetzt laße ich es dabei...

und fertig...

ich kann diesen doppel emitter so nicht verstehen...

ok. dann habe ich es so gemacht, simulator geht nicht. dann nachher in ttl gattern, auf der platine...

Jetzt fangen wir an, heute Morgen gleich loslegen, mal wieder den Stift in die Hand nehmen

Und zwar TTL Ausgangstufe mit Transistor gegen Taktstufe

Erste übung auf Papier

Zweite übung am Computer mit Tabelle

Langrage Interpolation von neulich weiter mit Tabelle, aber sonst geht's nicht

Und wenn wir dann fertig sind dann TTL bauen soweit vorhanden plus Vorarbeiten mit Diagramm nämlich soll man sagen, ASM aber in dem Fall Flow Chart nämlich habe ich überhaupt Dioden und habe ich Widerstände, nämlich ich brauch 1 Kiloohm

D.h. den Prozeß hier versuchen in sozusagen Modell zusammenzufaßen also prüfen sozusagen sind entsprechende Dioden vorhanden. Entsprechende Widerstände

Das ist sozusagen Teil der Sache und wenn keine Widerstände, da keine Diode da entsprechend dann Sache gescheitert

Gut, jetzt diese drei Dinge. Erst mal fang gleich an erst Kaffee

Ich bin begeistert, aber der Feldeffekt Transistor zeigt scheinbar seine Wirkung. Ich werde eine Kaffeepause machen und es gleich vorstellen.

Was halt mein Problem ist ist ich hab jetzt hin deswegen war ich heute auch beim Baumarkt. Die haben das leider alles nicht. Ich hab mich auch entschloßen an die Straße zu setzen weil naja es tut auch mal gut. Manche Leute gucken ja zur Entspannung Fernsehen. Ich gehe jetzt mal in die Straße und guck da man sieht das hilft dann auch manchmal trotzdem. Ich hab beim Baumarkt gefragt oder nicht gefragt ob ich hab geguckt haben die solche Kabel haben sie leider nicht. Muß mich jetzt wieder bestellen müßte ich hab ich nicht genug die hier reinpaßen und deswegen wird's jetzt auch schwierig weil ich muß ja immer noch meßen wie viel da läuft

Ich bin begeistert, aber der Feldeffekt Transistor zeigt scheinbar seine Wirkung. Ich werde eine Kaffeepause machen und es gleich vorstellen.

Jetzt zeige ich mal den Feldeffekt Transistor wie ich mir den vorstelle in Wirklichkeit hat mich beschreibe das erst mal der vier Pole eigentlich fünf Pole

Es gibt nämlich zweimal die Maße für das Input Signal und das Output Signal und eben halt die Signale selber. Die sind vier Pole. Allerdings ist es Input Signal ja Steuern Signal beim Feldeffekt Transistor gibt es diese Maße so an sich trotzdem weil sie wird mit der Maße des sozusagen gesteuerten Signals verbunden. Trotzdem ist es eigentliche Signal zum Steuern des Inputs.

Eigentlich laut Buch ich habe ein Buch vom technischen Gymnasium. Das hatte ich beim technischen Gymnasium mitgenommen, war pink und steht an der Straße oder stand. Ich hab das inzwischen neu gekauft. Allerdings ist es ne grüne Version da ist nichts für Informationstechnik aber ist das selbe Buch ist von der Handelskammer oder so zur Ausbildung ich hatte es in pink oder wir hatten es in pink gehabt sie so nicht mehr gefunden aber war auch Informationstechnik drin aber grün hab es jetzt gut da ist auch ein bißchen Informationstechnik drin jedenfalls

Muß man sich die so vorstellen Feld Effekt Transistor funktioniert anders. Er funktioniert anders und deswegen vermute ich, es würde verschiedene Kristalle tun. Ich möchte darauf hinweisen auf folgendes.

Eine Germanium Diode sitzt meistens in einem Metallglas. In dem Metallglas ist ein kleines Germanium Plättchen und quasi mit einer Spitze aufgebracht. Ein Draht aus Aluminium. Meist. Dadurch entsteht sozusagen die Germanium Diode A.

Des Weiteren, hier ist es reine Germanium. Im Spiel kann man sich folgendes vorstellen. Ich hatte einen Kosmos Baukasten bestellt. Die Kristalle werden über Waßer gezüchtet und entstehen im Waßer. Trotzdem sind es Kristalle. Ich denke sie laßen sich wie eine Legierung einfach verunreinigen. Man muß sozusagen im Periodensystem mehr oder weniger nur auf sozusagen die Außen Elektronenzahl im Periodensystem achten gut dann könnte man zum Beispiel vielleicht Kohlestaub hinzufügen das wäre eine Möglichkeit.

Und ich weise noch darauf hin, daß ja sozusagen Legierungen für Goldschmiede machbar sind und daß es öfen im Internet gibt. Ich mache auch gleich eine Zeichnung. Trotzdem weiß ich darauf hin, daß ein Feldeffekt Transistor anders

funktioniert

Als zum Beispiel ein bipolarer Transistor

Wenn wir den Kristall und dann sagen wir, das ist der größte Klumpen an Transistor. Wenn er tut, den wir je gemacht haben. Wir stellen uns einen Stromkreis vor.

Wir haben ein schwarzes Kabel und ein rotes Kabel an der Spannungsquelle. So, jetzt machen wir einen Stromkreis. Wir haben die Spannungsquelle und inzwischen einen Verbraucher

Also, wir schalten in Reihe Spannungsquelle Verbraucher und dann, wie gesagt, diesen Kristall

Ist es ein Stromkreis, auch sozusagen drei Elementen, Kristall, Verbraucher und Stromquelle

Wir können uns einen Zylinder vorstellen, ein Kristallzylinder und sozusagen im Querschnitt oben und unten weiß nicht wie d.h. legen wir das schwarze und rote Kabel an und dann fließt es durch den Stromerzeuger und durch den Verbraucher oben und unten. Legen wir also so in der Mitte von der Höhe, sozusagen im Zylinder legen der weiße Kabel an. Die haben beide den selben Pol die sind an sozusagen VCC ebenso angeschloßen, einfach um die Sache zu regeln und es ist sozusagen oben und unten und um die bilden sich sozusagen ein Feld also um diese Punkte ein konzentrisches Feld

Wir haben quasi im Mittelpunkt oben das weiße und unten des weiße sozusagen Kontakt dann wie beim Sender sozusagen die Ringe, die sich darum ausbilden. Dabei können die mir nach innen gehen. Also mehr zu den Rändern von dem Kristall dann ist das Feld kleiner oder weiter ganz nach innen zum sozusagen Mittelpunkt von dem Zylinder zu der senkrechten und dann wird das Feld stärker also die gehen auseinander und zueinander und dann bildet sich der Feldeffekt Transistor das möcht ich jetzt ausprobieren und ich weiß auch inzwischen wie ich das mache. Ich habe leider keinen guten Klemmen. Also ich sollte das irgendwo reinklemmen den Kristall aber ich versuche den Strom durchlaufen zu laßen. Ich mache vorher eine Zeichnung und ich Verweise noch auf das Buch, von wo ich den Feldeffekt Transistor erklärt gekriegt habe, was darauf hinweist, daß ein Kristall hier genügen würde, ist folgendes

Zunächst einmal habe ich den Widerstand gemeßen, und der Kristall konnte durchaus Strohm leiten oder nicht. Dabei zeigt er auch ein nicht 100-prozentiges Verhalten. Der andere Teil bei der Sache ist ja sozusagen, daß dieser Kristall entsprechend dem, was sich zum Beispiel an Strohm drauflege oder nicht sich anders, sich einfach als kristallfeilt wir werden sehen und notwendigen falls paßen wir die Sache an. Ich bin mir sicher, wir werden das Problem lösen, und wir werden auch den Kosmos Baukasten zur Hilfe nehmen, zum züchten von Kristallen wir haben unser Leben lang Zeit. Daran müßen wir denken.

Ich habe einen Versuch, Experiment

Drei Poltheorie oder wie nannte man das?

Mir fehlt auf, wenn man zwei drähtchen an einem Germaniumplättchen anschließt

Hat man eine Germanium Diode? Doch wie soll man entscheiden, welche der beiden Richtungen vorgeht? Man sagt, wo die Spitze auf das Germanium trifft glauben kann ich das nicht ganz?

Trotzdem eigentlich ist ein Transistor mit vier Polen unterzogen. Eine gemeinsame Maße

GND für Signal Quelle und ein Maße für das Ziel von der Quelle das Eingangßignal und am Ziel ebenso an der ausgangsvariabel Ich habe das Meßgerät angeschloßen dann habe ich darüber nachgedacht und ich habe einen dritten Dritten genommen. Das wäre eine drei Poltheorie könnte man ja sagen ist das?

Könnte. Man besteht darin, daß ich jetzt entweder von Maße oder dem Signal ein verdoppelten, sozusagen ein übermaß des einen schaffe

Dann findet die Meßung des Widerstandes mit der Batterien in dem Meßgerät statt doch beim Widerstand meßen darf kein Strom fließen

Der nicht von dem Gerät ausgeht. Daraufhin ist mir aufgefallen, daß ich

dann kann sich diesen Kristall vorstellen, indem ich sozusagen drei Aluminium Verbindung mache etwa indem ich es irgendwo einspanne. Daraufhin ist mir aufgefallen, daß die Maße ja aber eigentlich von dem Eingangßignal und dem Ausgangßignal gleichzeitig er rührt. Ich probiere es mal mit einer Versuchßchaltung

Drei Pol. Transistoren sozusagen aber das sind gleichzeitig vier Pole, Transistoren Source , drain Gate. aber auch noch Bulk. Sind Feld,effekt Transistoren also genau das was ich beschreibe durch sozusagen den übermaß einer eines sozusagen. Ich habe ja stellen wir uns im Dreieck Modell am Computer vor unseren schönen 3-D. Darstellung hier ist ein spannendes ein möglichst ohne des Kristall zu beschädigen. Sozusagen mit drei Aluminium Dingern drei Aluminium sagen wir ordentlich mit Folie, die wir zusammen haben so in diesem geometrischen Form dreieckförmig ich weiß nicht wie heißt das Pyramidenform und dann leiten wir das durch dann haben wir sozusagen sagen wir mal ökonomisch ein Wort Monopol geschaffen d.h. vielleicht ein über übermaß von Feldern und das ist ja eben der Feldeffekt Transistor

Dann gibt es die elektrische Feldstärke die elektrische Feldstärke ist. Gleich herrscht zwischen den Platten. Ich nehme hier eine große Platte. Ich hänge sie in meinem Zimmer auf und hänge hier eine große Platte in meinem Zimmer auf. Dazwischen an die Decke mache ich einen Faden und hänge ein Kügelchen runter. Das Kügelchen ist wie eine Perle und ich tue es allerdings aus Metall nehmen. An den Platten lege ich eine Spannung an. Das Kügelchen mag in Cent Krankaratsgruppe gemeßen sein Cent, Kran, Karat, Scale Skrupel wie Skrupel. Mensch so also gut und darin Markt ist gemeßen sein das Gewicht dann mache ich mal eine andere als mit dem metrischen Maße. Dabei bewegt es sich sozusagen zwischen den Platten durch die Ladung ein wenig nach links oder rechts und ich tue darunter 1 cm Maß. Da ich allerdings nicht metrische Maß bevorzuge, nehme ich Punkt Linie soll gleich Finger handbreit Fuß dementsprechend wie es sich bemerkt hat ist das eine entsprechende Länge und groß U durch D ist die elektrische Feldstärke? Das entspricht sozusagen der Spannung durch die Länge. Also das Feld ist sozusagen die Spannung durch die Länge doch wir erleben es ein anderer Stelle durch die Fläche. AAD kann die Länge sein doch mal A gleich groß. A also klein mal klein A gleich A ist die Fläche und das bezieht sich auf OD die elektrische Flächenladung Dichte oder Verschiebung Dichte wären das andere die Feldstärke ist und dann haben wir folgendes. Wir teilen jetzt die elektrische Flächenladung Dichte beziehungsweise Verschiebung Dichte durch die Ladung groß oder den Strom Q also die Ladung Q unter den Strohm I anders als du bei dem bei der Spannung bei bei der elektrischen Feldstärke anders als bei Magnetismus geht es eigentlich um U, weil die magnetische, ich weiß nicht Flußdichte oder Feldstärke nein, das ist komplementär, also die magnetische Flußdichte ist rot. I durch D sozusagen groß, nicht rot. I durch D. Aus dem folgenden Grund ist das die Flußdichte und nicht die sozusagen Feldstärke, denn wenn bei der

Feldstärke bei der Spannung E gleich U durch herrscht, dann ist doch logisch, daß wenn sozusagen die Flächen ungs Ladung Dichte wiederum A durch Q ist, dann ist es doch logisch, daß auch das Gesetz hier komplementär ist d.h. das ist die magnetische Flußdichte die sagt I durch D also die Feldstärke. übrigens eilt die Spannung dem Strom voraus und deswegen kommen wir zu solchen Dingen, wie wir Widerstand wie das wirken. Das hat er hier falsch gemacht das wirken der blind Widerstand also der blinde und der Schein Widerstand entsprechend ist aber die Spannung dem Strom um 90 voraus oder zurück und so weiter

Also wir können ja ne Reihe machen die Fibonaccireihe, das geht los mit dem Element A0 gleich eins und A1 gleich eins und dann rechnen wir das also F von N ist gleich von N -1+ von N -2 beziehungsweise N +1 ist gleich von N plus F von N -1 und dann ist A2 gleich A0 plus A1 und AN +1 ist gleich AN plus AN -1 und das heißt ich probier's jetzt aus dem Kopf A0 ist gleich 1A1 ist gleich eins und A1 plus A0 sind gleich 2A2 ist gleich 2 A1 ist gleich 1A1 plus A2 sind gleich 3 A2 ist gleich 2A3 ist gleich drei A4 ist gleich 55 ist gleich fünf nein A4 ist gleich 5+ A3 ist gleich 3A5 plus A4 ist gleich A6 ist gleich 8 A6 ist gleich 8 A5 ist gleich 5 A7 ist gleich A5 + 9 ist gleich 5 + 8 ist gleich A6 plus A5 ist gleich elf A6 vermute ich oder A7 ist gleich elf A6. Ich bin jetzt mit dem in die C durcheinander ist gleich 8A8 ist gleich 19 vermutlich ich weiß aber jetzt grad auch nicht und rechnen wir mal woanders weiter und verzetteln uns nicht.

1 + 1 gleich 21 + 2, gleich 31 + 1 + 1 gleich 31 + 3, gleich 41 + 1 + 2 gleich 41 + 1 + 1 + 1 gleich vier Pause. Ich mach zum Kopf weiter, hoffe ich

hlich läßt sich der Marker hervorragend von diesen Stück herunter waschen was aus Kunststoff ist dann holen wir uns mal ein Schwämmchen und ein Waßer und dann können wir das immer wieder putzen und ja mit der Kamera daß die gewißen Abstand Fotos macht's, können wir immer noch einrichten. Ich probier's mal aus, jetzt kann man am Stück üben.

Na gut für heute Abend eine Alternativlösung. Wir können es uns ja außuchen. Erhalten wir das Papier oder wischen wir es von der Wand wieder weg oder von der Tafel oder wie auch immer der Unterschied ist ja wir haben ja jetzt unter Fotoapparat Smartphone. Ich mach ein wenig Pause. In wenigen Minuten hab ich vor dieses Papier da in den Schrank zu häften egal wie und einen ersten leisen Versuch zu machen, der dann auch erhalten sein wird auf Papier nämlich zum Beispiel üben wir die gegen Taktstufe der ttl Und weiteres vielleicht ökonomie oder was auch immer Begriffe, gucken wir mal

So ein white Board habe ich nicht gefunden das hätte mich auch gewundert.

Ist auch nicht so dringend notwendig aber diese Filzstifte, die sind für Whiteboards und damit probier ich gleich mal einfach aus das typische ttl muß mich noch um meine Mutter kümmern

und die Musik ab zu schalten, die im laden laeuft. trotz oder mit ohne frieden oder gerade mit.

und zu lernen ankuendingskommando ausfuehrungskommando... 1.) Trupp 2.) Gruppe = Rotte. 3.) Zug 4.) Staffel ... Emergency! System is unusable mit oder ohne drögen! ohne Musik im laden!

Also ich hab jetzt folgende Erkenntniße im Internet sind mehrfach ttl Gatter Beschreibung natürlich vorhanden Das erste, worum es mir jetzt geht. Ich will eine CPU bauen und zwar mit den existierenden TTL. Gattern. ALU74181 Nach dem beschriebenen da hab ich ja zehn bestellt. Ich hab zehn von den Baustücken Bauteilen gleich bestellt. Ich habe ja gesagt, man könnte jetzt, wenn man das Mikrogramm steuerwerk nimmt einfach sozusagen daran. Eine AL U anschließen vorausgesetzt die Befehle im Mikrogramm steuerwerk sind die Befehle ja im Mikrogramm steuerwerk und es läßt sich einfach In einem eeprom unterbringen Der Unterschied ist

Folgendes zunächst mal ist da auch die Verzweigung drin, wenn ich für beide Seiten jetzt mal rein Mikrogramm steuerwerk Wenn ich, dann hab ich doppelt so viel Speicherplatz. Ich brauche doppelt so lange Befehle allein für die Adreßen. Alleine für dieses Konzept würde ich eine Lösung anstreben. Bei dem sozusagen. Alternative Adreße im hinteren Speicherbereich angelegt ist da es meistens viel weniger. Verzweigungen gibt, ist der hintere Teil im Speicher. Kleiner Dieses Konzept wurde ersetzt durch

Den Programmzähler Trotzdem haben wir ja gesehen, bei dem letzten Mikrogramm steuerwerk, was ich geübt habe Eigentlich und es gilt nur für den Fall, daß ich keine Alu habe, sondern sagen wir einen addierer subtrahier multiplizierer und fürs dividieren hätte ich hier wieder vier Entscheidungen Jetzt ist es so, das ist in der ALU drin. Typische Prozeßoren können zunächst vor allem addieren und subtrahieren. Das kann unser 74181

Okay, wen hat es kann dann muß ich ja nur das Mikrogramm steuerwerk, die Frage ist. Was machen wir mit den Registern Ich sag mal so, das Ding bisher einfach das abarbeiten, was in dem EP R OM drin steht und würde lustig vor sich addieren und subtrahieren und so weiter, dann nehmen wir noch Register Wir nehmen Register Und zwar einen Satz einen RAM den gibt es auch mit TTL 4 4

Worauf achten müßen wir nehmen zwei dieser Dinger, dann haben wir einen unten Bereich und einen oberen wegen dem Zielregister Das Problem ist es gibt

Speicher, die haben gleich zwei Ausgänge Das brauchen wir sozusagen für die Quelloperanten Das Problem ist, daß wir das bei diesem wahrscheinlich nicht haben. Wir machen aber was anderes wir nehmen einfach zwei Dann gibt es leider noch das Ziel und das Problem ist wir müßen das sozusagen obere Takt high und unter Takt Low nehmen und übernehmen. Da haben wir in der Vergangenheit einen Fehler gemacht, wenn wir es gemacht haben wie ich. Vielleicht erklärt das den buffer

Funktioniert trotzdem Smart funktioniert zwar auch trotzdem, wenn ich die Quelle anschließe und dann geht es durch das übergangßchalt Netz ist noch die Frage wenn ich in den nächsten Tag übergehen also runter abschalten wo bleiben die Daten am Ausgang mit gut Glück kommen sie zwar an bloß Ich hab auch manchmal gemerkt irgendwas machen die Schalt Werke falsch. Die Daten sind ja am Ausgang eigentlich weg Gibt aber ne Möglichkeit der Schalten Kondensatoren dazwischen. Möglicherweise ist das ein buffer Wir schalten einfach Elko dazwischen Und wenn dieses ich hab das Gattern nicht vor Augen aber wenn das Aktivierung für Eingang hat, dann können wir uns multiplexer und so weiter ersparen. Wir nehmen dann vier elko . Und der Witz ist, daß unser Prozeßor bisher nichts anderes tut Mikrogramm steuerwerk plus der ALU vielmehr ist ja nicht zu machen. Die Register sind quasi schon Zusatz Weil der aber kein Mikrogramm Schuhwerk nehmen wollen, nehmen wir noch ein sozusagen achtes Register dazu. Das muß aber lang sein, damit wir also mehrere nebeneinander schalten damit wir nicht die ganze Zeit nur 16 Befehle ausführen können wegen 4 Bit Vom Zähler her nimmt er das als Programmzähler ebenso also der nächste Befehl. Die Adreße des nächsten Befehls steht dann da drin dann können wir uns und es können wir kodieren wie wir wollen ist die Basis Architektur erst da sozusagen würde ich das mal nennen Könnt ihr im Mikrogramm Stellwerk unterbringen, was wir wollen? Wir müßen uns da keine großen Sorgen machen, nötigenfalls ein bißchen beim Programmzähler basteln des nehmen und übernehmen. Geht gleichzeitig mit dem Registers Satz dann muß die Sache fertig sein. Ich habe jetzt bei TTL geguckt Eigentlich haben die diese Gegend Takt Außenstufe des muß ich untersuchen. Ich hab auch Transistoren hier. Ich probier mal sowas dann irgendwann zu bauen. Ich übe jetzt erst mal wie das funktioniert.

Ich kenne immer noch nicht den Namen des Herstellers des ersten IC, aber ich habe ihn im Netz gefunden das ist logisch Im Labor hat das ja gemacht, zwei Transistoren untergebracht das lerne ich auch auswendig Und auf der Seite ich geb jetzt mal den Link nicht an, aber sie finden solche steht auch, wie man Halbleiter herstellt Auf modernen Dingern und da steht zum Beispiel der Name von dem ein Entwickler. Des muß ich auch auswendig lernen von dem, der das hergestellt hat und jetzt sind wir gleich mal ein paar Ideen gekommen. Also wie gesagt, man kann Halbleiter irgendwie wahrscheinlich mit Legierung machen. Hab jetzt noch andere Ideen das müßte man ausprobieren kann man zwei Dioden benutzen im Internet steht erst mal nein Sei nicht so einfach und vielleicht Vielleicht trotzdem entsprechend über die entsprechende Spannung und dann

steht da aber auch noch zum Beispiel. Man kann ja einfaches Germanium nehmen so ist ja ne Germanium Diode die sind ja aus Glas da kann man ja immer reingucken man nimmt einfach ein drähtchen ob das dann so einfach ist und da steht aber jetzt ja hier das Beispiel wenn man sehr nah aneinander unterbringt, daß man damit Germanium Transistor machen kann Gut, das könnte man ausprobieren und eine Groß Rechenmaschine könnte man dann zum Beispiel machen, indem man normale Transistor nimmt. Also was heißt normale die man im Handel erhältlich bekommt und dann sozusagen stufenweise ist halt dann so ne große Rechenmaschine Platine für Platine am besten zum stecken und dann kann man die, wenn man zu einmal gelungen ist stückchenweise sozusagen ersetzen durch eigene und hat man dann irgendwann so eine Schrank sag ich mal also Stehschrank Kein ganz großen, aber sowas relativ kleines, das wär ja ne Möglichkeit und der ist dann ja mehr oder weniger vollständig eigenständig hergestellt Wie gesagt, man muß auf diese Stufe achten, daß da am Ende zwei Transistoren sind. Und zwischen denen ist wieder ein Diode . Das kann ich heute im Experiment auch herausfinden.

Ich ziehe das auswendig lernen vor aber ich hab Transistoren da und wenn ich das gelernt hab, dann kann ich mal probieren, ernsthafte Gatter zu machen Ja, wie gesagt und ich hab gerade am Kontakt zu der Elektronikfirma bekommen und die das Paket mit den 74181 kam zurück wahrscheinlich weil an diesem Haus noch mal eine Adreße am Briefkasten nicht angegeben ist. Das ist natürlich ne gute Idee und dann bekomme ich die auch noch. Vielleicht erweitere ich einfach mal die Lieferung, daß dann noch diese RAM noch dazukommen und dann kann man mal gucken, ob man damit allmählich anfangen kann oder man kann auch erst mal nur das Mikrogramm steuerwerk nehmen und die ALU und dann mal gucken ob das vielleicht schon jedenfalls es führt etwas aus würde ich mal so sagen gut gucken wir mal

Ja, man kann einen funktionierenden Transistor (einen sogenannten Spitzentransistor) aus Germanium und feinen Drähten im Eigenbau herstellen, allerdings funktioniert das nicht, indem man einfach zwei Germaniumstücke aneinanderdrückt.Hier ist eine kurze Erklärung, wie das historisch und im Hobbybereich funktioniert:Funktionsweise und PrinzipDas Grundprinzip: Die ersten Transistoren der Welt (1947 bei Bell Labs entwickelt) waren sogenannte Spitzentransistoren (Point-Contact-Transistoren).Der Aufbau: Sie bestehen aus einem winzigen Plättchen aus dotiertem Germanium (als Basis) und zwei extrem dicht nebeneinander platzierten, spitzen Metalldrähten (Emitter und Kollektor).Der "Doppel-DrahtTrick: Man benötigt zwei feine Kontaktnadeln (z. B. aus Phosphorbronze oder geschärften Gitarrensaiten), die mit sehr geringem Abstand (Bruchteile eines Millimeters) auf die Germanium-Oberfläche drücken.Wie man es im Bastelbereich machtGermanium gewinnen: Viele Bastler öffnen vorsichtig alte Germanium-Dioden (wie die klaßische 1N34A), um an das darin enthaltene winzige Stück n-leitendes Germanium zu gelangen.Die Drähte: Als zweiter Kontakt wird ein spitzer Metalldraht neben den bereits vorhandenen Diöden-

Draht positioniert.Formieren („Burn-in“): Durch einen kurzen, kontrollierten Stromimpuls (z. B. durch Entladen eines kleinen Kondensators) über die Kontakte wird der winzige Halbleiterübergang im Inneren „geformt“(formiert), damit er den Verstärkungseffekt (Transistorwirkung) zeigt.

Erweiterte Signalliste erweiterte Liste der Signale. Nr. Name Aktion Bemerkung

```
4 SIGILL Core & Ende Ung"ultige Instruktion wurde ausgef"uhrt.
5 SIGTRAP Core & Ende Unterbrechung (Einzelschrittausf"uhrung).
6 SIGABRT Core & Ende Abnormale Beendigung.
7 SIGBUS Core & Ende Fehler auf dem System-Bus.
8 SIGFPE Core & Ende Problem bei einer Gleitkommaoperation, z.B. Teilung durch Null.
11 SIGSEGV Core & Ende Speicherzugriff auf unerlaubtes Speichersegment.
13 SIGPIPE Ende Es wurde in eine Pipe geschrieben/versuch mit O_NONBLOCK zu schreiben, aus d
14 SIGALRM Ende Zeituhr ist abgelaufen  alarm().
17 SIGCHLD Ignoriert Der Kind-Proze"s wurde beendet oder angehalten.
20 SIGTSTP Anhalten Der Proze"s wurde von Hand mit STOP angehalten.
21 SIGTTIN Anhalten Proze"s wollte aus einem Hintergrundproze"s der Kontroll-Dialogstation l
22 SIGTTOU Anhalten Proze"s wollte in einem Hintergrundproze"s der Kontroll-Dialogstation sc
27 SIGPROF Ende Timer zur Profileinstellung ist abgelaufen.
28 SIGWINCH Ignoriert Window-Gr"o"se hat sich ver"andert.
31 SIGSYS Core & Ende Ung"ultiges Argument bei System-Call.
```

Proceßed Through USPS Facility

```
Facility
Through
Throughput: Durchsatz
Distribution:
Released:
Customs:
Custom Cleared:
Shipping Partner:
Shipping Label Created:
Departed:
Arrived:
tracked:
feature:
Upon request:
purchase:
Busine"s customer gateway
Gateway
Privacy policy:
```

```
Terms of use
Careers
Carrier:
```

gebrauchswert wert warenwert tauschwert kennziffer kennzahl kennnummer use-value value commodity value exchange value indicator figure, identification number

characteristic value: kennwert Nennwert, Nominalwert: nominal value
goods: Waren the trade: handel cash: Kaße value of goods: Warenwert
cash on delivery: geld gegen ware
bip: GDP
groß domestic product: brutto inlandsproduct

```
Facility: Einrichtung
Facility: Logistics center / Lager / Einrichtung.
Through / Throughput: Durchsatz (the volume of goods proce"sed).
Distribution: Vertrieb / Verteilung.
Released: Freigegeben (ready for the next step).
Customs: Zoll.
Customs Cleared: Verzollt / Zollabfertigung abgeschlo"sen.
Shipping Partner: Versandpartner / Transportpartner.
Shipping Label Created: Versandetikett erstellt.
Departed / Arrived: Abgereist (Abgang) / Angekommen (Ankunft).
Tracked: Sendungsverfolgt / Nachverfolgt.
Upon request: Auf Anfrage
Purchase: Kauf / Einkauf
Busine"s customer gateway: Gesch"aftskundenportal
Gateway: Portal / Zugang / Schnittstelle
Privacy policy: Datenschutzerkl"arung
Terms of use: Nutzungsbedingungen
Careers: Karriere
Carrier: Spedition / Frachtf"uhrer / Netzbetreiber (je nach Branche)
Shipping: versandt

SIGKILL
SIGHUB
SIGBINT
SIGILL
\enc{verbatim}
Terminal: Das Abfertigungsgeb"aude (am Flughafen), der Endbahnhof, eine Datenstation/Konsole
Gate: Das Flugsteig (am Flughafen), ein Tor, eine Pforte oder im Bereich der Elektronik/Tech
Gateway: Ein "ubergang, ein Netz"ubergang (Computertechnik), ein Tor oder ein Zugangspunkt.
\begin{verbatim}
SIGHUP
SIGINT
SIGQUIT
```

```
SIGKILL
SIGTERM
SIGALRM
SIGPIPE
SIGILL
```

Nohup Idle: untätig
    Nohup unterdrückt das HUP Signal

```
Strg+C
Strg+Z

SIGINT
SIGSTOP
```

Signal Interrupt: Prozeß wirklich beenden, mit Strg+C

```
kill -KILL
kill -9

sind: extra

SIGINT
SIGSTOP

KILL
TERM
```

Systemsignale Gerätesignale Benutzersignale
    idle: Leerlauf
    müßig idle, pointleß, futile, otiose, of leisure
    untätig idle, inactive, paßive, dormant
    leer empty, blank, void, vacant, bare, hollow
    faul lazy, rotten, foul, idle, putrid, decayed
    träge sluggish, inert, lazy, indolent, lethargic, idle
    eitel vain, conceited, idle
    unbeschäftigt idle, unemployed, unoccupied, free
    außer Betrieb idle, dead, out-of-order
    nutzlos useleß, futile, vain, unneceßary, idle, unhelpful
    unnütz useleß, pointleß, idle, wasteful, valueleß
    tatenlos idle
    ruhend resting, dormant, stationary, idle, reclining, recumbent
    vergeblich vain, futile, useleß, fruitleß, unsucceßful, unavailing
    unproduktiv unproductive, idle, inefficient
    töricht foolish, stupid, misguided, vain, mistaken, idle
    bequem comfortable, convenient, easy, cushy
    still-liegend idle
    stillliegend idle

stillstehend stationary, stagnant, idle
bummelig idle, slow
faulenzen laze, lounge, idle, laze about, have laze
untätig sein idle
nichts tun idle
bummeln stroll, wander, loiter, dawdle, saunter, slack
Signale: Call-nr: Error: Devices/Interrupts:

```
Devices:
1.) TTY
2.) Keyboard
3.) Clock
4.) Wini XT
5.) Wini AT
6.) Floppy Disk
7.) Printer

Schnittstellen:
1.) V.24 (rs-232)
2.) Parallel  printer

XT: Extended: Old Version
AT: Advanced: New Version
Wini: Hard Disk Drive

XT: 13
Keyboard: 9

Error:
OK: 0
Error: -1

NERROR: 34
OK: 0
ERROR: -1

SIGNALS

HANGUP
KILL
QUIT
TERM
```

ASCII Codes und Signale

```
ACK: Acknowledge
NAK: Not Acknowledge
```

```
CAN: Cancel
ESC: Escape
DEL: Delete
ENQ: Enquiry
STX: Start of Text
SOT: Start of Transmi"sion
```

Transmit Receive Send
Prepare Acceß
Request Ready Clear
Carrier line
Terminal Set Gate Gateway
Enquiry und Request
idle: untätig
übersetzungen von idle Adjektiv
müßig idle, pointleß, futile, otiose, of leisure
untätig idle, inactive, paßive, dormant
leer empty, blank, void, vacant, bare, hollow
faul lazy, rotten, foul, idle, putrid, decayed
träge sluggish, inert, lazy, indolent, lethargic, idle
eitel vain, conceited, idle
unbeschäftigt idle, unemployed, unoccupied, free
außer Betrieb idle, dead, out-of-order
nutzlos useleß, futile, vain, unneceßary, idle, unhelpful
unnütz useleß, pointleß, idle, wasteful, valueleß
tatenlos idle
ruhend resting, dormant, stationary, idle, reclining, recumbent
vergeblich vain, futile, useleß, fruitleß, unsucceßful, unavailing
unproduktiv unproductive, idle, inefficient
töricht foolish, stupid, misguided, vain, mistaken, idle
bequem comfortable, convenient, easy, cushy
still-liegend idle
stillliegend idle
stillstehend stationary, stagnant, idle
bummelig idle, slow Verb
faulenzen laze, lounge, idle, laze about, have laze
untätig sein idle
nichts tun idle
bummeln stroll, wander, loiter, dawdle, saunter, slack Weniger anzeigen

```
Strg+C
Strg+Z
```

```
NO MEM
E NO ENT
```

ENOENT steht f"ur Error NO ENTry (oder Error NO ENTity) und bedeutet Keine solche Datei oder

E NO ENTRY:

Keine solche Datei, kein Eintrag nicht keine Environment

The ENOENT error is a common error me"sage that indicates a file or directory does not exist

Common Indicates
EPERM: operation not permitted
Operation not permitted No Entry

INTR
IO
NOEXEC
NOMEM
ACCE"s
BUSY
EXIST
NOTTY
PIPE

Strategie Operative Planung Taktik
Vertrieb, Strategie: Wie komme ich an mein Geld: Brezeln oder Staubsauger
Versand, Operative Planung:
Strategisch Staubsauger oder Brezel
Send: To start or initiate the movement of data from a source. Transmit: To move or carry the data signals acroß a physical or digital communication channel. Receive: To capture, decode, and acquire the data at the destination end
abort: abnormale Beendigung
abbrechen abort, stop, break off, terminate, quit, sever
abtreiben abort, float away, float off, be driven off course
verwerfen discard, reject, dismiß, condemn, quash, brush aside
abortieren abort, miscarry
entfernen remove, take off, detach, dislodge, take out, excise
fehlgebären miscarry, abort
abgehen leave, depart, come off, go off, exit, sail
die Schwangerschaft abbrechen abort
emergency: notfall alert: Alarm crit: kritisch
quit: aufhören
alert: alarm

SIGHUP, SIGINT, and SIGKILL are standard POSIX signals used in Unix-like operating systems
Here is the direct breakdown of their differences, behaviors, and standard use cases:

```
Overview Comparison

Signal Name Signal Number Trigger Method Catchable / Ignorable? Primary Modern Purpose
SIGHUP(Hangup) 1 Closing terminal, kill -1 Yes Reload configuration files or notify terminal
SIGINT(Interrupt) 2 Pre"sing Ctrl+C, kill -2 Yes Interrupt foreground proce"ses gracefully.
SIGKILL (Kill) 9 kill -9 No (Handled by OS kernel) Forcefully and instantly destroy an unres
```

eigenbemerkung:

1. Errors
2. Call Nr
3. Signals

Daneben gibt es

1. Errors
2. Call Nr
3. Signals
4. Interrupts (Interrupt Vektor)

typische Fehler

```
34 Fehler? NERROR 34
#define NERROR 34
#define OK 0
#define ERROR -1

// Fehler Allgemein: ERROR (-1) immer, NERROR, 34 Betrieb"system Minix
// OK: 0

/*
ENOMEM
EIO
ENOENT (Oft\dots)

E NO ENVIRONMENT (No Environment - keine Umgebung)
ENOMEM (Kein Speicher)
EIO (IO Fehler)
EDIV (Device Fehler)

Geraet, IO, Memory

\dots

SIGNALS sind kla"sischer weise:
```

SIGHUB
SIGKILL
SIGINT
SIGILL
SIG\dots

alle lernen

/h/signal.h
/h/callnr.h
/h/error.h

SIGHUB?? wu"ste ich nicht

Closing Terminal

Also: Konsole: Bash Schlie"sen
Ctrl+C (das beruehmte, womit man immer sein programm beendet ist:
SIGINT

also, das Ctrl+C, was man immer drueckt, ist SIGINT

Die Konsole selber schlie"sen ist SIGHUB

Dieses Ctrl+C ist -2,
und dann sagt man in Zukunft mehr (ich - sie vielleicht schon - zum lernen)
Ctrl+C sondern SIGINT

und beim Beenden der Konsole SIGHUB \dots

ok: SIGKILL
ist KILL proze"s beenden und zwar von au"sen\dots also wenn ich

pkill Proce"sname/Prog Name..

oder

kill -KILL Proce"sNr

Dann ist das

SIGKILL

Sighub - HangUp

automat08272026F.pdf.jpeg-127.jpgautomat08272026F.pdf.jpeg-128.jpgautomat08272026F.pdf.jpeg-

129.jpg

```
kernel, mm, fs, (h)

./kernel/proc.c
./kernel/main.c
./kernel/mpx88.s
./kernel/klib88.s
./kernel/memory.c
./kernel/stty.c
./kernel/floppy.c
./kernel/clock.c
./kernel/disk ?? wini\dotsc

./h/glo.h
./h/const.h
./h/type.h

.. (proc.??nein)

ok:

virtual
physisch/physical/physikalisch

clicks
bytes

vir
phys
clicks
bytes

vir, phys, clicks, bytes

map
vir
phys
len

vir phys len

mem_map
mem_vir
mem_phys
mem_len
```

```
struct mem_map {
   vir_clicks mem_vir;
   phys_clicks mem_phys;
   vir_clicks mem_len;
};

NR_REGS
NR_SEGS

11 REGS: ax bx dx cx sp bp si di cs ds "s

T: Text (Code): cs
D: Data: ds
S: Stack: "s

PIC 8259A: Programmable Interrupt Controller 8259A

Port: 0x20, 0x21, 0xA0, 0xA1 (0xa0, 0xa1):

0x20: Port
0x21: Mask
0xA0: Port
0xA1: Mask

X: Gift: 13: alt: wg. untere Nummer
X pa"st zu 13
XT: alter IBM PC XT

AT: neuer
AT: advanced\dots

#define XT_WINI_INT ?? 13
CLOCK: 8
KEYBOARD: 9

SYS_INT, ja

XT_WINI_INT
KEYBOARD_INT
CLOCK_INT
TTY_INT??
FLOPPY_INT\dots

trap divide interrupt
```

```
divide: abbildung auf den 0 vektor, eintrag in der interrupt vektortabelle, mit index 0 in d
trap: alle interrupts > 15, bedeutet: alle: die nicht am PIC 8259A sind
interrupt <= 15

suprise, trap, divide:
suprise: interrupt nicht erwartet

ok: botschaften:

sys_call: die einzigen system_calls die ausgetauscht werden sind:
SENDEN und EMPFANGEN, nachher in der
./kernel/proc.c?? nachher nachdenken

ok:
mini_send und mini_rec (interrupt und: sys_call)

in der
kernel.h?? oder proc.h <- zweiteres ist die

proc structure:

struct proc {

} procs [NR_TASKS+NR_PROCS];

darein kommt:

int p_regs [NR_REGS];
extern struct mem_map p_map [NR_SEGS];

usertime und systime
realtime time clocks ticks

jedenfalls:
usertime, system time vom proze"s
vom aufgerufenen kind

stack pointer, anfang, grenze
int pid

proce"s id;

..

also:
```

```
extern struct proc {
   extern int p_regs [NR_REGS];
   extern struct mem_map [NR_SEGS];
   int pid;
   // stackpointer, times\dots
} procs [NR_TASKS+NR_PROCS];

gut:
./kernel/main.c

gro"se routine: main:
MM, FS, INIT

Der Kernel selber wird durch INIT eher repraesentiert, was weitere proze"se startet

TASKS: Betrieb"system
PROCS: User

schleife:

und: einsprungsaddre"sen:

keyboard_int, clock_int, floppy_int, disk_int
sys_call divide trap \dots.

erst: in der schleife die segmente:
T, D, S: anfang und laenge
und weitere\dots

dann interrupts initialisieren

set_vec (

set_vec (keyboard_int, \dots)

ok:

mpx88.s:
klib88.s

mpx88.s

MINIX:
TTY_INT:
CLOCK_INT:
\dots
```

klib88.s:
hilfsroutinen:

dma_read
dma_write
lock
unlock
getchrome
video buffer copy\dots

proc.c:
mini_rec, mini_send, sys_call, interrupt

bei dem einen, werden proce"se gleich mit aufgebaut
bei dem anderen, nur die botschaften gesendet

diese sind caller, empfaenger und botschaft

dann kommen:
ready, unready, schedule\dots

ok\dots ich lerne heute mal auswendig, dann lerne ich das
spaeter weiter\dots
egal, wie vollstaendig, noch der ARM Cortex

ARM: Advanced Risc Machine

Advanced Risc Machine ldt.

ldt.: gmbh:

Linien:
A: Operating System, \dots mit allem
R: Embedded System
M: Microcontroller

ARM Cortex-M7
ARM Cortex-A7
ARM Cortex-R7

ok: Dann Versionen:

zunaechst: ARM7TDMI \dots what ever:

ARM: Architecture Application Level Reference Manual

TRM: technical Reference Manual

ARM, ARM, ARM

ISA: Immer: Instruction Set Architecture

ARM: Advanced Risc Machine
ARM: Architecture Application Level Reference Manual
ARM: Operating System, Embbeded, Microcontroller

Das sind Profiles

ok: Waynes?

ok:
Apple Computer, Arcorn und: VLSI Technology

VLSI: Very Large Scale Integration

VUSE: Very Ultra Super Extra

Small Medium Large

Memory Modell: x86: Tiny, Small, Medium, Compact, Large, Huge, Flat \dots

Frequency: Low, Medium, High Frequency
Very High, Ultra High, Super High, Extra High

Very Large, Ultra Large\dots

ok: Very Large Scal Intergration Technology

Apple Computer, Arcorn und: VLSI Technology: ARM

STMicroelectronics: Atmel (heute)

dazu: PHILLIPS (PX\dots?)
OKI, \dots

ARM Cortex-M7
ok: v4, v5, v6, v7

Thumb: daumen

bei dem proze"sor:
Thumb-2: Erweiterter Befehl"satz: 32 Bit

```
Thumb-Instructions: 16 Bit\dots

ok:

Graph:
(auf Gut Deutsch: User und System)

User-Thread, Kernel\dots
ok: wie rein und raus:\dots gerade nicht praesent\dots

ok:
16 Register
13 GPR: General Purpose Register

r13 \dots r0
Stack: Main Stack Pointer, Proce"s Stack Pointer
Link Register: Ruecksprungaddre"se
\dots noch was\dots

es gibt befehle, wie
ADD, ADC,
B: Branch, BC?? wie carry

flags sind generell: wie x86
sign, carry, overflow\dots (aehnlich)

atmega8 nebenbei gibt es ja noch neben
SIGN auch NEGATIVE \dots
ok:

dann gibt es:
3 Formen der\dots Interrupts, oder wie:

jedenfalls generell:

NMI: Non Maskable Interrupts
Siehe: PIC 8259A, 0x21, 0xa1

Nested Vectored Interrupt

Verschachtetele Interrupt Routinen, oder der Vektor??

dann gibt es:

ABI: Application Binary Interface
EABI: Enhanced ABI
```

```
Advanced Periphal Bus
Advanced High Perphormance Bus
Advanced

Port, Bus:

Serial Wire: Seriell (V.24, RS232\dots z.B.)

JTAG: Joint Test Action Group

Debug Schnittstelle: JTAG\dots

Debug Port
JTAG Debug Port\dots

Es gibt mehrere Einheiten:

MMU: Memory Management Unit
MPU: Memory Protection Unit
LSU: Load and Store Unit
ALU: Arithmetic and Logic Unit
Power Management Unit

Neben, Neumann/harward\dots:
I/O, CPU und Mem:

Peripherals, auf dem IC, Debug Interface, what ever\dots

ok: pause, ich mache aufgaben weiter\dots
```

Also bei mir war noch ein Fehler bei mealy und Moore drin, was das betrifft ist es hier zufälligerweise der Ziel Zustand war zufälligerweise in diesem Fall da stimmt was mit dem XOR nicht zufälligerweise ich dann sozusagen mit dem Zielzustand sozusagen die Ausgabe identisch deswegen änder ich mal die Ausgabe, daß das die 0 und 1 betrifft, das ist dann ins Konzept paßt

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach

import random
random.seed ()

V=[1,2,3,4]
```

```
V1=[1,2,3,4]
V2=[1,2,3,4]

print ("# (c) david vajda")
print ("# 08/27/2026")
print ("# python3, zustaende, vier zustaende")
print ("# \dots graph: schaltwerk, einfach")

i=3
while i >= 1:
 j1 = random.randint (0,i-1)
 j2 = random.randint (0,i-1)
 if i+1 in V1:
 yes1 = True
 else:
 yes1 = False
 if i+1 in V2:
 yes2 = True
 else:
 yes2 = False
 if yes1:
 V1.remove (i+1)
 if yes2:
 V2.remove (i+1)
 v1 = V1 [j1]
 if v1 in V2:
 yes11 = True
 else:
 yes11 = False
 if yes11 and (i > 1):
 V2.remove (v1)
 j2 = random.randint (0,i-2)
 v2 = V2 [j2]
 V1.pop (j1);
 V2.pop (j2);
 if (v1 != v2) and yes11:
 V2.append (v1)
 if yes1:
 V1.append (i+1)
 if yes2:
 V2.append (i+1)

 print (i+1, " (", v1, ",", random.randint(0,1), ") (", v2, ",", random.randint(0,1), ")")
 i = i - 1

v1 = V1.pop (0);
```

```
v2 = V2.pop (0);
print (i+1, " (", v1, ",", random.randint(0,1), ") (", v2, ",", random.randint(0,1), ")")

xdavidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 0 )
3  ( 1 , 1 ) ( 2 , 0 )
2  ( 4 , 0 ) ( 4 , 1 )
1  ( 3 , 0 ) ( 1 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 1 ) ( 3 , 1 )
3  ( 2 , 1 ) ( 1 , 1 )
2  ( 4 , 0 ) ( 4 , 0 )
1  ( 3 , 1 ) ( 2 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 2 , 1 ) ( 1 , 0 )
3  ( 4 , 0 ) ( 2 , 0 )
2  ( 1 , 0 ) ( 4 , 1 )
1  ( 3 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 1 , 0 )
3  ( 2 , 0 ) ( 4 , 0 )
2  ( 1 , 1 ) ( 3 , 0 )
1  ( 4 , 0 ) ( 2 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 3 , 0 ) ( 2 , 1 )
3  ( 1 , 1 ) ( 4 , 0 )
```

```
2  ( 4 , 0 ) ( 1 , 0 )
1  ( 2 , 0 ) ( 3 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 3 , 0 )
3  ( 4 , 0 ) ( 2 , 1 )
2  ( 3 , 0 ) ( 1 , 0 )
1  ( 2 , 1 ) ( 4 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 2 , 1 )
3  ( 4 , 0 ) ( 1 , 1 )
2  ( 3 , 1 ) ( 4 , 1 )
1  ( 2 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 1 )
3  ( 1 , 1 ) ( 2 , 1 )
2  ( 4 , 1 ) ( 4 , 0 )
1  ( 3 , 1 ) ( 1 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 1 ) ( 2 , 0 )
3  ( 4 , 0 ) ( 1 , 0 )
2  ( 3 , 0 ) ( 4 , 0 )
1  ( 2 , 0 ) ( 3 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 2 , 0 )
3  ( 4 , 0 ) ( 1 , 1 )
2  ( 3 , 1 ) ( 4 , 1 )
```

```
1  ( 2 , 0 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 1 ) ( 2 , 0 )
3  ( 4 , 1 ) ( 1 , 0 )
2  ( 3 , 1 ) ( 4 , 1 )
1  ( 2 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 1 , 0 )
3  ( 1 , 0 ) ( 4 , 0 )
2  ( 4 , 1 ) ( 3 , 1 )
1  ( 3 , 0 ) ( 2 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 3 , 1 )
3  ( 2 , 0 ) ( 1 , 1 )
2  ( 4 , 1 ) ( 4 , 1 )
1  ( 3 , 0 ) ( 2 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 3 , 1 )
3  ( 2 , 1 ) ( 1 , 0 )
2  ( 4 , 0 ) ( 4 , 0 )
1  ( 3 , 0 ) ( 2 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 0 )
3  ( 1 , 0 ) ( 2 , 0 )
2  ( 4 , 0 ) ( 4 , 0 )
1  ( 3 , 0 ) ( 1 , 1 )
```

```
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 1 , 0 ) ( 2 , 0 )
3  ( 2 , 0 ) ( 1 , 0 )
2  ( 4 , 1 ) ( 4 , 0 )
1  ( 3 , 1 ) ( 3 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 1 )
3  ( 1 , 0 ) ( 2 , 0 )
2  ( 4 , 1 ) ( 4 , 0 )
1  ( 3 , 1 ) ( 1 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 1 , 1 )
3  ( 1 , 1 ) ( 4 , 1 )
2  ( 4 , 1 ) ( 3 , 0 )
1  ( 2 , 1 ) ( 2 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 2 , 0 )
3  ( 2 , 0 ) ( 1 , 0 )
2  ( 1 , 0 ) ( 4 , 1 )
1  ( 4 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 3 , 1 ) ( 2 , 1 )
3  ( 2 , 1 ) ( 1 , 1 )
2  ( 1 , 0 ) ( 4 , 1 )
1  ( 4 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
```

```
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 3 , 0 )
3  ( 1 , 0 ) ( 2 , 0 )
2  ( 4 , 0 ) ( 4 , 1 )
1  ( 3 , 1 ) ( 1 , 0 )
davidvajda@localhost ~ % python3 graphautomat08272026C.py
# (c) david vajda
# 08/27/2026
# python3, zustaende, vier zustaende
# \dots graph: schaltwerk, einfach
4  ( 2 , 0 ) ( 1 , 1 )
3  ( 4 , 1 ) ( 2 , 0 )
2  ( 1 , 1 ) ( 4 , 1 )
1  ( 3 , 1 ) ( 3 , 1 )
davidvajda@localhost ~ %
```

Gut nächster Schritt am pc, der klaßische Automat, erst der Graph mit Python 3, dann . . . mit Latex/tikz selbst Mealy Moore + schaltnetz Quine schaltnetz Kodierung später Und Sonderausgabe Transistor: notwendig für Gatter auch schaltnetz das möglich in kaneda zum üben von Gatter auf Transistor Basis..

So, jetzt mache ich weiter also als Nächstes hab ich beschloßen und das Mikrogramm steuerwerk auf Papier zu machen ebenso das Operations Werk des Mikrogramm steuerwerk enthält ein Dekodierer der ist mit den Mitteln, die ich gerade eben gefunden habe, einfach zu vollziehen, ebenso ein Register und ebenso einen multiplexer Die 16 Leitungen am Dekodierer sind relativ einfach zu machen mit der Leitung, die ich getan habe, ebenso die Kreuzungspunkt te durch diese Sache erfordert mehr übung Ich mache nach wie vor die Sachen auf dem Papier, was wiederum keine übung erfordert ist Relativ einfach das ASM Diagramm Das werde ich tun Relativ einfach sind die normalen Automaten mit vier Zuständen wenn es um mealy geht und etwas komplett. Ich weiß nicht, ob kompliziert, aber mehr Zustände, wenn es um Moore geht. Vielleicht läßt sich das weniger gut verteilen Und wie man sieht am Ende doch die Gatter laßen sich ganz gut verteilen Wenn man das hat, kann man das übergangßchalt Netz und das Ausgangsaußchaltnetz nicht so einen relativ einfachen Automaten machen Wie gerade eben beschrieben ein Automaten mit vier Zuständen Damit der Graf alle Bedingungen erfüllt, nämlich das wenigstens ein Kante mit einer eingehende Kante das ist eingerichteter Graf in jedem Knoten führt, kann man eine Art Stapel beziehungsweise eine Art Tabelle nehmen, ein Stapel wo rausgenommen wird. Das hab ich ja vorher gesagt und es mit einem Python drei Programm machen. Das ist das Thema und hier kann man allerdings mit diesen Mitteln. Ich habe hervor gezeigt, oder wir haben es vorher erfahren, wie gehen die und Gatter und wir gehen die Oder Gatter die nicht Gatter sind auch nicht schwierig. Das

kann man machen. Die einzige Schwierigkeit sind immer noch die senkrechten Leitung mit sozusagen den Punkten an den entsprechenden Stellen. Trotzdem, das kann man dann immer noch als eine übung sehen und wiederum Sozusagen ist trotzdem auf dem Papier machen Es hat jetzt etwas länger gedauert, weil ich noch Wäsche waschen mußte und mich daran gewöhnen mußte, daß ich jetzt diese neuen Zeichen habe. Ich werde jetzt anfangen es wäre gut wenn ich mich noch umziehe und sauber machen und man muß nicht unter Druck sein, obwohl man dieses ich möchte dazu nichts näher erläutern. Sie haben das verstanden. Danke schön.

Gut, dann fangen wir jetzt mal an meine Idee für das ASM Diagramm gestern mit RTL Notation. Ich mache daraus nachher auch einen Beitrag ist daß ich das vorher mit kaneda zeichne. Das ist überhaupt eine Idee

Ich nehme ja IEEE-1984/91 Schaltpläne. Es lohnt sich sowieso, das zu üben, bevor man Latex nimmt. Mein Problem war ja auch das Papier Trotzdem wollte ich es ja gleich mit tikz üben. Damit es auch in Latex schnell geht.

Trotzdem ist das mit kaneda eine Idee? Ich beschreibe die Gedanken danach, aber ich schreibe es nicht auf Papier, es folgt, ebenso das einfache Schaltwerk, auch mit der Umwandlung von mealy nach Moore. Für den Algorithmus, daß der Graf ordentlich ist, hab ich eine Idee erdacht, die in einer Art Stapelbetrieb funktioniert in dem zumindest auf der einen Seite die Kanten heraus genommen werden. D.h. die Nachbarn, damit es nicht zu der Situation kommt, daß ein Knoten ein Anfang Knoten ist und ein anderer auch

Das mache ich natürlich ebenso mit Latex und die haben wir's wieder mit Zustandsdiagramm zu tun und eben aber auch und das auch bei dem Operations Werk hier voll Register an bei dem komplexen ebenso wie beim steuerwerk, fallen auch noch ein multiplexer, ein Register und ein Dekodierer an

So daß ich das übliche, Quine so mache ich an und erkläre das Problem nachher ALU 74181 habe ich zehn Stück bei mükra Elektronik spott billig gekauft das hab ich damals im Laden auch. Ich dachte die Dinger sind aufgekauft verkauft für eins 90 geht es noch. Ich fange jetzt an lange Rede kurzer Sinn auch noch auswendig lernen tue ich Arm Cortex und Minix wie immer das übliche Halt Code gehören auch dazu und halt das übliche und auch auswendig lernen so

Zwischen Notizen

Also erste Notiz was notiere ich mir?

Ich notiere hier jetzt also Prognosen für meine Prozeßor

Leider hat sich herausgestellt, daß wohl das Vertrauen in solche Sachen fehlt. Kurz und gut das muß ich jetzt nachher vom Prozeßor her erklären, was hier an der Sache fehlt.

Hinweis auf, sozusagen des Pipeline von Intel und so weiter

Daß wir sozusagen also all das was dazugehört muß, hier Rein,prognosen

Danke, kommt noch dazu, daß wir müßen hier. Ich mache jetzt Bemerkungen, was mir aufgefallen ist.

Das ist leider ein Problem

Beim einkaufen sozusagen also spielt eine Musik. Ich vermute, daß manche das fällt mir jetzt einfach nebenbei auf das ist Teil des Arbeitsprozeß. Ich notiere mir hier keine Gesellschafts Notizen und so weiter

Es ist seltsam, wenn jetzt hier irgendein Richter meinte, sich hier um die Sachen angucken zu müßen. Das ist nicht optimal. Ich muß mir das jetzt merken. Also wir sind in folgender Situation das ist mir nebenbei aufgefallen.

Irgendwelche Doktoren der Medizin schreiben wir wahrscheinlich mal wieder fleißig. Irgendwelche Prognosen dabei sitzen Sie vor sich und gucken auf Papier und schreiben irgendwelche Sachen wobei, der Kopf ist wohl mal wieder woanders gelandet. Wozu brauchen wir das weiß ich nicht das ist keine optimale Sache.

Kein Vertrauen in den Prozeßor von außerhalb, aber keine Kenntniße. Das wird so irgendwie halt sowas abgetan. Wißen Sie, wie man als junger Kerl vorm Papier sitzen, schaut er die ganze Zeit drauf und grins vor sich hin und nimmt so seine Tinte in der Hand mit und schreibt da irgendwas sowas vor sich hin das macht keinen Sinn.

Mir fällt derzeit die akut gefährliche Situation Situationen auf. Man spricht von einer Ausbruch einer Kriegsgefahr. Ich sag es einfach mal so jetzt notiere ich mir das ist alles Teil vom Arbeitsprozeß, was mir nebenbei aufgefallen ist.

Wenn jetzt hier des Bundesverfaßungsgericht meint, irgendwelche d.h. halt in Deutschland so deswegen kann ich den Namen gut nehmen und wer weiß wahrscheinlich sehr viel moralisch mitgehört. Ich sehe die Situation nicht so optimal an.

Sich hier um welche, sagen wir mal im im Laden ist mir aufgefallen. Es kam

eine Werbung. Haben Sie für ihren Einkaufswagen einen Führerschein? Unter diesen Umständen müßen wir drauf aufpaßen. Es gibt viele Gesetze rund um den Verkauf und ich würde dieses Thema jetzt nicht abtun.

Eine vermutete Prognose innerhalb der Sache ist fast das gesellschaftliche betrifft, daß man sozusagen als man macht. Dir sehr schnelle Schlüße. Es ist ja die 74 L S 181 , das wundert mich ehrlich gesagt.

Wie schnell man hier dahinter kommt, daß angeblich Insel aufgrund der Prozeßoren sehr schnelle Zange aufgrund der Transistor und aufgrund der Halbleiter sehr schnelle Prozeßoren baut. Es gibt andere sozusagen kann ich jetzt hier nicht erwähnen das kommt gleich in die Sache rein also

Also wie gesagt, ich hab Computersystem eins und zwei also das Problem bei der Zeit wenn ich jetzt lauter Transistoren hab, dann ist es das eine warum ich nicht weiß was in 74 LS 181 ist auf der einen Seite wenn ich auf der einen Seite nicht weiß, was das ist und auf der anderen Seite argumentiere ich so das sind ja irgendwelche Bauteile. Da muß ich eigentlich doch wißen was in 74 XXX ist muß ich eigentlich wißen sonst werde ich ja auch nicht argumentieren. Hier werden Sachen einfach durcheinandergebracht vermutlich von außen

Das zweite ist in dem Laden Spiel ständig eine Musik die ist ja nicht mehr auszuhalten. Mir ist aufgefallen, um sozusagen den Arbeitsprozeß einfacher zu machen und hier müßen wir noch uns Katastrophen kümmern. Es ist eindeutig davon auszugehen, daß es eine Kriegsgefahr gibt.

Mir ist aufgefallen, daß sie dabei keine eindeutigen Namen, was das Ausland betrifft, gibt trotzdem also was das Ausland betrifft, dürfen Sie eigentlich keinen Namen nennen von ausländischen sozusagen das ist auch nicht erwiesen

Die ganze Sache ist ungeheuerlich, weil mir ist aufgefallen. Es wurden digitale Frequenzen eingeführt, die Amplitude Modulation, die also, wo die sozusagen Langwelle Mittel ist, bleibt die, die sagen wir übertragung aus, das kann schon alleine ein kritisches Zeichen sein. Mir fällt die Verwechslung es gibt ein ständiges, sozusagen das Katastrophe sei nur.

Und es ändert nix weil ich sag den Leuten sie sollen sich konzentrieren. Schon vor dem Laden laufen sagen wir aber tausende könnte man sagen Leuten rum da kann man sich nur noch wundern.

Die werden so oder so in die Katastrophe kommen, wenn sie so weitermachen d.h. sollten die aus irgendeinem Grund bewußt an die Stelle gesetzt. Finanz übersehen ihre Situation. Die müßen in Deutschland nicht davon ausgehen, daß sie sich kein Buch schnappen können und der Widerstand in sich selber bleibt natürlich. An sich eine harte Sache des immer so deswegen können Sie da nicht drum rum diese ganzen Ar

Jetzt folgendes Beispiel der Obama wird immer wieder so dargestellt, wie nach vorne auf irgendeine Art Leinwand guckt also wenn man jemand von der Seite filmen zeigen weil da sitzt der Obama und der guckt nach vorne auf die Leinwand. Sagen wir mal ist es interessante Bundesverfaßungsgericht im Fernsehen zu sehen in Amerika und dabei guckt der Obama einigermaßen grinsen in die sozusagen auf die Verfaßungsgericht und macht mit dem mit dem Finger nach oben und das wird aber total unterschätzt. Dabei sieht man ihn sozusagen einigermaßen lachend genauso ein Foto viel mehr auf Facebook, auch von einer Frau die guckt lachen mehr oder weniger nach oben. Ich war nur diese Bilder können bewußt oder unbewußt verstellt worden sein.

Mir ist aufgefallen, daß es beim einkaufen ein Gesetz gibt, daß man eigentlich keine Musik spielen darf. Was heißt keine Musik die Musik an sich ist eine Katastrophe da singt sozusagen eine. Ich war nicht nur vor ausländischer Intrigo Intrigen. Sowas ist nicht zu unterschätzen, wenn beim einkaufen eine sagen wir relativ freundliche Stimmung und diese Musik ist ja eine Katastrophe das ist keine Hippie Musik das sind nicht die Hippies die sind nicht irgendwer die sind irgendwie das klingt wie eins sagen wir Gesang irgendeine eine sehr komische Gesangsmelodie muß ich sagen die würde ich mir schon per se nicht anhören. Die hat ja der Kollege auf dem Radio abgespielt. Ich hab mir sowieso schon etwas auf die Ohren gestoppt. Wir müßen hier aufpaßen darauf fiel mir ein ich steh an der Kaße und vor mir steht ein Mann das ist Teil des Arbeitsprozeß, das sind die also die um mich herumliegen den Umstände der steht da und holt stundenlang sein Geld raus und dabei muß ich diese Musik anhören.

Ich hab zum Glück ein Klavier und hab mich damit abgelenkt. Das ist auch notwendig und das Problem bei der Sache ist gleichzeitig diese vielen Menschen die rumlaufen diese Leute selbst wenn sie bewußt dahingestellt sind, müßen sich trotzdem ein normale geistige Tätigkeit suchen. Ab dem gewißen Alter arbeitet man in einem gewißen Alter nicht physisch und es ist sowieso fragwürdig, wenn sie, wenn diese Leute sich nicht eine geistige bewußte zu sagen Inhalts Vole Sache rein in den Kopf also lernen, dann kann es sein, daß die unbewußt dazu kommt. noch der Alkohol, das weiß ich nicht. Das kann sich seltsam auswirken in so einer Situation und dann kommt aber noch dazu.

Abgesehen von dem Alkohol, wenn die Leute das einfach so machen und tun weiter nix dann können Sie so oder so dieses diese Bilder das hat man doch in der Vergangenheit gesagt. Es gibt Schattenspiele und so weiter das wurde bereits in China erfunden. Meine Mutter hatte da auch so Schattenfiguren und Beispiele gibt es aber auch bewußt Kino. Es gibt für alle, was diese Sachen betrifft, gibt es psychologische Beeinflußung, die man unter Umständen merkt. Es hat meine Mutter auch immer gesagt denn zum Beispiel der Mafia in Italien da wurden immer so heimlich so Sachen an der Oper abgespielt, sowas kann paßieren und man merkt es nicht und dazu gehört zum Beispiel dieses Bild der Obama, wo der sozusagen oder weniger grinsen nach vorne guckt ob das jetzt bewußt ist oder unbewußt ob der vielleicht machtlos ist. Das können wir nicht erkennen. Das ist aber gefährlich, weil die Leute die sozusagen vor irgendeiner Arbeit und es ist immer immer wieder die selbe Katastrophe. Darauf hat mein Mathelehrer auch hin hingewiesen und es ist nicht irgendein Thema, wenn da jetzt so ein Doktor sitzt und er schreibt vor sich stundenlang Papier und er weiß nicht mehr was er schreibt und dann macht er hier ne Randbemerkung montiere ne Randnotizen dann geht er mit der Hand irgendwie so an den Kopf und guckt vor sich hin und ein schmeiß das Papier auf dem Tisch und sagte am Ende noch meine ich ja die ganze Zeit aber er geht einfach nicht von allein aus seinem Job raus, sondern man muß sich also das ist Katastrophe was da aufgeschrieben hat. Da werden Sachen aufgeschrieben da muß man also wirklich da muß man nur noch schreiben lesen und schreiben können. Das kann ich auch. Ich kann dich auch da hinsetzen und stundenlang irgendwas aufschreiben und wenn mir was auffällt und dann sitzen die da scheinbar hat. Der Mathelehrer hat immer wieder drauf hingewiesen. In solchen Situationen sind Leute oft nicht in der Lage gewesen gewesen, ihren Beruf zu verlaßen. Ich mein, was will ich da in so einem Scheiß das muß ich doch nicht machen. Also es gibt doch wirklich größere Probleme. Also jetzt hier steh ich auf dem Papier Scheiß zu schreiben da die ganze Zeit schreiben schreiben schreiben irgendwas was angeblich paßt macht doch keinen Sinn das ist immer wieder dieselbe Problem. Sei mal es gab Leute die wußten wenn sie gehen und sie machen's und ich sag trotzdem es gibt genug andere Leute die nehmen die Situation also akut an unangenehm war und das Problem ist, daß genau die nichts lernen. Sie können sich trotzdem Buch schnappen, dann sind sie nicht so komisch wie der Doktor, der damit böse ist, wenn da jemand sitzt und er schreibt und schreibt und schreibt sein ich sag noch mal und dann müßen die andern sich trotzdem versuchen. Irgendein egal was regulären sagen wir Inhalt suchen und den auch versuchen nicht einfach so lesen, sondern richtig verstehen lernen und richtig auswendig lernen so oder so, sonst führt es zu Problemen.

Und es sind so, die umstehende, die mich bewegen und wenn man im Laden reingeht, wenn man in der antike irgendwo ich glaub, das ist seitdem schon verboten da steht irgendeiner und verkauft und daneben stellt sich eine unbewußt daneben. Die schlimme ist, daß solche Situation scheinbar nicht erkannt, weil es sich scheinbar um moderne Sachen handelt dann könnte man jetzt wieder-

um behaupten, aufgrund dieser Situation ist es eine bewußt gemachte Sache ist es eine bewußt gemachte Sache und die Sache wird in Ordnung sein und das stimmt leider überhaupt nicht. Wir haben gehört die Situation ist nicht so gut, wie sie außieht und dann können Leute da im Laden, wenn sich da auf dem antiken Markt, die man hinstellt und daneben Trailer hat einer ein Lied dann beeinflußt ist unser Verhalten sehr schlecht und er ist, wenn es eine gezielt gemachte Sache ist wegen dem Einkaufswagen müßen Sie die Sache andersrum stellen. Ist der Einkaufswagen vielleicht schon die Frage mit dem sagen wir mit dem Führerschein vielleicht schon eine seltsamer Frage und darüber müßen Sie und wenn es aber so geht, dann sag ich, das ist der Zeit wo die Situation so ist und man kann nicht einfach einfach beliebige ausländische Namen nennen. Man kann nicht sagen der schuldige und so weiter und sonst was. Leider kannst du mit den Schatten spielen wenn da ein Mensch so grinsen nach vorne abgespielt wird, kann das auch unabhängig von der Situation statt schlimme ist das leuchtet zu bemerken scheinen und sich dann dieser Situation hingeben und es ist katastrophale, ob sie das bemerken oder nicht dann müßen Sie trotzdem in der Tätigkeit bleiben und ich habe gesagt, ich erteile Befehle, damit die Sache, so was sie eigentlich nicht machen darf. Ich hab kein Oberkommando ordentlich erhalten. Toll in diesen Leuten muß ich dir sagen, weil die Katastrophe kommt und die zweite, was mir an Umständen aufgefallen ist dann müßen Sie trotzdem nicht sich das angucken und somit mal man weiß nicht was in einer Situation paßiert das weiß man nicht und dann wird sie trotzdem ausgenutzt

Das zweite was mir aufgefallen ist, ist es mit dem Prozeßor. Man kann an dieser Stelle nicht sagen, weil das einfach nur sagen. Der altmodische Baustelle Bausteine sind diese Speicherbausteine die haben wir ja n Fenster erstens mal ist das sagen wir ein Konzept und das Konzept wird falsch eingestuft. Mir wird immer wieder klargemacht, daß wenn ich zum Beispiel sage. Jetzt mache ich also ich hab das Konzept es ist ja jetzt eine Beschreibung des Arbeitßachen also die Umstände sind sehr wieder muß ich sagen und man kann einfach keinen Namen nennen selber aber man muß schon drauf achten. Solche Sachen können immer ernst werden und sind es auch und sind es und ich sage noch mal Musik die vor sich hin geht. Beim einkaufen ist eine Reinigung Potthast

Es ist eine reine Katastrophe, und man weiß nicht, was mit diesen anderen Frequenzen paßiert ist. Diese Ahnung hab ich nicht und man kann dieses Signal ich hab da auch ein Rauschen. Gehört es an sich klang verdächtig auf einigen, als ob das eins zerlegt ist. Signal ist auf der einen Seite und dann kommt noch dazu man weiß nicht wann gehen ich ich kenne mich da nicht aus aber es gab glaub ich mal Außagen wie die Radiosender gehen dann wieder online. Also also die alten Radiosender auf den auf diesen Frequenzen, wenn die Situation so ist, wenn die Situation so ist, daß die Situation so ist und es kann möglich sein das wird unterschätzt. man redet früher, das wird genauso gemacht jetzt ist

Jetzt Einschätzung des sozialen Umfeld ist, wenn ich solche Speicherbausteine sind nehme und die sind mit dem Fenster und die sind alte Programmierbarre Speicherbausteine kann ich nicht aufgrund dieser Außage machen das ist irgendein Ding was nicht tut und ich kann nicht sagen aufgrund deßen muß ich eine Ahnung haben und die wirkt relativ sagen wir mal simpel ist es aber letztenendes nicht der Kurs campest eins und zwei ist nicht irgend so ne Sache und ich müßte quasi sehr lange drüber nachdenken, bis ich diese Sache habe und es ist Teil der Einschätzung, daß um mich herum, ob das ältere Bausteine sind, ob das acht beziehen und so weiter. Ich kann hier nicht einfach sagen, daß diese sagen wir Sache eine langsame Sache ist. Ich kann auch folgendes nicht machen.

Kann nicht sagen, daß nicht versucht hat diese Situation oder das Intel nicht versucht diese Sache zu machen. Diese Sache ermöglicht alles bis Pipeline, wenn ich die so löse, daß ich an den Adreße Eingängen sozusagen noch mal von den sozusagen Rambausteinen kann ich sozusagen noch mal die festen Spalte machen und es ist Teil des Mikrogramm steuerwerk. Da kann man sich schon vorstellen wird man gefragt aber das sind ja viele gespeichert werden parallel programmiert. Es ist ein steuern also eine eine Steuer ein Steuer Voort ist diktiert immer falsch das regt mich auch immer auf. Der macht immer vor Ort oder sowas okay dann hab ich das und wenn die Sachen soweit ist, dann muß ich trotzdem wißen. Das ist ja die CPU. Wenn ich eine andere CPU habe wo das Speicher sagen wir hier kann man ja größere andere Größen erreichen. Heißt das nicht ich kann den Speicher hier reinmachen und in dem Speicher landen die Daten, wenn ich das es ist jetzt Aufgabe und es ist halt der Unterschied. Ich muß an die Grenzen gehen mit dem, was ich programmiere, ist allerdings der Festwert Speicher7 programmiert. Ich könnte die Sache jetzt aber so machen, daß dieser Festwert Speicher7 in der Sache zum Reinmikrogramm Steuerwert wird. Trotz des Idee, daß der Datenspeicher an Bord ist. Das einzige, was ich jetzt noch habe, das hat Insel wahrscheinlich so nicht gemacht. Es gibt trotzdem keine Garantie, daß die Sache nicht mit Mikrogramm steuerwerk und ALO alleine funktioniert und es ist ein reiner Quatsch. Hier können ja auch spekulieren das ist trotzdem reiner Quatsch. Ich brauche nur diesen Festwert Speicher7. Ich brauche nur diesen diese IU und ich brauche nur zum Beispiel hab ich auf dem normalen Prozeßor noch ein Cash und dann hab ich den Registers Satz jetzt hab ich einen großen Registers Satz und das stimmt auf jeden Fall nicht, daß er Intel seine Probleme so nicht irgendwo löst, weil aus vor im Grunde wir haben zum Beispiel den Wechsel der der Register also der nicht den Programmwechsel mache, heißt es in Wirklichkeit im Prozeßor. Aber da ist das Fenster sind die gehen über das ganze Ding drüber und so weiter da kommt noch viel dazu. Es stimmt nicht, daß ich hier kein Pipeline machen kann. Das ist das eine Problem. Ich kann hier das Pipeline machen weil eine Pipeline bedeutet Zwischenspeicher die kann ich also bei fünf Stufen und die Inte geht bis 64 kann ich hintereinander sind es einzelnen Stufen die haben Zwischenspeicher in diesen Speichern. So wie ich die eingebaut hab kann ich auch die Daten für sozusagen. Bei dem Admin acht kann ich die

Daten einbringen. Die sind sagen wir die also des UBRR und so weiter also die gesamten sozusagen Daten für die für die Serie Schnittstelle und alles alle Ports und alle iO Register kann ich alles darein machen das ist das Problem und ich kann trotzdem Pleining machen und das ist halt der Unterschied, weil wenn es im Zyklus geht dann dann werden die hinteren Daten zwar nicht verarbeitet wie auch immer diese Maschine kann dann sie verarbeiten nicht parallel und trotzdem also mit dieser mit dieser. Also die verarbeiten das nicht parallel Problem haben sie zwar, daß sie durch eine einzelne Stufe leiten und trotzdem nicht parallel verarbeiten. Sie haben aber eine einzige Pipeline geschaffen also sie müßen unterscheiden zwischen der einzelnen Pipeline und parallel arbeiten den Pipelines. Können Sie eigentlich trotzdem machen muß man jetzt über nachdenken, weil sie haben trotzdem keine Parallelpipeline aber Sie können die Sachen auf dem großen Maßstab auch nebeneinander bringen, wenn sie in einer einzelnen Pipeline sozusagen haben sie ja die einzelnen sie nimmt die Speicher und das sind die Register und die tun sie durch die ALU nur sie braucht nur die Speicher DLU und den Festplattenspeicher in den Festwert. Speicher7 wiederum ist das Mikrogramm und das ist ein Hardware Algorithmus ist die Schaltung aufgebaut müßen Sie die Sache nur noch entsprechend der Grenzen, die sie hier auf diesem Speicher auf dieser Platine haben umbauen das Problem ist diese ständige Ungläubigkeit dieses Mißtrauen auch ständig in sich selber das ist eine Katastrophe. Sie müßen auch das Pipe rein. Können Sie damit mehr stufig machen, indem sie diese Sache parallel halt anordnen und trotzdem können Sie einen großen Maßstab natürlich wieder anders machen. Die Pipelineprinzip bleibt, weil eine Einstufige Pipeline die haben halt keine mehrstufige hat ja trotzdem nein haben sie auch nicht. Sie haben einen einzelnen haben sie nicht. Ist auch egal trotzdem oder doch weiß nicht, kann ich grad nicht beantworten. Trotzdem haben Sie diese diesen diesen Zyklus und den können Sie als Mikrogramm trotzdem benutzen. Also die Sache wird total unterschätzt und es stimmt alles trotzdem nicht, weil ich hab hier auch keine parallelen Festwert Speicher7 das stimmt einfach nicht. Ich hab hier keine parallel Festwert Speicher die einzelnen Programme haben und das eigentliche Programm kann ich immer noch von von woanders her beziehen. Das ist das Problem wenn das hier das feste Programm ist und ich habe hier eine Harvard Architektur sag ich mal da muß ich trotzdem drauf achten kann ich das Ende des feste Programm immer noch von woanders her beziehen und da muß man einfach sehen ist es ein davon gehe ich aus und wenn die Sache programmiert ist, hab ich diese hab ich gesamte Mikrogramm die Hardware hier vorliegen und muß entsprechend der muß entsprechend der umstehende, die ich hier habe, muß ich entsprechend den Hardware Algorithmus unterbringen und er ist Teil in diesen festen Speicher und der Unterschied bei der Sache ist irgendwo. Wenn ich diesen festen Speicher hier habe, muß man einfach akzeptieren. Dann ist das ist einerseits unser dekodieren. Das ist halt unser Deut i.R. dann muß ich mir jetzt darüber keine Gedanken machen, wo die Sache ist, sondern entsprechend der Umstände und der dekodieren ist immer sagen wir ein solches Wort und da gibt es nicht viel mehr als ALU das muß man wißen und dieses Wort, was eigentlich ein Scheißnetz ist, paßt eben in ein Speicher und ist programmiert. Trotzdem sind speichern im Gegensatz

zu einem anderen nicht ein Durcheinander und wir wißen da Speicher Matrix Matrizen sehr einfach auf ein Baustein unterzubringen sind. Im Gegensatz zu anderen Sachen sind die sehr einfach auf einen Baustellen unterzubringen und paßen viele drauf. Besonders bei kurzen Zeiten. Das ist das eine Problem so und das kann ich dir genauso machen und das andere ist der harte Rhythmus gehört in die Dinger rein dann bleibt mir nicht viel mehr viel als ein Hardware ein harter, fester Speicher und diese diese Speicher7. Die kann ich nachher immer noch anders benutzen oder ich kann sie auch drauf laßen und die ALO vielmehr ich hier nicht zu tun. Das müßen wißen solche Sachen der hat vollkommen falsch eingestellt und ich sag mal das Umfeld muß ich trotzdem reduzieren. Es ist nicht natürlich, daß ich das nicht veröffentliche. Ich bin trotzdem Teil von Facebook, auch wenn ich das hier mache. Und die Leute bringen sich in Gefahr mit den Umständen auch, daß ich diese Sachen hier falsch einstufen. Sie können nicht sagen, das sind alte Baustellen. Natürlich müßen Sie sehr gute also sagen wir maximale Ausnutzung müßen Sie diese Sachen extrem anpaßen sag ich mal alle mich extrem anpaßen, müßen Sie natürlich die Sache überdenken, daß dies noch zu ergänzen ist. Trotzdem sag ich aufgrund dieses Konzept wie gesagt und das ist jetzt mein Konzept, daß ich eben diese Hardware Algorithmus habe der ist auf den Speicher baustein, der ist nie woanders bei großen Prozeßoren macht ist es dann einfach, daß ich sozusagen trotzdem diese quadratische Struktur habe. Das ist das eine Problem und dann bleibt es trotzdem bei dieser Sache und ansonsten nur die AO und ich kann trotzdem noch den Datenspeicher hier vielleicht unterbringen und selbst wenn ich das bei dem Prozeßoren ich, dann mache ich das woanders. Das Prinzip der Pipeline kann ich mit irgendwelchen Mitteln, wenn du jetzt auch so nicht direkt 100 pro des Umsatz, weil es in der hinteren Pipelinestufe nicht gleichzeitig mitarbeitet das ist ja der Unterschied es müßte mitarbeiten könnte ich trotzdem umsetzen. Ich kann diese Sache parallel nehmen. Ich kann die Sache wieder parallel nehmen dann hab ich sozusagen eine Pipeline und dieses Pipeline kann ich wieder wie auch immer so kann man die Sache aber intern lösen es garantiert nicht, daß es nicht so gelöst ist und trotzdem was in den Hardware Algorithmus reinkommt, d.h. in den festen Speicher ist Teil der Programmieraufgabe und damit muß ich mich nur um die festen Umstände kümmern. Es ist allerdings gefährlich, wenn Leute anfangen, solche Sachen auf den leichten Schulter zu nehmen. Wenn sie solche Sachen sehen, dürfen sie dem nicht mehr trauen. Mit solchen Mitteln läßt sich selbst wenn das hier kein Intel Prozeßor ist, läßt sich hier ein Ergebnis erzielen wird sich mal prophezeie, das ist eine natürlich irgendwie prophezeit und trotzdem sehe ich, was ich damit jetzt erreichen kann. Im Gegensatz zu anderen Maßnahmen habe ich einen Prozeßor. Weil ich eben zum Beispiel den Datenspeicher nicht nur den Registers Satz habe, sondern hier den Datenspeicher habe, kann ich einerseits jetzt die Register nehmen und die Register sind mit dem Datenspeicher verbunden und er habe zwei Datenspeicher. Damit kann ich den Steck da kann ich mir komplizierte Sachen ersparen. Die sind in dem festen Speicher in der Mikrogramm drin. Ich habe sozusagen die Trennung nicht so und trotzdem ist sie vorhanden aber dadurch mache ich meine Befehle einfach einfacher und damit kann ich ein extremes Tempo erreichen eine extreme nicht und nur Tem-

po. Die Leistungsfähigkeit von dem Ding ist extrem groß, weil ich hier ja den Datenspeicher habe und die allen die Einfach kalt macht es extrem leistungßtark ein extrem leistungßtarke Sache dafür, daß es diese Sache ist. das darf ich nicht unterschätzen wenn ich das auf großen Konzept hab ich hier ein extrem und es gibt bei der Frau Neumann Architektur nichts sag ich mal nichts und bei Harvard was mir verbieten, würde das eben das Mikrogramm also das ganze Programm was in diesem Testen Speichen untergebracht ist da gibt es keine Außage, wie das nicht so geschehen hat. Der Unterschied ist, daß die programmiert sind. Die können aber auch einmal fest im Prozeße untergebracht sind. Da bleiben Sie quadratisch und es ist das Problem damit hab ich einen extrem sozusagen Ausnutzung der Zellen und es gibt nichts was mir nicht garantiert Arbeitßpeicher, diese festen Speicher und die Recheneinheit. Damit hab ich schon alle Teile erledigt und es ist halt der Unterschied viel mehr als Recheneinheit und Speicher brauch ich nicht damit hab ich schon an sich Leistungsfähigkeit plus den Datenspeichern Leuten sage ich trotzdem auch wenn es hier so stattfinden das darf man nicht unterschätzen. Man muß dringend sein. Ich faße jetzt alle Papiere zusammen. Ich faße die alle zusammen.

Dieses Zeugnis sind teilweise Tippfehler drin aus ALU wurde Arlo und laut so Sachen. Das ist natürlich keine optimale Sache und es wurden Sachen falsch das hat er Diktier falsch übernommen. Ich muß das zusammenfaßen und ich mach das jetzt die Hardware automatisch nicht immer alles auf einmal und ich kann auch die Kabel schrittweise miteinander verbinden. D.h. ich muß das nicht in einem Tag machen. Ich hab jetzt hier die schwarzen und roten Kabel und die muß ich langsam verbinden d.h. ich mach jetzt so schwarze rote grüne weiße Kabel und so weiter ich stecke eins nach dem andern in die Sache rein und wenn die Hardware aufgebaut ist es auch nicht in einem Schritt geschehen dann ist es soweit erst mal gut erst mal und jetzt mache ich einen Beitrag aus einem inzwischen Zigarettenpause.

Eine Schwierigkeit bei der Sache ist wohl, daß ihr gewiße Leute irgendwen, die ganze Zeit nach Affen, also mit nach Affen und aufschreiben und menschlichen nachvollziehen und es wollen die notiert haben also ich soll sie darin gewiße Leute darin notieren das sag ich jetzt einfach mal so, daß sozusagen sie mein, was sie da falsch machen. Also das ist jetzt nicht von Intereße und sie sollen nie aufhören, Wirtschafts Verdacht gegen andere zu schüren. Ich sehe hier nur noch Leute die haben keine Ahnung die probieren sich aus der Situation raus zu bewegen, indem sie sozusagen eine Initiation bekommen, indem man in der so erklärt. Ich sag die Leute sollen sich jetzt endlich mal zusammenreißen und nicht den anderen mit MED drohen und den Leuten hier irgendwie die die Sache schwierig machen und irgendwie so rum sondern die sollen sich jetzt mal aktivieren. Ich brauch einfach keinen Doktor der hier probiert Oberhand zu gewinnen, obwohl er ein Idioten Job hat und es will ja nicht einsehen also es wurden weltweit schon Sachen überholt und er kann seinen eigenen Beruf nicht

einsehen und früher hätte auch jeder den Doktor davon gejagt, und er lebte in irgendeiner Form von Vergangenheit sei und dann kommt er, und der findet es interessant, daß ich ihn damit beschreibe. Also Beschreibung und Beschreibung von Menschen. Es gibt beßere Sachen, wenn sie irgendwelche erotischen Zeichnungen machen wollen. Nächstes Jahr das eine das haben sie wohl verwechselt da allein sehr Schwierigkeiten was die gesamte Sache beträgt. Politische Nachrichten sind nicht besonders optimal hab ich die Leuten gesagt das ist sowas des Umfeld betrifft nicht nur das Umfeld auch weitere. Ich verstehe nicht wie ein Doktor und auch nicht ein Verfaßungsrichter die ganze Zeit da sitzen kann. Es ist doch an sich Unsinn. Ich kann die Situation nicht lösen. Ich hab keine ernsthaften sagen wir Ansprüche darauf bis jetzt zu tun, weil es sind ja immer die gleichen, die hier ihre Situation lösen müßen und dann also dann warum bleibe ich dann bei der Sache dann hab ich dafür also dich einzig sehr seltsam und die wollen, daß ich ihn disk kommentiere. Ich verstehe nicht warum die da sitzen und sich das nicht einsehen können und warum sie nicht merken, daß sie da total ohne ohne möglich also das ist reiner Quatsch und wie so jemanden im Stift da sitzt und probiert die Leute zu kommentieren. Dann sag ich die sollen sich mal auf ne Arbeit konzentrieren und was sich bei den Leuten sehen sind die auch nicht in der Lage dezimalen weil die haben sich da scheinbar drauf eingeschworen und die müßen sich dringend ne geistige Arbeit suchen. Also wenn ich mal sagen Computer sind hart dann gibt's ja also ich verstehe nicht wie jemand die ganze Zeit dann wird man nach geimpft und so weiter und das macht die Arbeit nicht einfacher und dann versteh ich mal nicht was ein Doktor eigentlich in seinem Leben noch falsch machen wenn er sich dann um andere Sachen kümmert und dann sage ich müßen die Leute eben umso mehr sich auf konkrete Sachen kümmern. So vom Arbeiten fällt weiß ich Ihnen nicht gern quittiere. Warum erwarten diese Leute die so ein Scheiß machen? Ich sag mal Jörn irgendwo anders Hin. Warum erwarten die von mir daß ich sie dafür kommentiere? Also jetzt muß also damit ihr nicht alle so rumlaufen sollen die trinkt sich irgendwas im Kopf rein tun und nicht bei der Beschreibung ist ja schlimmer auf dem Fernsehen. Als Jugendlicher sagen wir ja jahrzehntelang vorm Fernseher und guckt in die Röhre rein und da paßiert irgendwas die scheinen nachzuvollziehen. Wie funktionieren Computer deswegen kann ich das ja nicht bleiben laßen die halten sich selbst für extrem wichtig. So Leute das ist überhaupt nicht wichtig und ich sag es außtechen Gründen, wenn sich diese Sache verwerten läßt und es läßt es auch. Da müßen die nicht davon ausgehen. Ja, die ließe sich nicht verwerten und reden da wieder mit Psychologie Maschen kann man sagen die sind ja nur noch Psych Psychologie Maschen im Gegenteil die Leute müßen sich ne Beschäftigung suchen. Kreth die Worte im Kopf reinmachen und so weiter also was in Büchern finden und nicht so es funktioniert nicht das muß ich denen aber nicht kommentieren. Die erwarten sozusagen, daß man sie in dem, was sie tun wahrnimmt, wie auch häufig in der Kunst und linken Zähne. Ich will sie aber nicht wahrnehmen und so ein Mensch ein Doktor zu schimpfen, das macht daher sind, weil der Beruf ungefähr es ist, so und man es nicht einsehen kann dann hat man doch irgendwo ne Dachlatte. Sagen wir mal machen doch genauso wie die linke Szene nicht im Theater, damit ich die

wahrnehme. Trotzdem ist die linke Szene in Gefahr und die können nichts dafür. Da muß man sich nicht mehr wundern, wenn's die gibt, wenn Leute nur noch in diesem Beruf sind, um Sachen zu schreiben. Ich sag ich kann auch nicht Sache schreiben. Ich kann auch tolle Zahlen irgendwo irgendwo hinschreiben, der mir einer Zukunft, dann tue ich so, daß da irgendwas steht, damit ich hier nicht aus der Faßung gerade als ob das nicht dem entspricht und dann können Sie der linken Sehne keine Schuld geben die haben genau eingesehen die Situation ist nicht optimal. Ich kann nicht sagen, der in anderen Ländern irgendwelche bösartigen Sachen macht. Es ist aus diplomatischen Gründen nicht legitim da sind sich alle einig und trotzdem ist die Situation schon an sich sehr seltsam. Wieso kann ein angeblich kleiner großer Doktor nicht verstehen und vor allen Dingen das neue Auszubildende ist in das die Situation zu verlaßen hat die selbe Problem, die mit dem Waßer Mathelehrer gesagt, als es gibt ein paar Leute, die schaffen das nicht sich aus der Situation warum soll ich ihn disk kommentieren ich muß sie dafür kommentieren und ihn Bericht schreiben nicht die Leute müßen was tun und sie haben das nicht in der anzweifeln, die haben in der ganzen Sache überhaupt nichts zu tun. Sie haben da nichts zu tun und sie müßen das nicht ausnutzen. Ich bin auch nicht so doof, daß ich und es ist niemand seine Schlüßel also sie müßen uns nicht die Schlüßel vor das selbe was im Fernsehen paßiert da sitzen Leute die erklären uns was sie genau schließen können. Also wir haben selber auch noch nicht. Wir haben auch noch selber ein Denk Apparat und soweit wird's noch reichen sonst müßen wir irgendwann ein stimmt was nicht und diese Sachen müßen Sie beachten, daß sie hier in richtige Sachen ich habe nicht vor das einzustellen. Ich habe nicht vor, das einzustellen, und das hat nicht infrage gestellt zu werden, weil angeblich irgendjemand darum der irgendwas, weil der sowieso notwendig ich sag die Leute müßen alle was normales tun und irgendwas konkretes und was ordentliches. Wenn sie das nicht machen, dann werden dann wird es schlimmer und es ist als Angriff zu bezeichnen und kein Kavaliersdelikt und keine Nachläßigkeit, sondern bewußt falsches Verhalten. Da kann sonst was in der stecken wie mit der Musik im Einkaufsladen wenn man das so macht. Die Leute müßen sich ne konkrete Sache suchen, ob sie jetzt mit der Situation insgesamt zufrieden sind oder nicht mal setzt sich nicht dahin und macht da irgendwelche machen. Schlußfolgerung kann jeder Mensch und sie müßen nicht drauf achten, was jetzt nicht paßt und wenn sie das nicht entscheiden können, dann müßen die auch nicht an der Sache weiter teilnehmen. Also die müßen das nicht jetzt hier einen auf sonst was alles machen das ist das eine Problem und dann sollen die das mal die Leute müßen sich hier selber und die Probleme haben sie von allein das ist, was das Arbeitsumfeld betrifft. Warum ist es beschreiben soll seh ich persönlich nicht ein die Leute können es nicht entscheiden. Die Leute müßen aber sich irgendwo hin bewegen. Warum die das nicht einsehen können, weiß ich nicht und solche Sachen wie das hier haben nicht zum mißachtet werden. Erstens Mal muß eine Tätigkeit nicht auf deren Grundlage die machen alles falsch übersehen, die die Situation nicht die sind ja schon das machen wir sowieso nicht weil sie müßen die Situation einschätzen das Gedudel im Laden das sagen wir verboten während dem während dem Verkauf eine Musik und es ist keine Hippie Musik kein Techno zu machen. Das

ist gemeingefährlich der überhaupt von der Verkauf her und der Patient hier können überall Spion im Radio sein. Die Doodle läuft da die müßen lernen sich diese Sache zurückzuziehen und die können dich da weiter zuzuhören und alles kommentieren. Die Leute können selber Schlußfolgerung machen. Die müßen sich zu Hause hinsetzen und was ordentliches tun und dann hab ich gesagt er lernt doch das hier mit der Armee dann ist es doch beßer und dann wird es wieder angegriffen. Es gibt Computerspiele Armee technisch ist nicht verboten sag ich, sonst wär es nicht im Handel und ein umso beßer das müßen Sie trotzdem verstehen lernt und daß er im Internet steht die Leute braune Beschäftigung und sei's dies ich hab auch auch ein Armee da kann man keine Kritik üben das ist das eine und das andere ist dieser Prozeßor trotzdem und sprengt. Alle Leistung von dem, was man sonst so macht weil er hat halt einen Unterschied. Er hat halt einen Unterschied er verbindet halt nur sozusagen die logisch und die Rechenlogikeinheit sozusagen sag ich ihr weil dieser dieser Diktier kann das Wort so schlecht aufnehmen. Die Rechenlogikeinheit die wird halt einfach mehr oder weniger nur mit dem Datenspeicher und mehr oder weniger nur mit dem Festwert Speicher7 verbunden und in dem Festwert speichert sie halt das Mikrogramm enthalten und das Mikrogramm ist der Hardware Algorithmus in der Realität. Was moderne Prozeßoren betrifft, habe ich dann sozusagen die je mehr etwas eine Matrixstruktur hat und so einfach ist es her zu stehen als wenn ich diese kleinen Bauteile alle nebeneinander haben müßte. So ist das Problem bei der Sache. Trotzdem kann ich das hier pragmatisch umsetzen und der Rest ist trotzdem hart das Mikrogramm weil das macht es zum Netzwerk erst, das macht es zum Netz das macht es zum genau dieser Speicher Baustelle nehme ich der Festwert Speicher mit seinen gespeicherten Daten ist das Netz dieser Festwert Speicher7 mit Mikrogramm und das Mikrogramm kommt Programm technisch rein ist das Schalt Netz. Trotzdem hab ich die Möglichkeit sozusagen die Sache anhand der sozusagen physischen Realität nämlich der Kabel und der Bausteine auf meiner Platine. Kann ich diese Sache so lösen, daß ich den Rest entsprechend der Grenzen, die durch die Kabel und die Verbindungen sozusagen gesetzt sind. Durch diese Grenzen kann ich den Rest egal wie mit diesem Steuer Algorithmus und es ist halt ein Hardware Algorithmus lösen. Also das ist ein Hardware Algorithmus das ist das Mikrogramm also der Mikrogramm und was ich hier auf dem Prozeßor vorfinde also auf der auf der realen an das sind sozusagen die Umstände und da muß ich sozusagen in der Lage sein dieses Mikrogramm entsprechend dem, was ich hier auf der Platine vorfinde anzupaßen. Trotzdem ist die Leistungsfähigkeit nicht genommen, weil allein die Datenspeicher sind hier so leistungsfähig. Das müßen wir wißen, daß ich diese diese Befehle zum Beispiel mit Laden und speichern und dem Registers Satz mit dem Stapel das ist ein mit der Sache. Das kann ich aber einfach dadurch lösen, daß ich das Mikrogramm jetzt auf dem Festwert Speicher7 unterbringe dadurch, daß ich das mache, löse ich das Programm mit dem Stapel entsprechend dem, was ich vorfinde auf dieser Platine, was aber einfach ist und trotzdem ist es das Mikrogramm und ich muß alles entsprechend dem, was ich hier machen kann und schon richtig verbunden ist damit lösen also zum Beispiel den Stapel und so weiter und trotzdem ist das ein harte Algorithmus und damit alles einfach zu machen und darüber muß

ich weitermachen das muß ich mal und sowieso ist es gegeben die Sachen nicht ruhen zu laßen. Man kann sich nicht hinsetzen und nichts tun. Man muß hier konkret weiter arbeiten und diese Sache ist eßenziell und fundamental. Ich wie gesagt was ich erklärt hab und ich kann das nicht 10 Millionen mal erklärt

Zum Beispiel die Leute denken sehr an sich das ist nur die Beschreibung des Arbeitsprozeß. Wenn ich sag wo ist der Mann von Tengelmann. Der Mann von Tengelmann ist abgehauen. Wann ist der abgehauen viel früher wahrscheinlich die sind keine Einzelfeldomen wenn wir jetzt an die Musik denken die im Rewe spielt, merkt der Ladenbesitzer. Ist es bewußt stecken hier ausländische Kräfte nicht, daß die Ausländer schlecht sind jetzt sind alle bedroht. Jetzt sind Leute angekommen die sind nicht so daß sie dann weiß ich auch nicht was es Gedudel sei. Also ich hab keine Ahnung oder? Spielt sich das aus Versehen ab wenn man den Obama von der Seite zeigt das sind Schattenspiele die sind gemein gefährlich und trotzdem paßieren also sag ich Kopfstütze auf dem Laden nicht zuhören das ist ja wohl und da muß man sich konkret zu Hause konkret, wenn man wiederum behauptet, daß diese Sache keinen Wert unter Umständen entstanden ist. Das stimmt nicht das stimmt einfach so nicht und wir müßen wie gesagt ich erklär noch mal die Sache wenn wir und dann kann ich andere ärgern. Ich kann hier nicht beliebig außagen. Solche Sachen sind schwierig zu erklären und man kann nicht einfach mal eben was sagen. Ich weiß, daß das, was hier paßiert ordentlich im Zusammenhang hat kann man trotzdem das andere hören ich kann den Leuten ich kann ich rede über diesen Prozeß und über den Arbeitsprozeß und nicht zu dem Prozeßor und nicht über irgendwelche Sachen, die außerhalb deßen stattfinden und das sind keine Sachen außerhalb verstehen Sie bin nicht mit was anderem beschäftigt das ist das eine Problem und das muß ich so und wenn man jetzt daran nachdenkt, dann ist diese Idee noch gravierend ich habe den Arbeitßpeicher, weil ich den auch doppelt nehme genauso wie dem Programm Speicher7, der allerdings hinterher wahrscheinlich eher das Mikrogramm enthält also den Hardware Algorithmus und der muß irgendwo rein. Ist die Sache trotzdem fundamental aus dem Grund, weil ich eben Den Rechenwerk habe des Rechen und die logische Einheit mit zwei Eingängen und dann nur noch zwei Datenspeicher und noch vier Programm Speicher7, wo an den Adreßen von den Datenspeichern wieder Programm Speicher7 sind, die sich aber nicht der Programm speichert. Das ist das Mikrogramm in meinem Beispiel und in meinem Beispiel. Da müßen trotzdem die Befehle in diesem Mikrogramm gelöst werden. Hab ich jetzt trotzdem die Einfachheit eben einerseits daß der Stapel einfach ist dadurch daß er im Mikrogramm gelöst wird und eben nicht der Stapel mit dem Stapel. Also ich mach nicht ich denke über die Register danach und habe nachher irgendwas anderes, sondern der Stapel und so weiter wird hier mit diesem harte Algorithmus in jedem Fall implementiert und wiederum die Programmierung des harte der Algorithmus und wie ich etwas auf diese Programm Speicher7 unterbringe, also auf diese festen Programm, Speicher7, Fotos des Mikrogramm ist. Da die Sache richtig verbunden ist, ist es jetzt sozusagen die unabhängige Aufgabe und zwar anhand was da ist, anhand deßen, was da

ist. Es sind diese zwei Arbeitßpeicher Speicher7 und Rechen kann ich die Probleme fast alle lösen, indem ich den Hardware Al Rhythmus deßen anspreche, wo die Grenzen sind, sagen wir, wir stellen uns die Straße vor und hier sind Häuser, hier sind Häuser und da ist im Weg und ich kann nicht beliebig laufen und trotzdem funktioniert alles. Es ist sozusagen geschaltet. Anhand deßen, was da ist, kann ich jetzt mit diesem harte Algorithmus in dem Festwert Speicher7 alles lösen aber ich kann es so lösen entsprechend dem, was ich sozusagen an Kabeln und so weiter, also an Straßen und so weiter vorfinde entsprechend der Umgebung kann ich hiermit alle Probleme lösen in dem HDI Rhythmus, die auf dem Programm stecken, so muß man sich das vorstellen und überhaupt nicht überflüßig, deswegen weil die Register so gut mit dem Datenspeicher verbunden und jetzt wieder umstellen. Ich muß jetzt meine Mutter anrufen und dann muß ich hier trotzdem gucken, daß ihr ein bißchen Ruhe und anders kann ich die Probleme nicht lösen. Anders kann ich das nicht machen

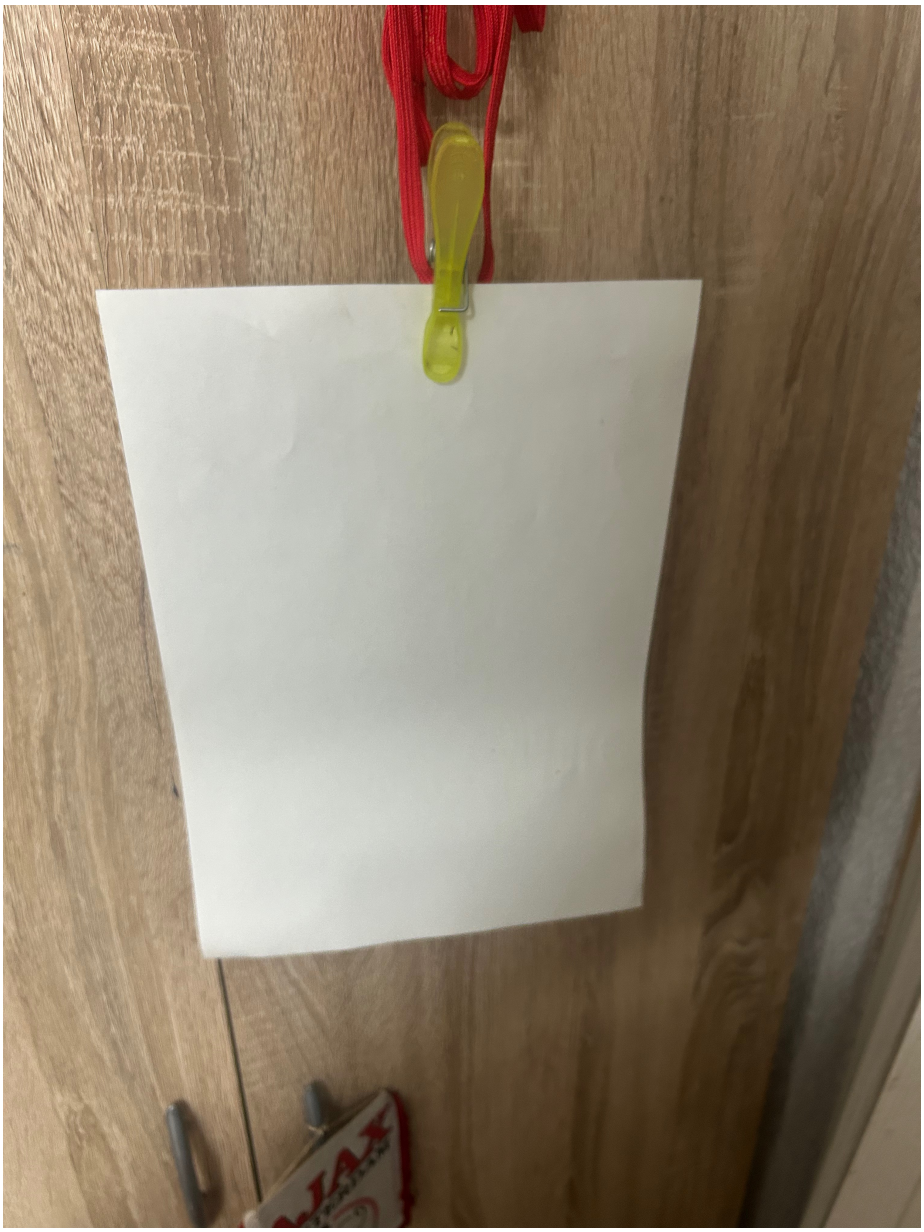
10.1 next

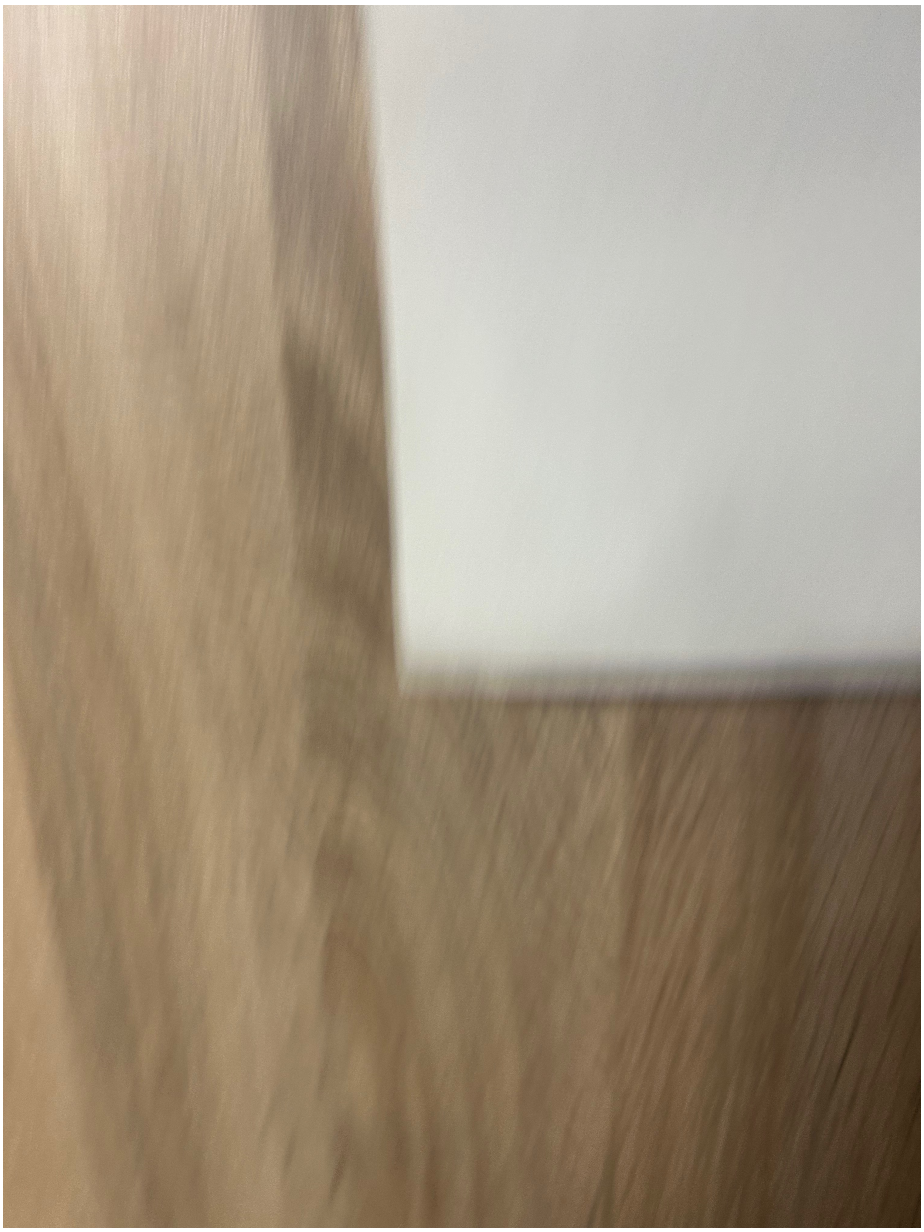
(D). Die Eingangsschaltung läßt sich zum besseren Verständnis auf ein Di-
odengatter zurückführen (→ Bild 8.9).
+U_B
R_1
R_2
R_4
T1
T2
T3
Eingänge
R_3
T4
D
Ausgang
Bild 8.8 Grundschaltung der
Standard-TTL
B
E_1
E_2
C
a)
b)
c)
Bild 8.9 UND-Entflechterung durch Multiemitter-Transistoren
a) bei TTL, b) Ersatzschaltung, c) Vergleich mit Diodengatter
Wirkungsweise:

edding 360
Whiteboard Marker
Whiteboardmarker · Marqueur pour tableaux blancs · Marcador para pizarras blancas
OPEN FOR 5 DAYS
REFILLABLE
dry wipeable
trocken abwischbar
droogafwisbaar
effaçable à sec
borrable en seco
1.5–3 mm
+ 40 %
edding
Made in Germany

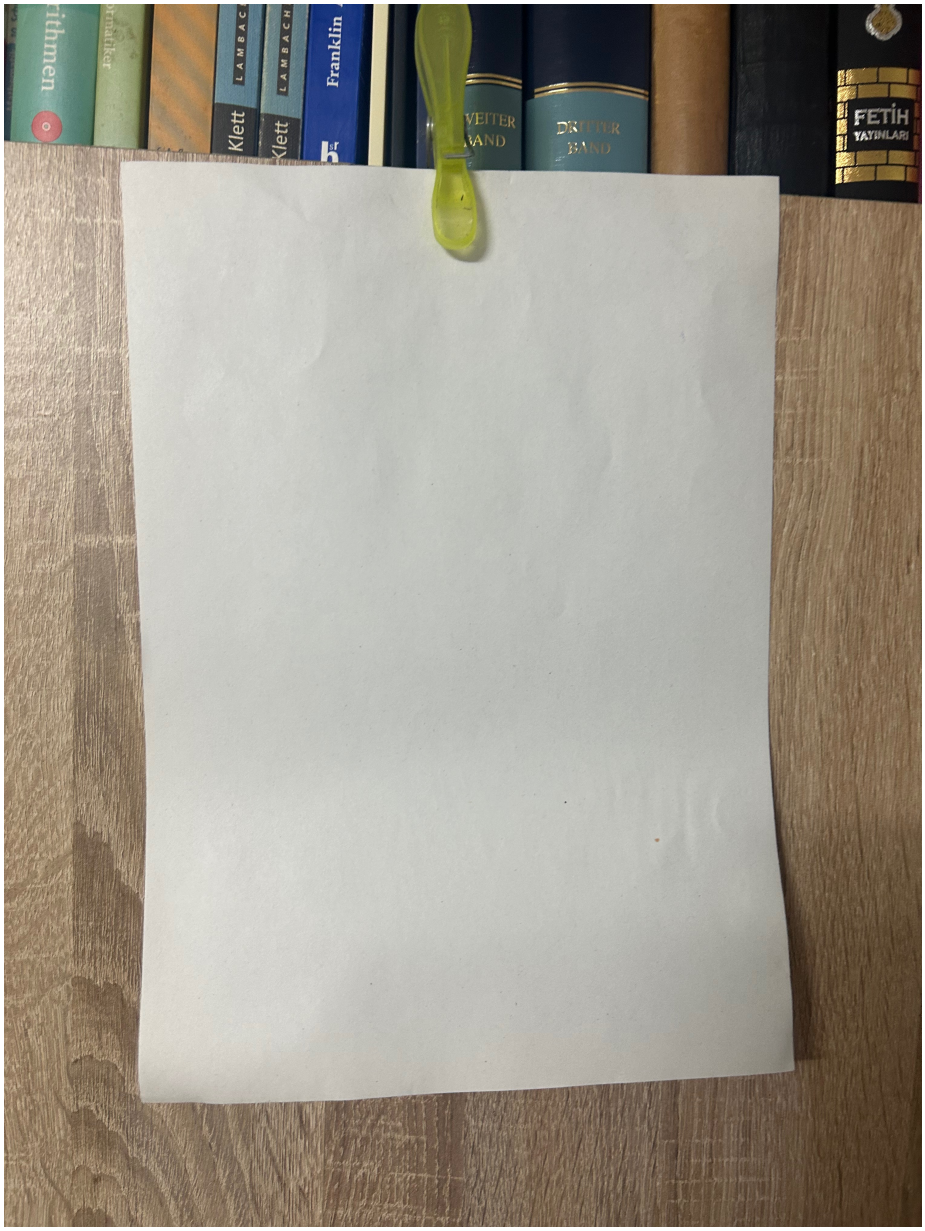


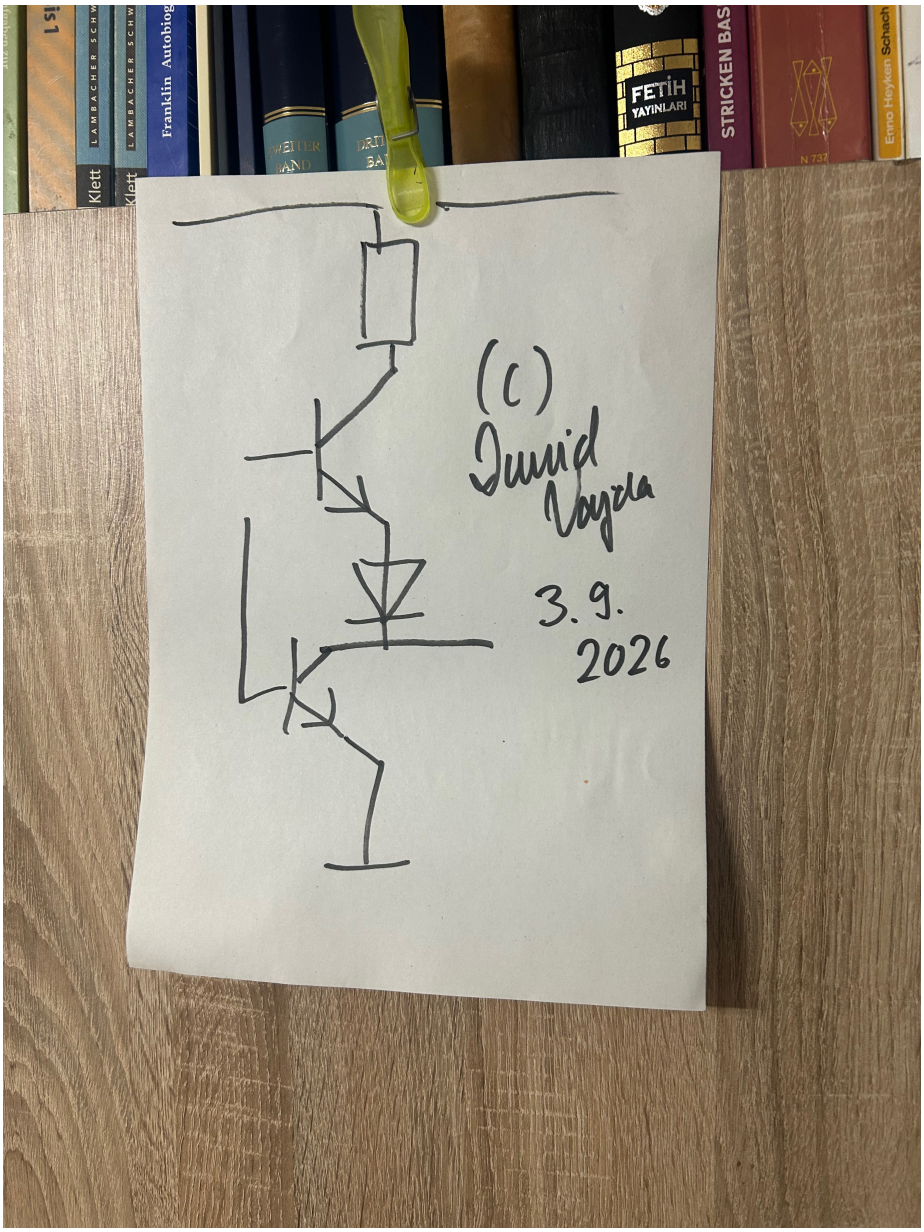


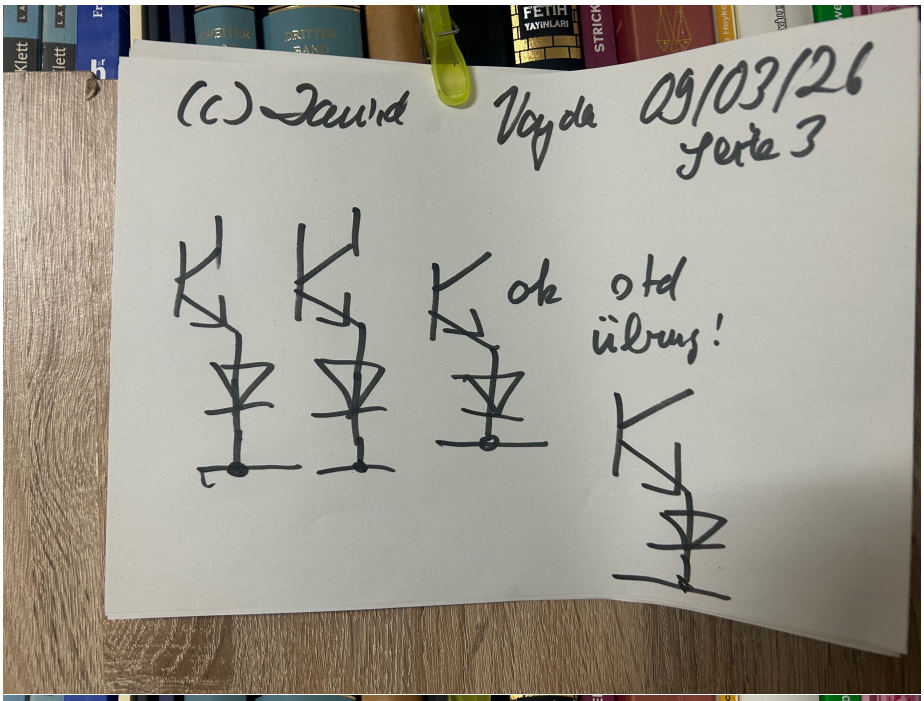


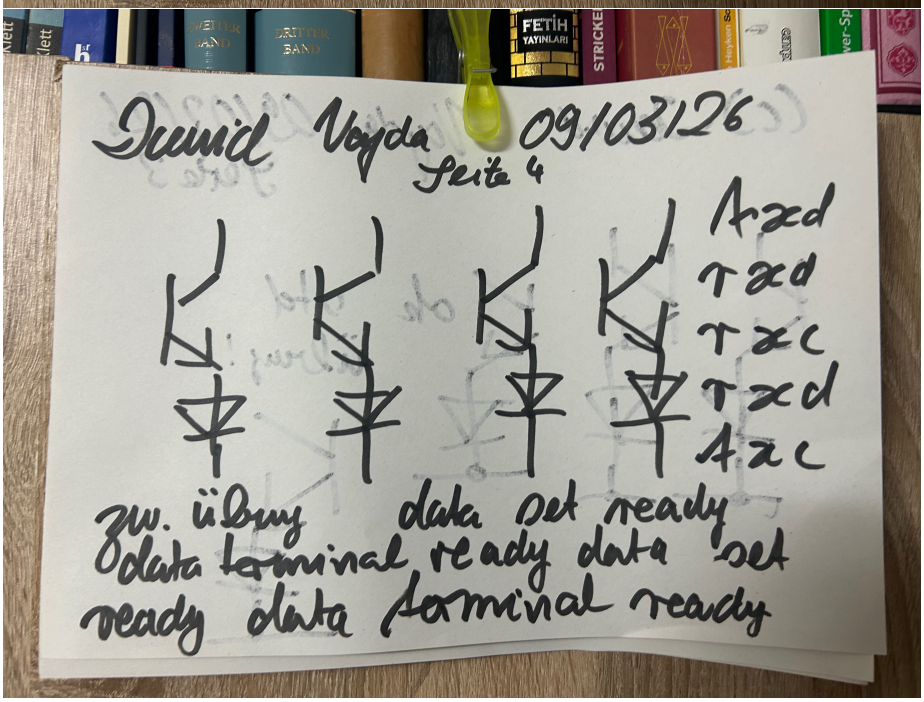


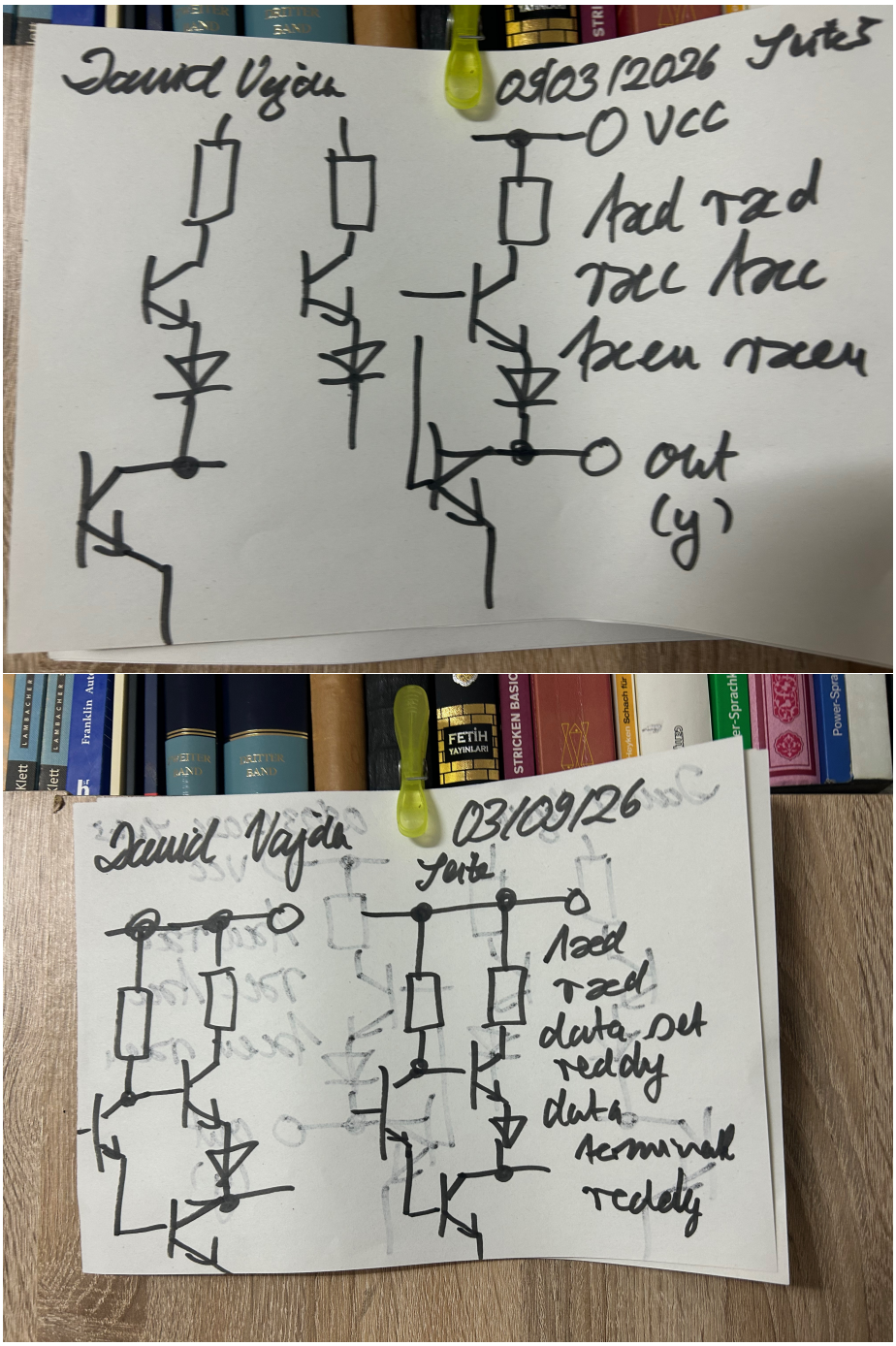
10.2 next

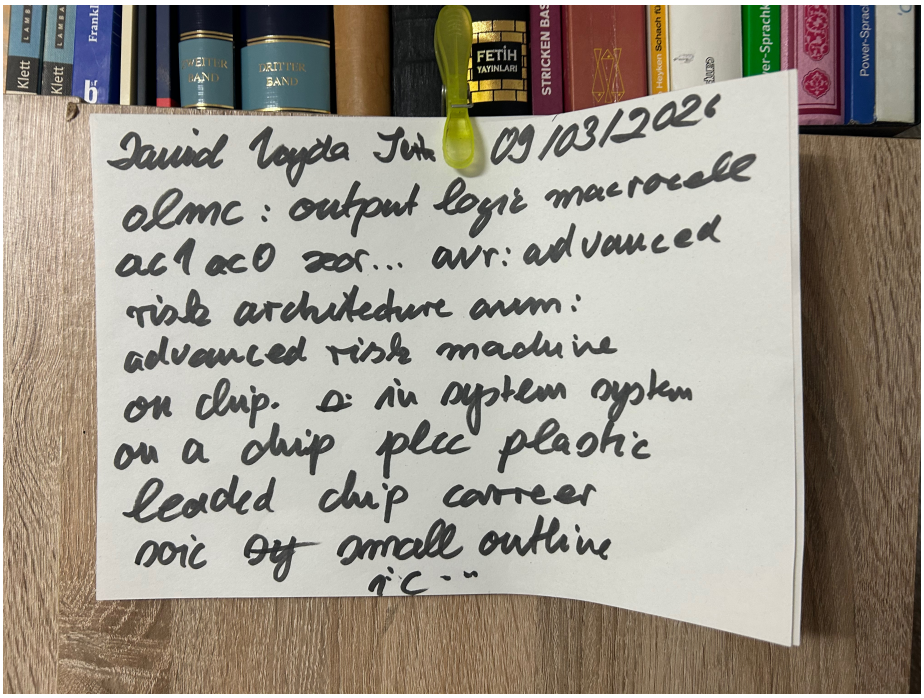



(c)
Junid
Bayda
3.9.
2026
Klett
Franklin Autobiog
FETİH
YAYINLARI
STRICKEN BAS
Enno Heyken Schach


(c) David Vayda 09/03/26
Seite 3
ok
otd
übung!


David Vayda 09/03/26
Seite 4
Axd
Txd
Txc
Txd
Axc
zw. übung data set ready
data terminal ready data set
ready data terminal ready


Vcc
out
(y)
FETİH
YAYINLARI
STRICKEN BASIC


Klett
Klett
FETİH
YAYINLARI
STRICKEN BAS
Power-Sprach
David Zayda Jut 09/03/2026
olmc: output logic macrocell
ac1 ac0 xor... avr: advanced
risk architecture avrm:
advanced risk machine
on chip. s: in system system
on a chip plcc plastic
leaded chip carreer
soic sop small outline
ic



10.3 next

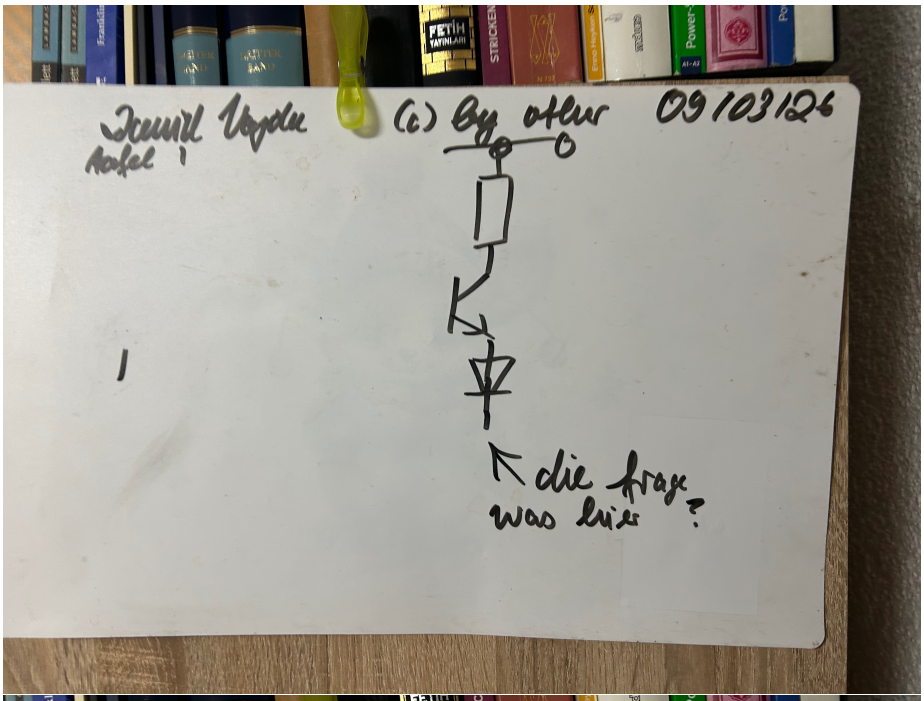


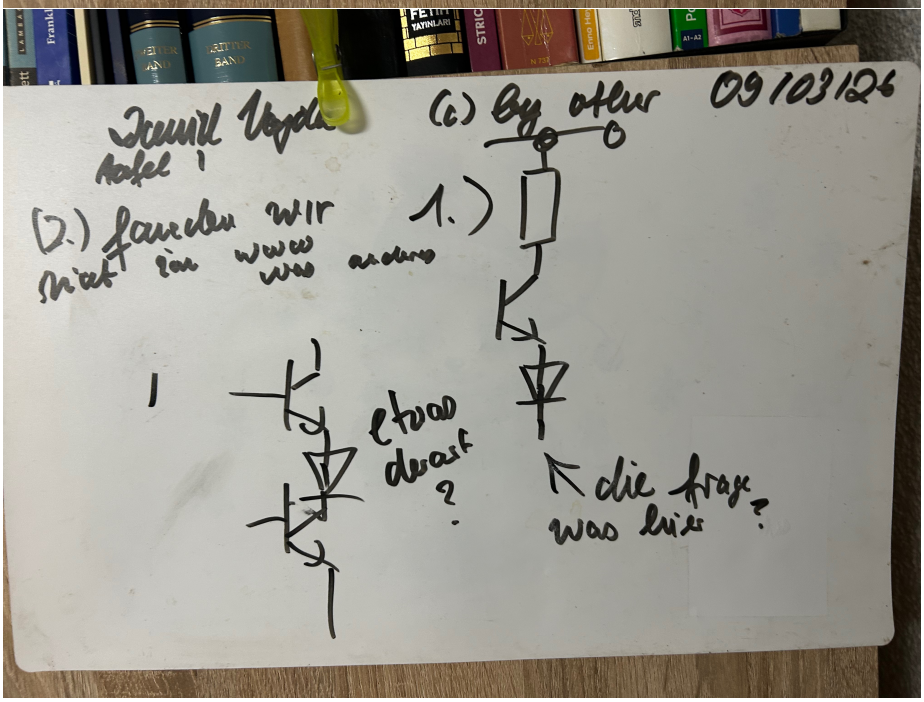


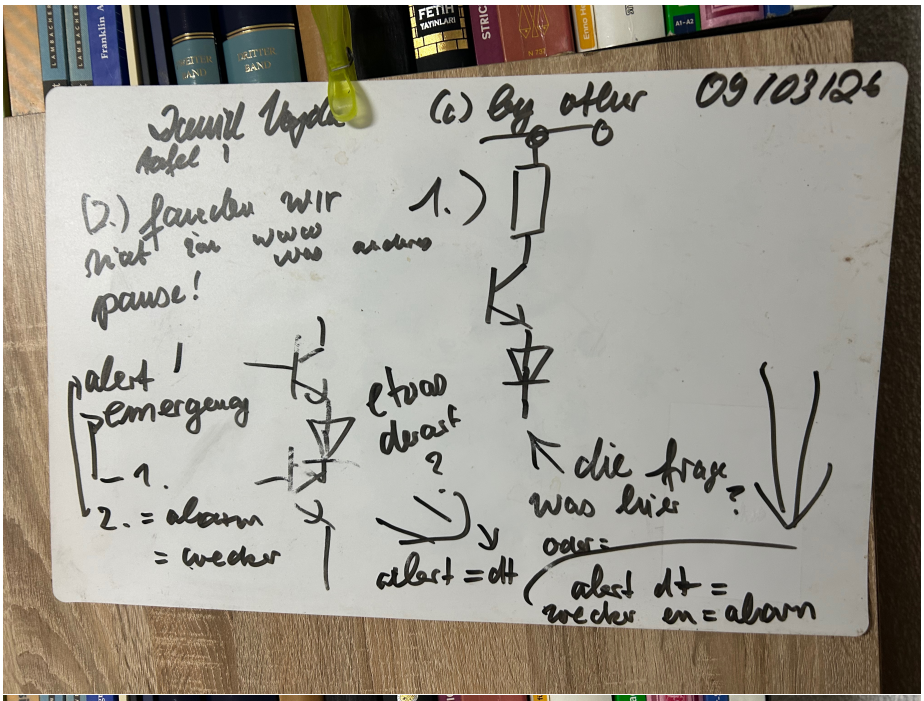


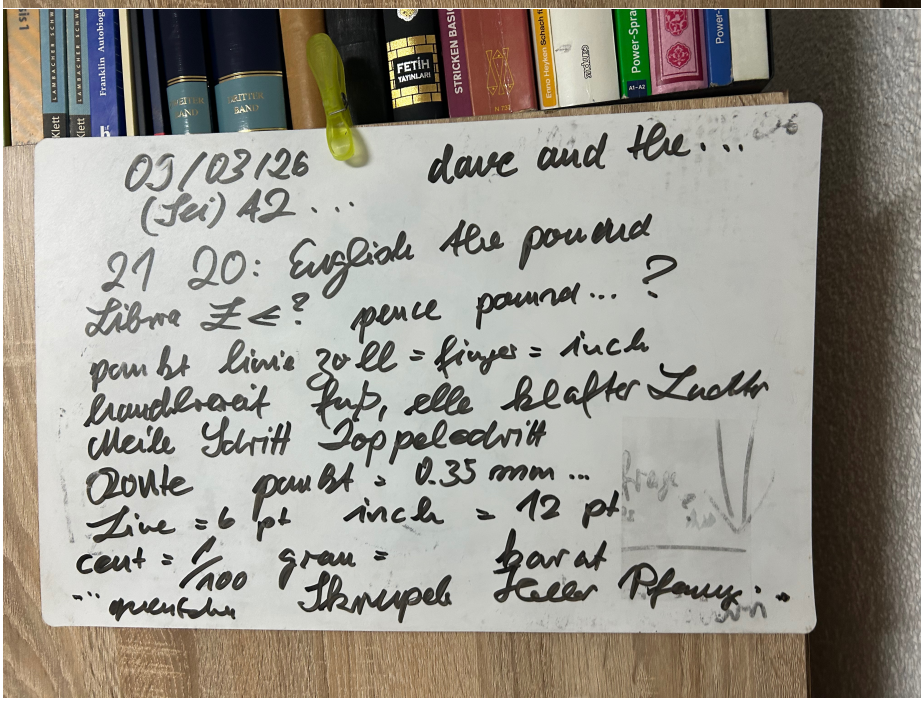





(c) By oder
09103126
die frage was lies?

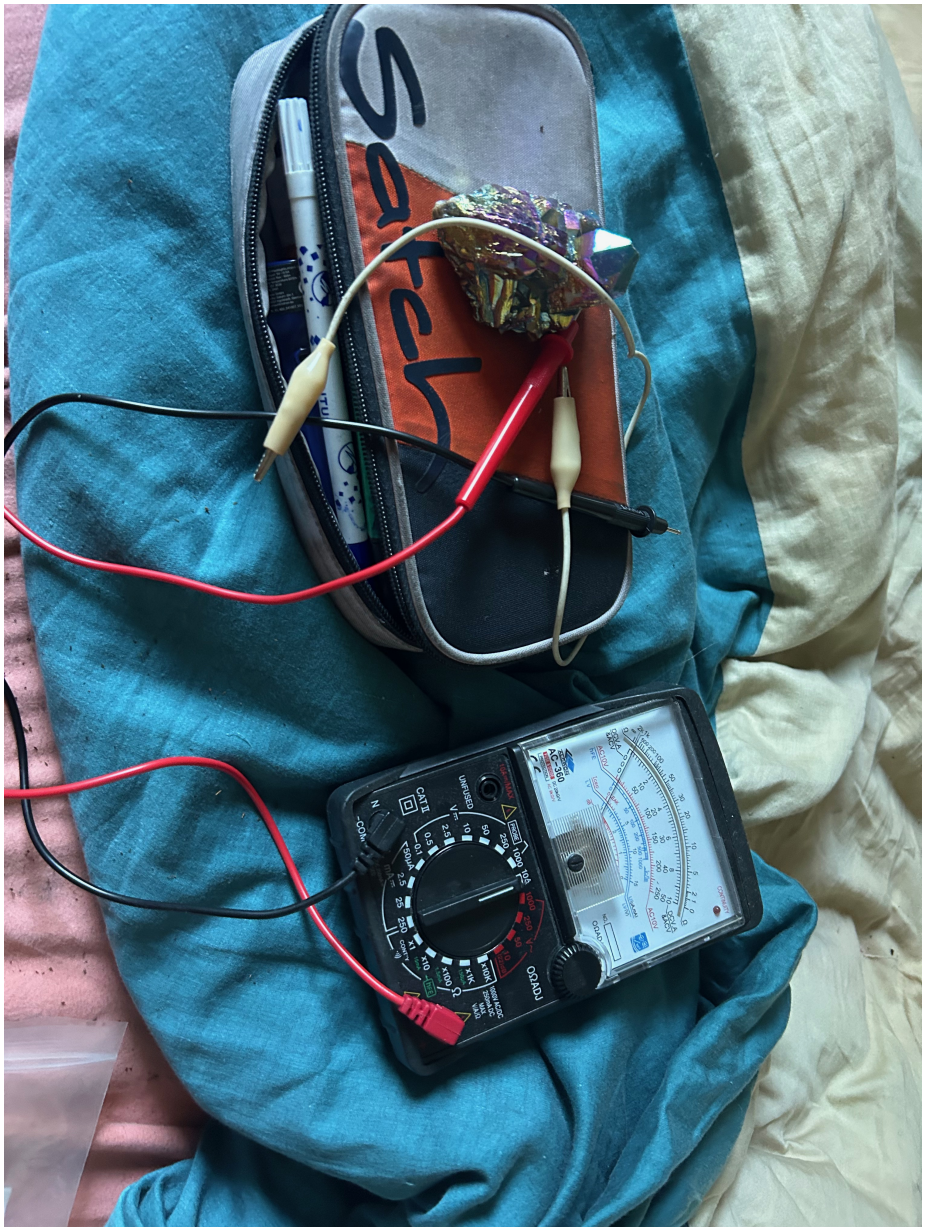

(c) By oder
09103126
1.)
(2.)
etwas dasst?
die frage was lies?


(c) by other
09103126
1.)
2.) fangen wir
nicht an wwww
wo anders
pause!
alert !
emergency
1.
2. = alarm
= wecker
etwas
derart
?
die frage
was lies ?
oder:
alert dt =
wecker en = alarm


09/03126
dave and the...
(sei) 42...
21 20: English the pound
Libra £ ? pence pound... ?
point linie zoll = finger = inch
handbreit fuß, elle klafter Lachter
meile schritt doppelschritt
point = 0.35 mm...
Zeile = 6 pt inch = 12 pt
cent = 1/100 gram = karat
... quentchen skrupel heller pfennig...


atch

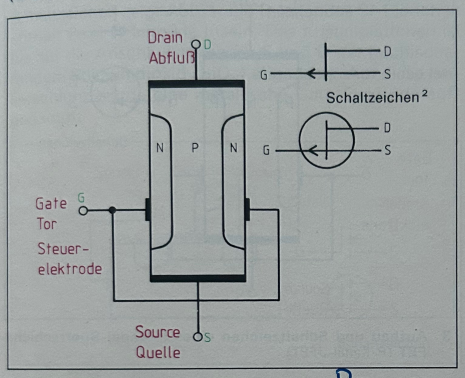
10.4 next



Laststrom stets nur eine Kristallzone gleicher Leitfähigkeitsart, also entweder eine N-Zone oder eine P-Zone, durchfließt, werden diese Transistoren auch **Unipolar-Transistoren** genannt.

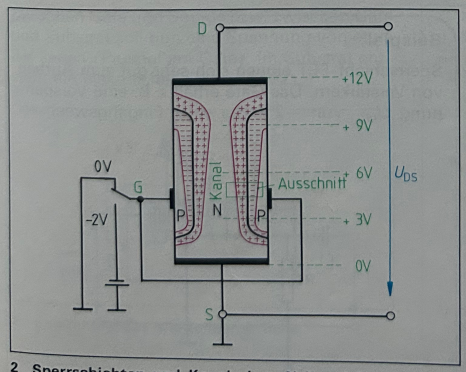
Sperrschicht-Feldeffekt-Transistoren (JFET[1])

Abb. 1 zeigt den Grundaufbau eines N-Kanal-Sperrschicht-FET. In ein N-leitendes Kristallplättchen sind zwei P-leitende Zonen eindotiert. Die Elektroden haben die Bezeichnungen S **Source** (engl., Quelle), D **Drain** (engl., Abfluß) und G **Gate** (engl., Tor).



1 Grundaufbau und Schaltzeichen eines N-Kanal-Sperrschicht-FET (N-Kanal-JFET)

An den Drainanschluß wird eine positive Spannung gegen Source angelegt (z. B. +12 V). Durch die N-Zone fließt ein Elektronenstrom. Entlang der Kristallstrecke entsteht ein Spannungsabfall, wie in Abb. 2 dargestellt.

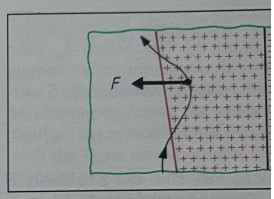


2 Sperrschichten und Kanal eines N-Kanal-Sperrschicht-FET

Beim Sperrschicht-FET fin gänge stets in Sperrichtun

Es entstehen Sperrschichten, wi **Diese Sperrschichten sind fü botene Gebiete.**

In den Sperrschichten herrscht e Feldlinien verlaufen von den po negativen Ladungen. Gerät ein schicht, so erfährt es eine Kraftw zur Feldlinienrichtung. Abb. 3 aus der Sperrschicht. Auf das wirkt die Kraft *F*. Das Elektron w herausgedrängt.



3 Ausschnitt aus der Sperrschic

Als Strömungsweg steht der neutrale Bereich zv schichten, der sogenannt gung.

Legt man an den Gate-Ansch nung gegen Source, z. B. U_{GS} = den PN-Übergängen liegende 2 V erhöht. **Die Sperrschicht Kanal wird schmaler.**

Je negativer die Spannu ringer wird der Kanalque ßer wird der Kanalwider

Mit zunehmendem Kanalwi durch den Kanal fließende Str nannt wird. Für eine Änderung praktisch keine Leistung erfor

Der Drainstrom I_D wird U_{GS} leistungslos gesteue

[1] JFET: Junction-Feldeffekttransistor
[2] Der für das Gehäuse stehende Kreis fallen. Bei integrierten Schaltungen

on Feld-
nsistoren
e. Da der
eitfähig-
P-Zone,
nipolar-

oren

al-Sperr-
chen sind
den haben
D **Drain**

D
S
zeichen[2]
D
S

-Kanal-Sperr-

pannung ge-
die N-Zone
ristallstrecke
2 dargestellt.

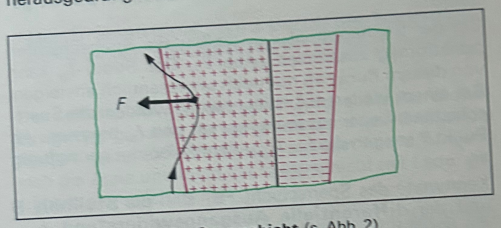
12V
9V
6V
3V
0V
U_{DS}

muß stets so sein.

Beim Sperrschicht-FET müssen die PN-Übergänge stets in Sperrichtung gepolt sein.

Es entstehen Sperrschichten, wie in Abb. 2 angegeben. **Diese Sperrschichten sind für die Elektronen verbotene Gebiete.**

In den Sperrschichten herrscht ein elektrisches Feld. Die Feldlinien verlaufen von den positiven Ladungen zu den negativen Ladungen. Gerät ein Elektron in eine Sperrschicht, so erfährt es eine Kraftwirkung entgegengesetzt zur Feldlinienrichtung. Abb. 3 zeigt einen Ausschnitt aus der Sperrschicht. Auf das eingedrungene Elektron wirkt die Kraft F. Das Elektron wird aus der Sperrschicht herausgedrängt.



3 Ausschnitt aus der Sperrschicht (s. Abb. 2)

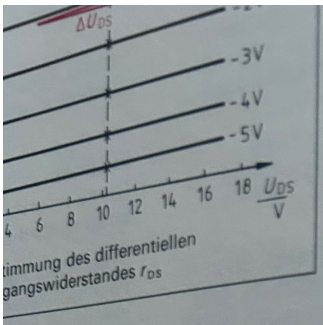
Als Strömungsweg steht den Elektronen nur der neutrale Bereich zwischen den Sperrschichten, der sogenannte Kanal, zur Verfügung.

Legt man an den Gate-Anschluß eine negative Spannung gegen Source, z. B. $U_{GS} = -2\,V$, so werden die an den PN-Übergängen liegenden Sperrspannungen um 2 V erhöht. **Die Sperrschichten werden breiter, der Kanal wird schmaler.**

Je negativer die Spannung U_{GS} ist, desto geringer wird der Kanalquerschnitt, desto größer wird der Kanalwiderstand.

Mit zunehmendem Kanalwiderstand sinkt aber der durch den Kanal fließende Strom, der Drainstrom I_D genannt wird. Für eine Änderung der Sperrschichtbreite ist praktisch keine Leistung erforderlich.

Der Drainstrom I_D wird durch die Spannung U_{GS} leistungslos gesteuert.



timmung des differentiellen
gangswiderstandes r_{DS}

hicht-FET

ines N-Kanal-
nderer Kennli-
nürpunkt ge-

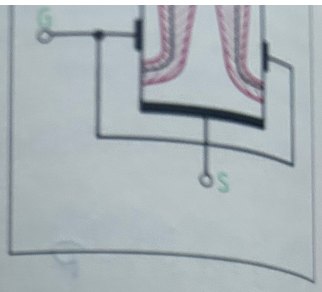
Sperrschichten
Strom I_D kann
tlich ansteigen.
den die Sperr-
$_D$ bremsen. Ab
nur geringfügig

ie **Steilheit** ***S***
derstand r_{DS}.
ann eine Steil-
ilt:

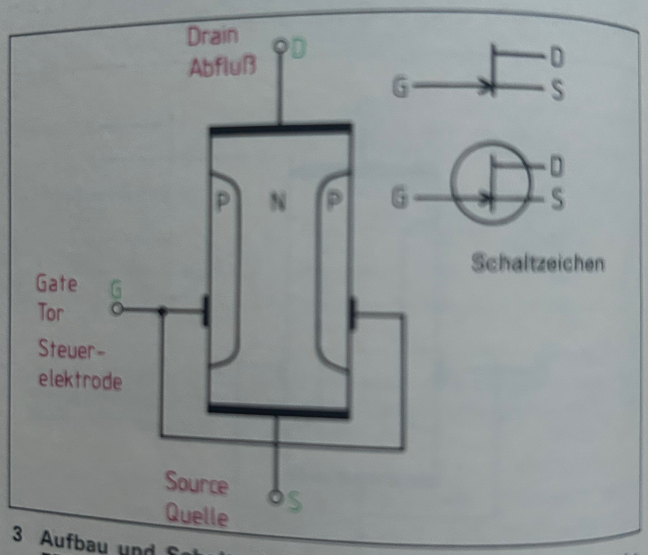
ng
derung

ennlinienfeldes
angswiderstand

gangswiderstand
nderung
ung



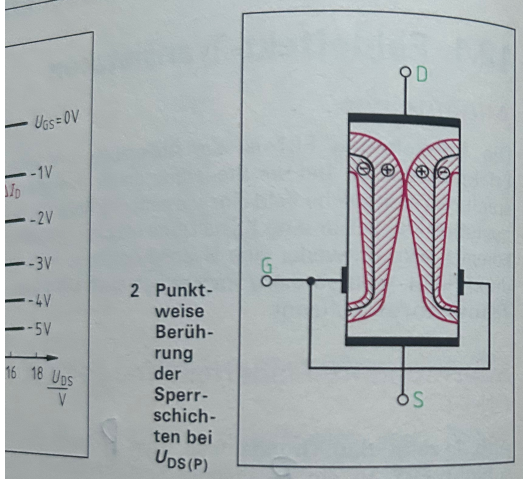
2 Punktweise Berührung der Sperrschichten bei $U_{DS(P)}$



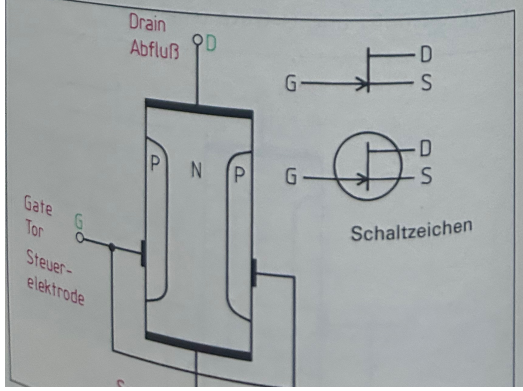
3 Aufbau und Schaltzeichen eines P-Kanal-Sperrschicht-FET (P-Kanal-JFET)

Die für den N-Kanal-Sperrschicht-FET angegebenen Kennlinien (Abb. 1) gelten entsprechend auch für den P-Kanal-Sperrschicht-FET. Die Polung aller Spannungen ist umzukehren. Die Kennwerte des P-Kanal-Sperrschicht-FET ergeben sich wie beim N-Kanal-Typ.

Beispiel:



2 Punktweise Berührung der Sperrschichten bei $U_{DS(P)}$



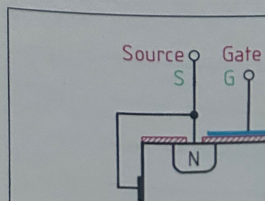
spannung von $\hat{u}_1 = 0{,}2$
Gatespannung von $-1{,}8$
wird entsprechend geste
werden mit Hilfe von $R_$
umgewandelt. Am Ausga
stärkte Wechselspannun
men werden.

MOS-Feldeffekt-T

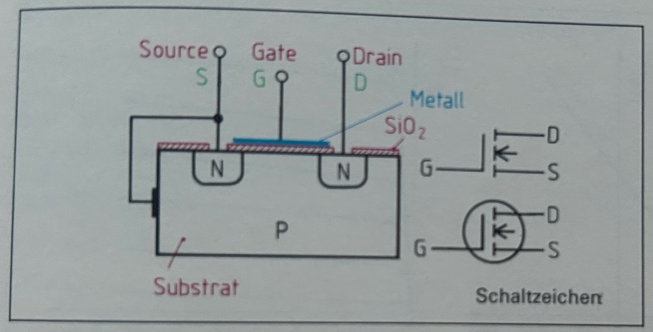
Die Bezeichnung MOS is
Oxide-Semiconductor od
ergibt sich aus dem Aufba
storen. Man unterscheidet
selbstleitende MOS-FE

Selbstsperrende MO

Zunächst soll der häufig
N-Kanal-MOS-FET be
Schaltzeichen zeigt Abb.
enthält zwei N-leitende I
bis auf die Anschlußstell
rendem Siliciumdioxid (S
lierschicht wurde eine m
gedampft.



N-Kanal-MOS-FET betrachtet werden. Aufbau und Schaltzeichen zeigt Abb. 1. Ein P-leitendes Si-Plättchen enthält zwei N-leitende Inseln. Das Kristallplättchen ist bis auf die Anschlußstellen für S und D mit hochisolierendem Siliciumdioxid (SiO_2) abgedeckt. Auf diese Isolierschicht wurde eine metallische Gate-Elektrode aufgedampft.



1 Aufbau und Schaltzeichen eines selbstsperrenden N-Kanal-MOS-FET (N-Kanal-IGFET)

Legt man zwischen S und D eine beliebige gepolte Spannung, so fließt kein Strom. Zwischen N-Material und P-Material bauen sich Sperrschichten auf. Der FET sperrt.

Zwischen Gate und Source bzw. zwischen Gate und Substrat soll nun eine Spannung von etwa 3 V liegen (Pluspol am Gate). Im Substrat entsteht ein elektrisches Feld (Abb. 2).

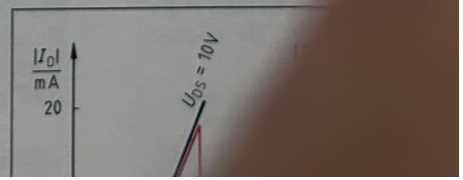
Brücke vergrößert. Mit abneh
wird die Leitfähigkeit vermin
Leitfähigkeit wird keine Leistu

Der Drainstrom I_D wird
nung (U_{GS}) leistungslos

Der selbstsperrende MOS-FE
rungstyp genannt. Es kann
fließen, wenn sich die Brücke
sich die Zone unterhalb der Is
trägern angereichert hat.

Die Kennlinienfelder eines se
sind in Abb. 3 dargestellt. A
linienfeld ist deutlich zu erke
Gatespannung U_{GS} von etwa
Brücke zu erzeugen. Bei
fließt noch kein Strom.

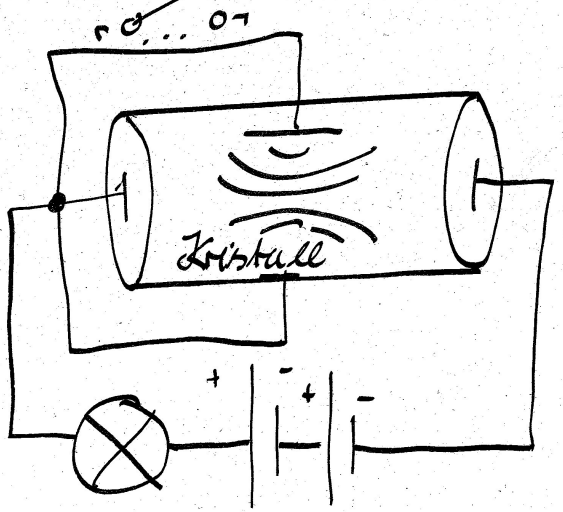
Kennwerte des selbstsp
Steilheit S und der di
stand r_{DS}. Für jeden
kann eine Steilheit a
erfolgt gemäß Abb



David Vayda, 09/04/2026

Seite 1 fet

feld effekt transistor mit Kristall...

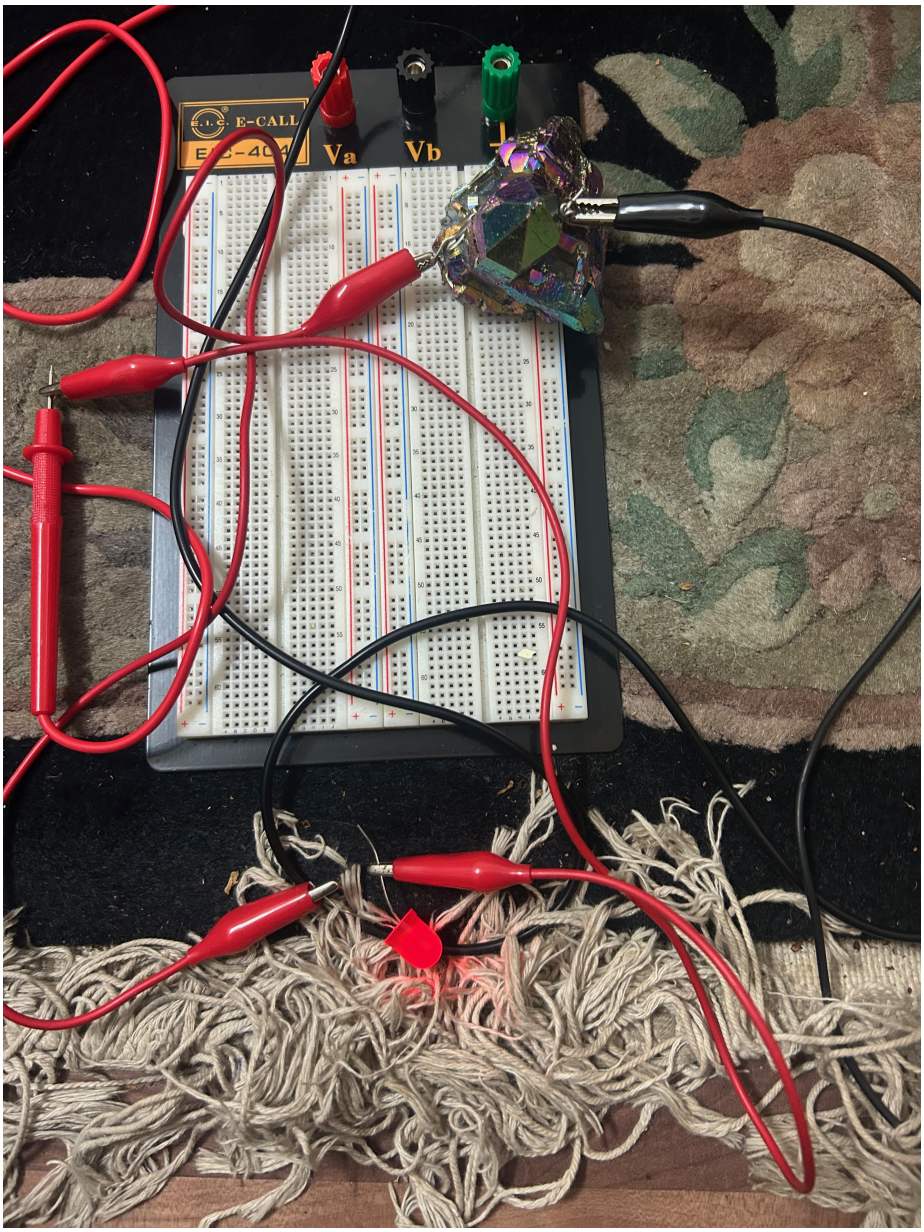



AC-360
UNFUSED
CAT II
-COM
OΩADJ
PROBE
CONTINUITY
AC10V
DCV.A
&ACV
hFE
x10K
x1K
x100
x10
x1
CONT'Y
50μA
mA
V
Ω

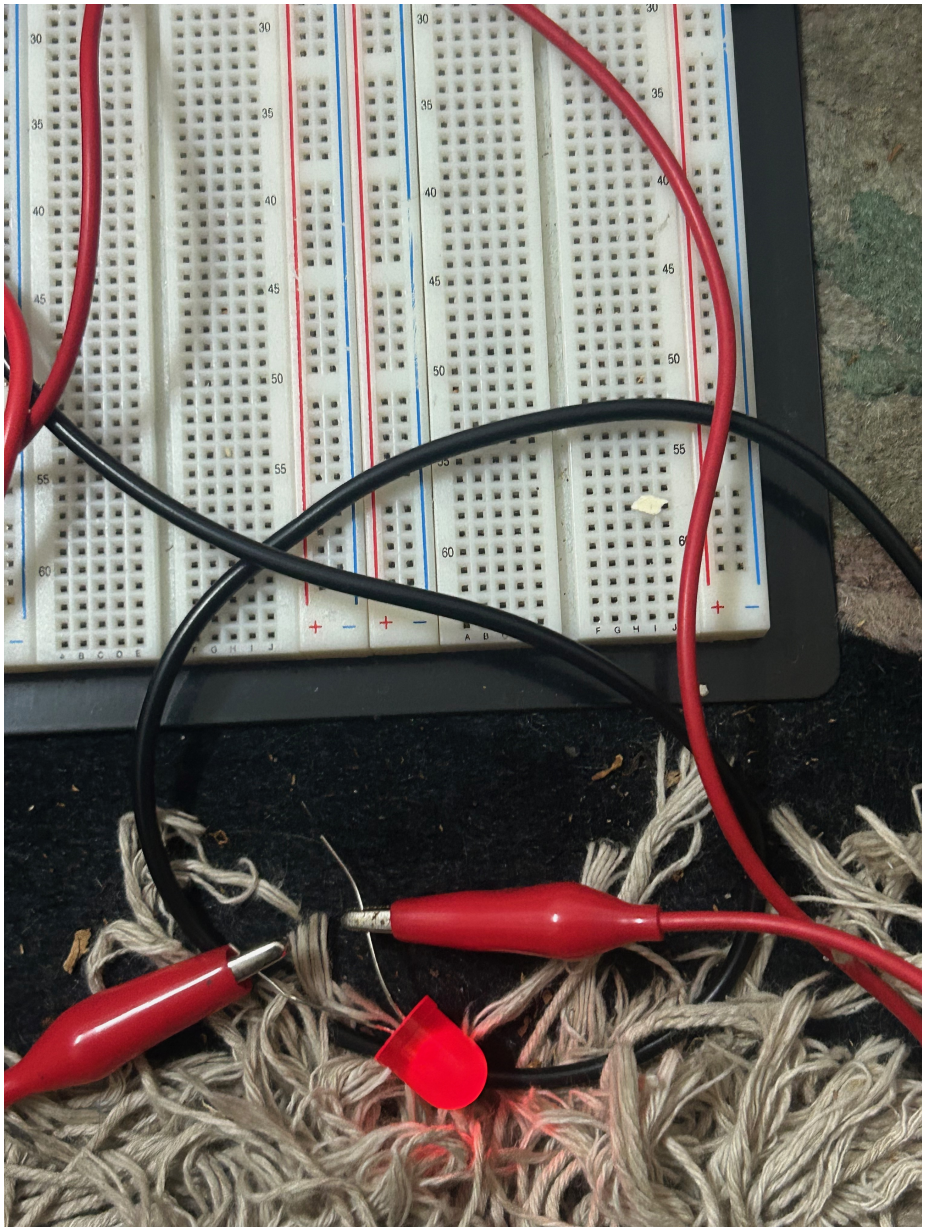

CONTINUITY
DCV.A
&ACV
AC10V
hFE
ALCRON
AC-360
NO.
10A⎓/MAX
PROBE
UNFUSED
V⎓
CAT II
N
-COM
mA⎓
MAX
1000V AC/DC 250mA⎓
FUSED
CONT"Y
hFE
OΩADJ
V~
1000V AC/DC
250mA DC
MAX
V/A/Ω
x10K
x1K
x100
x10
x1
Ω
P
+

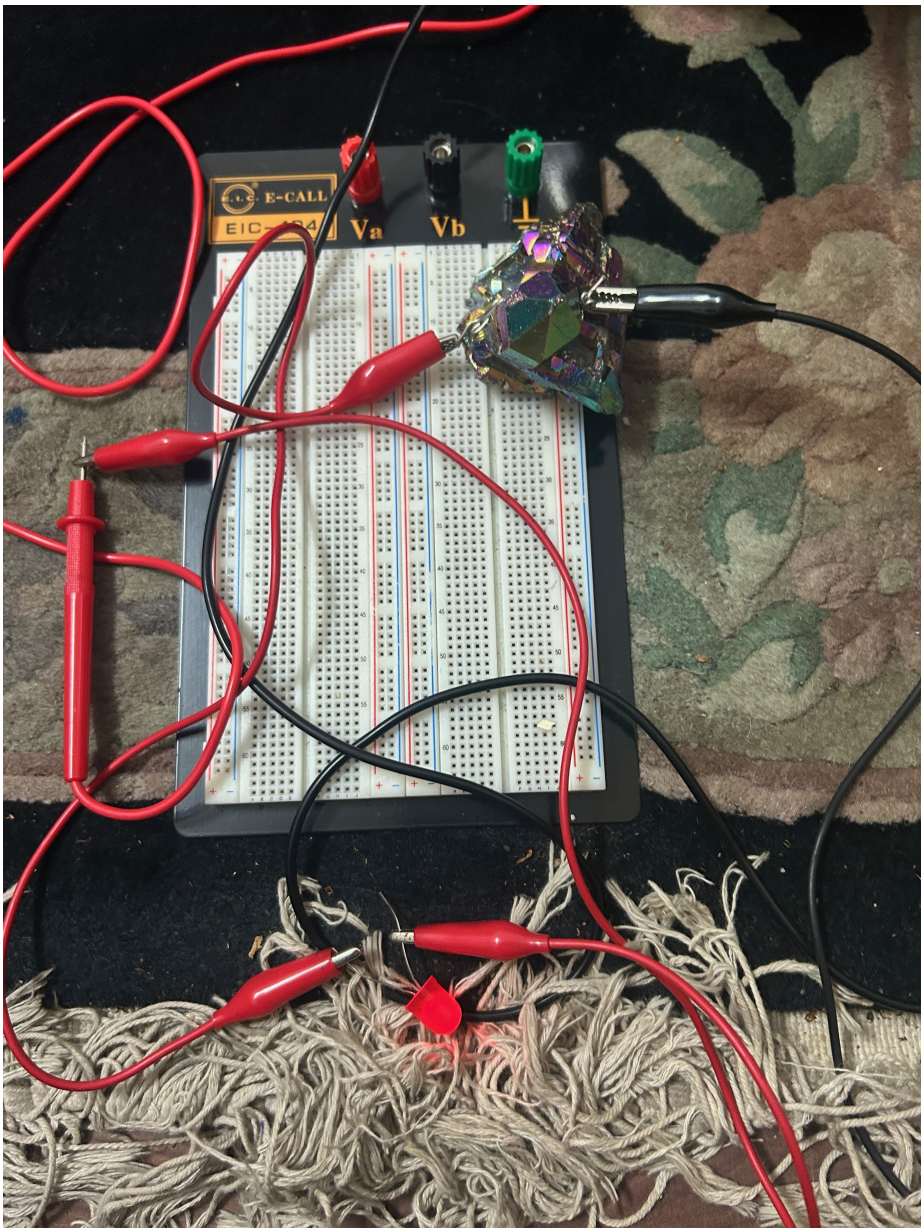

KORAD
U203 DIGITAL CONTROL DC
POWER SUPPLY 0-30V, 0-10A
C.V
10.00
V
0.0000
A
0000
VOLTAGE
COARSE


E-CALL
Vb

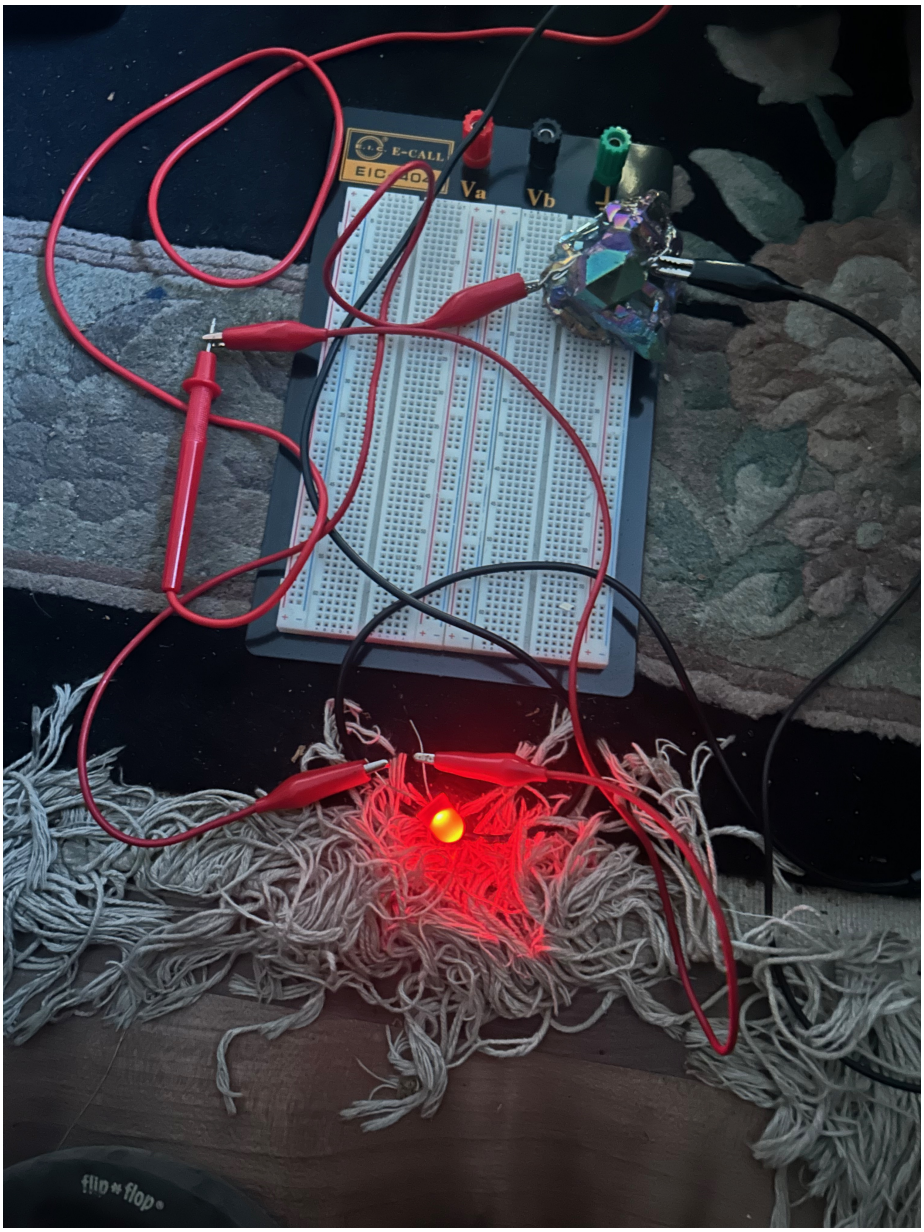

E-CALL
Va
Vb

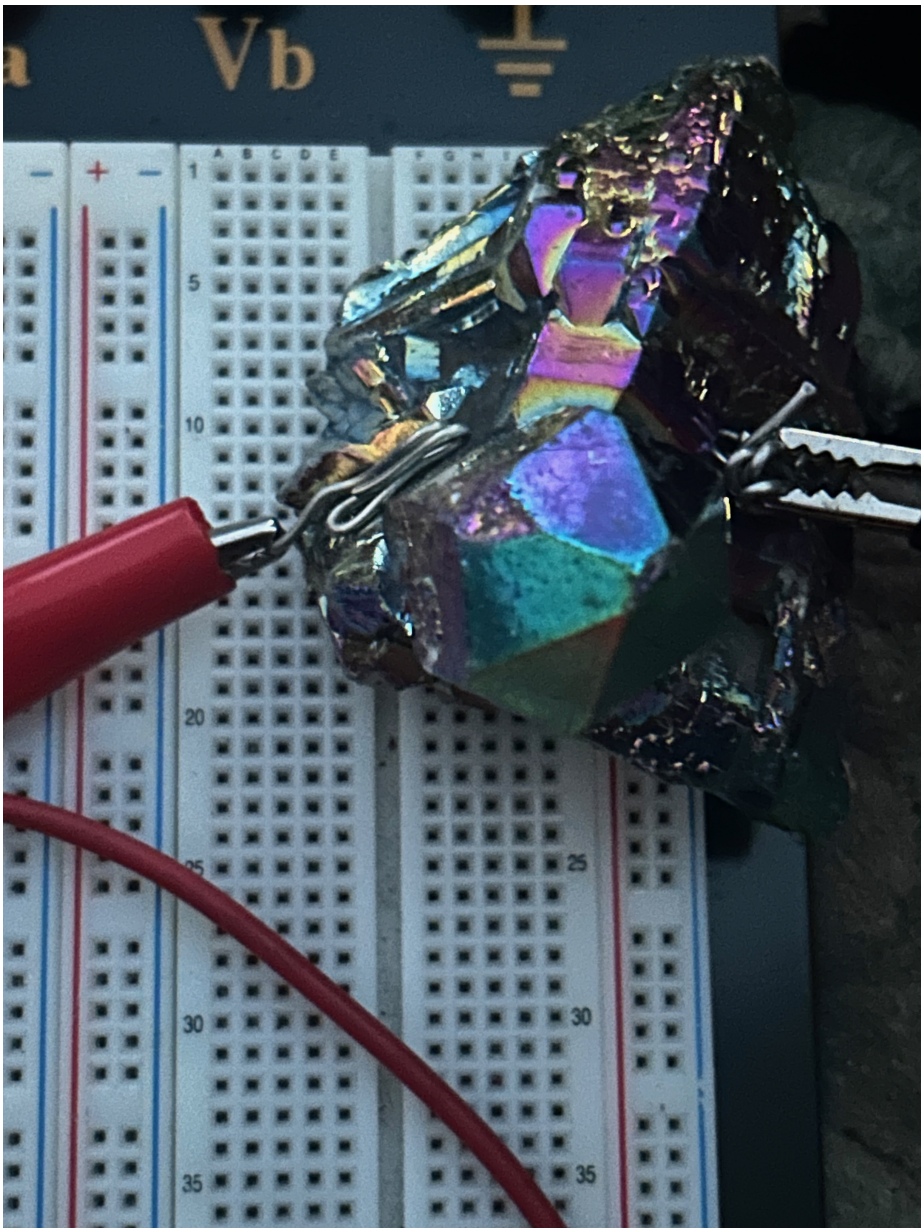
10.5 next

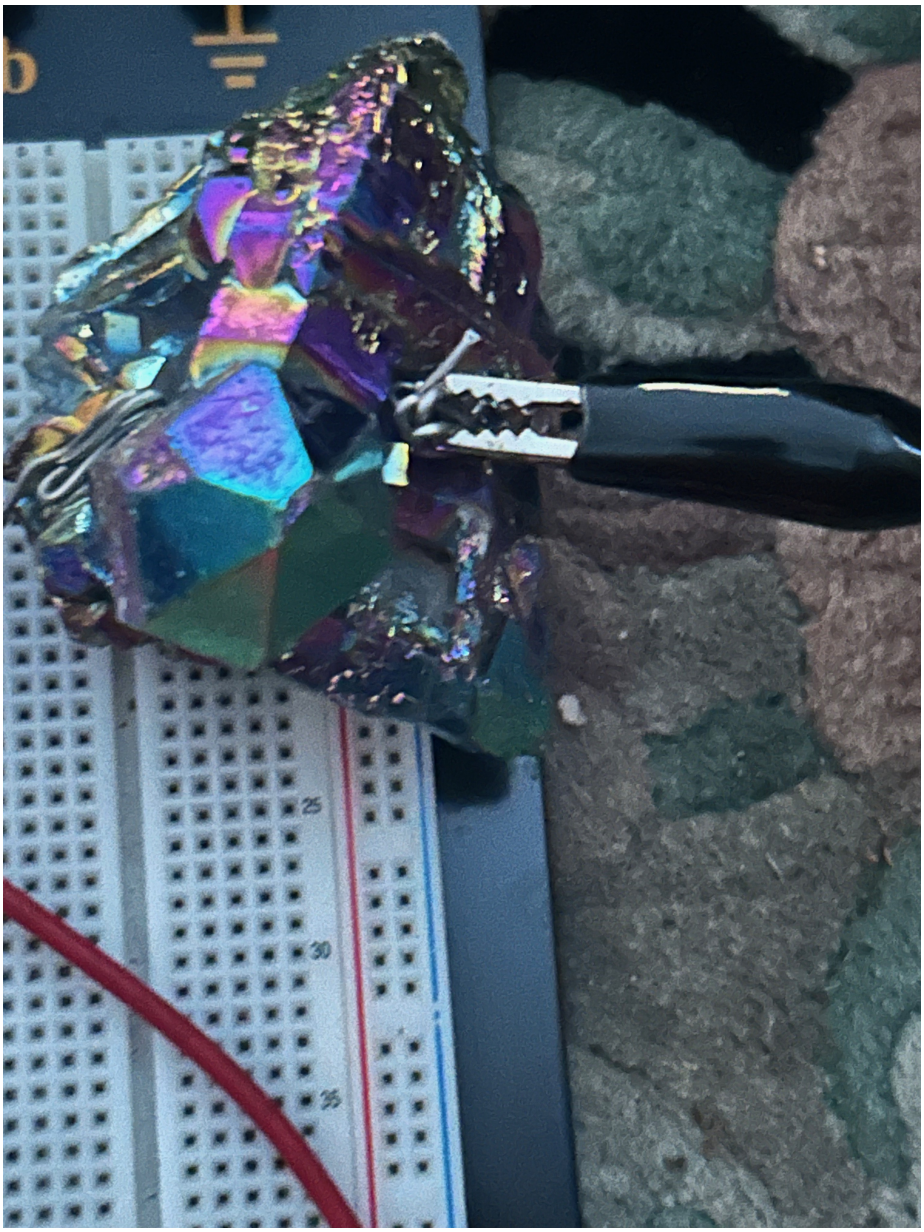



E-CALL
Va
Vb


WARNING
08.00
0.0000
0000
E-CALL
EIC-404
Va
Vb

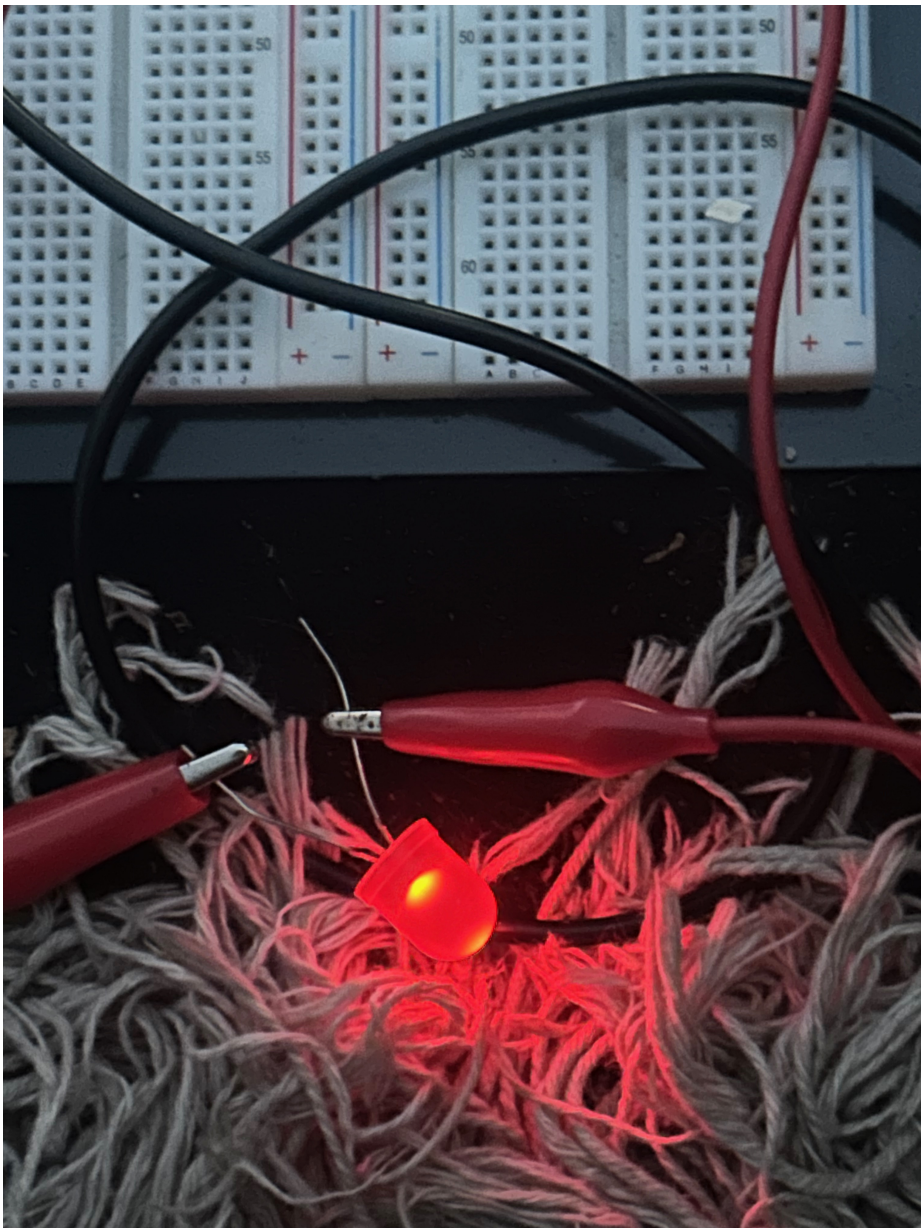

E-CALL
Va
Vb
flip flop

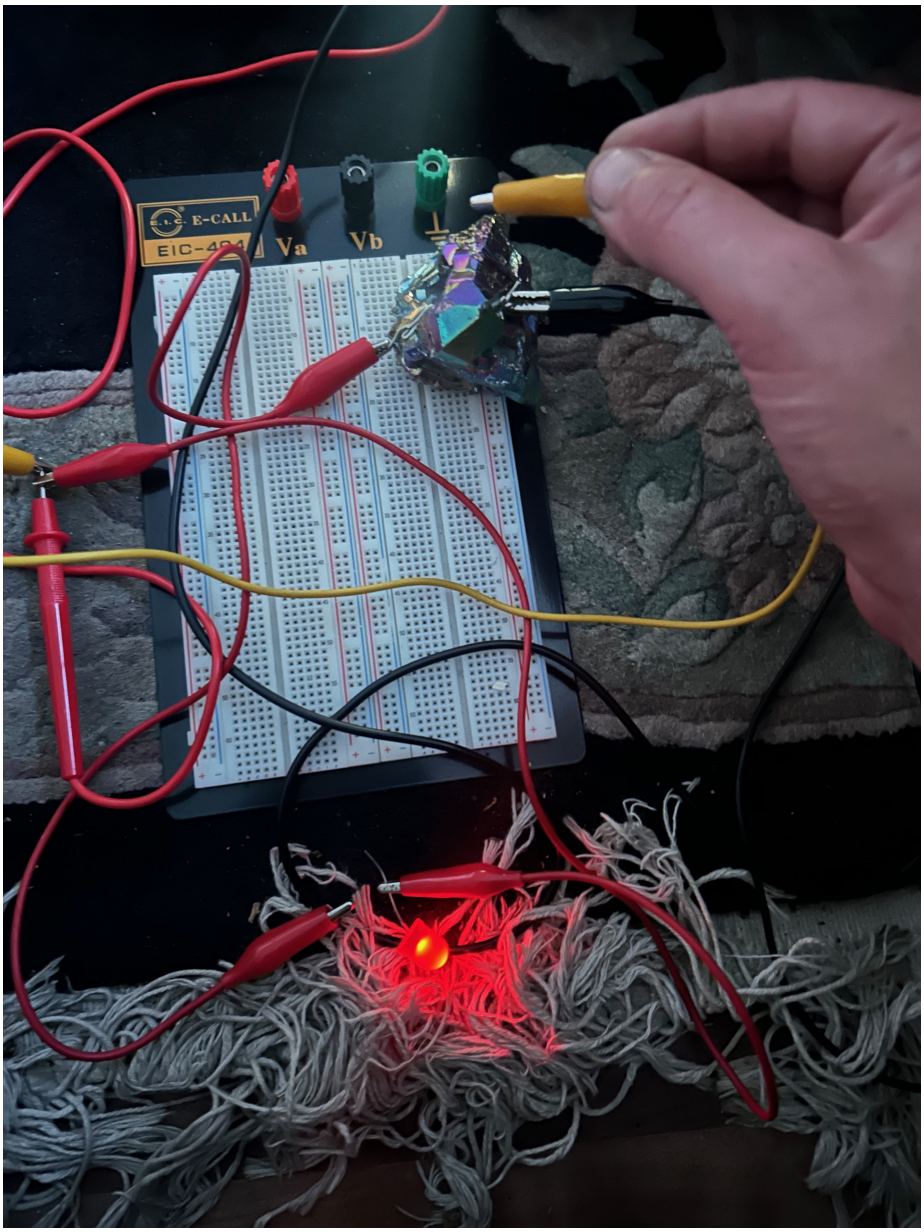

Vb
1
5
10
20
25
30
35

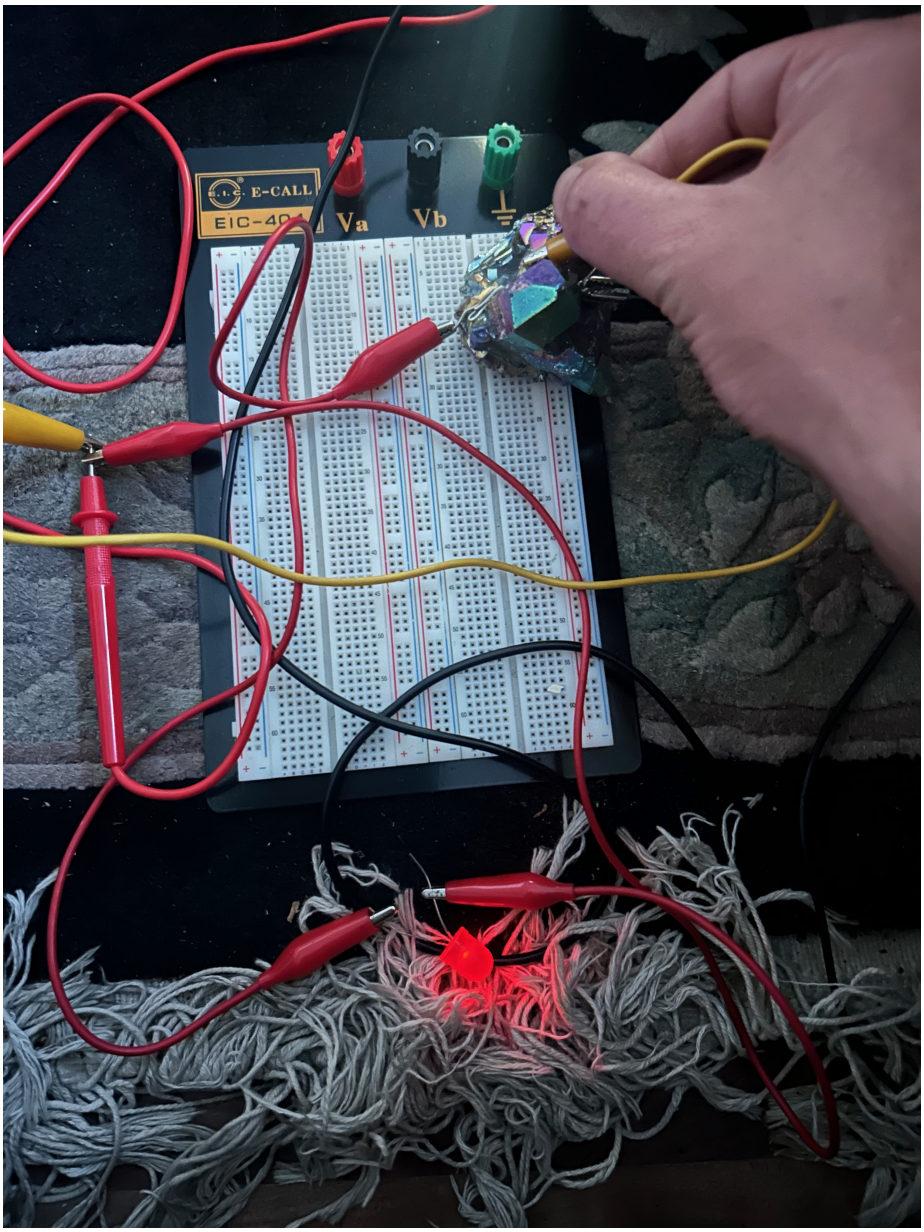

25
30
35

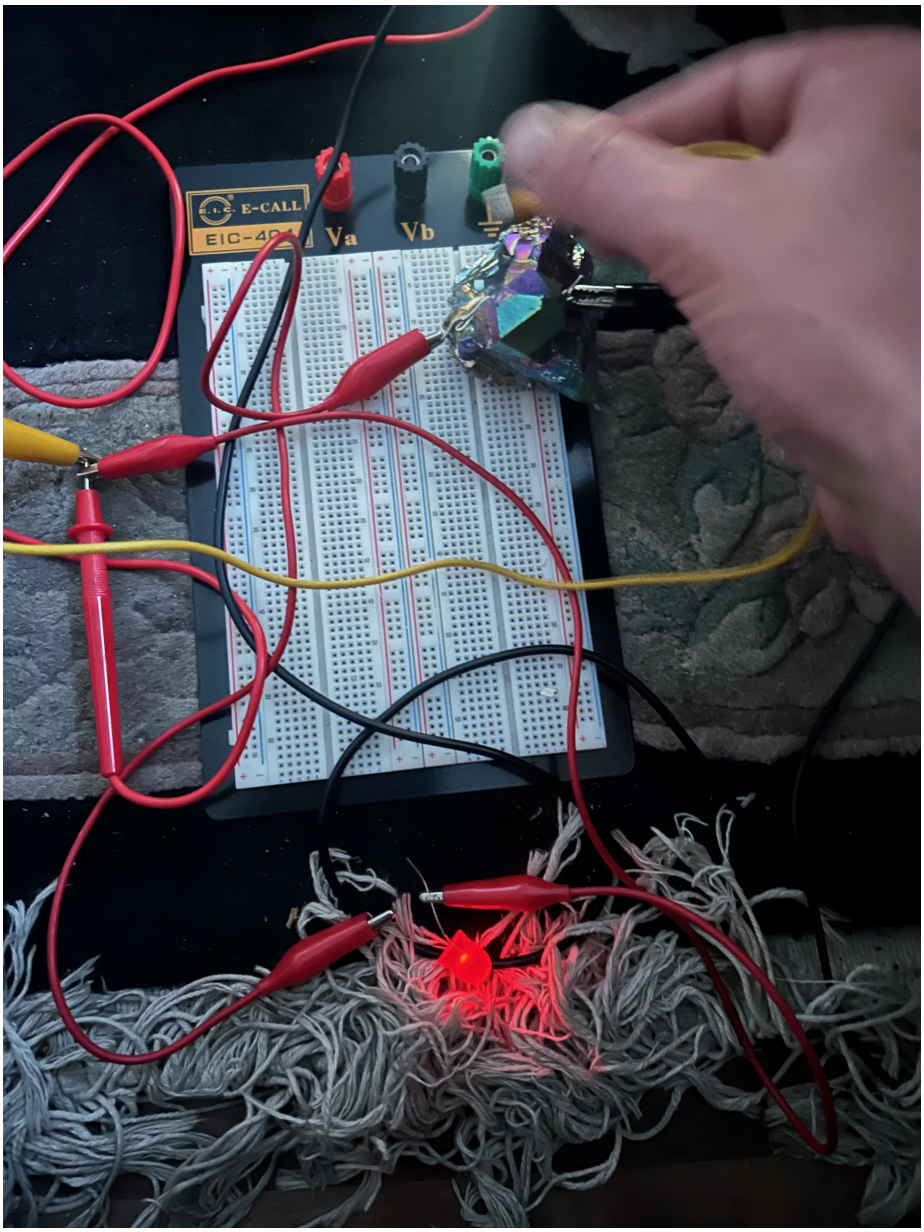

CHINA

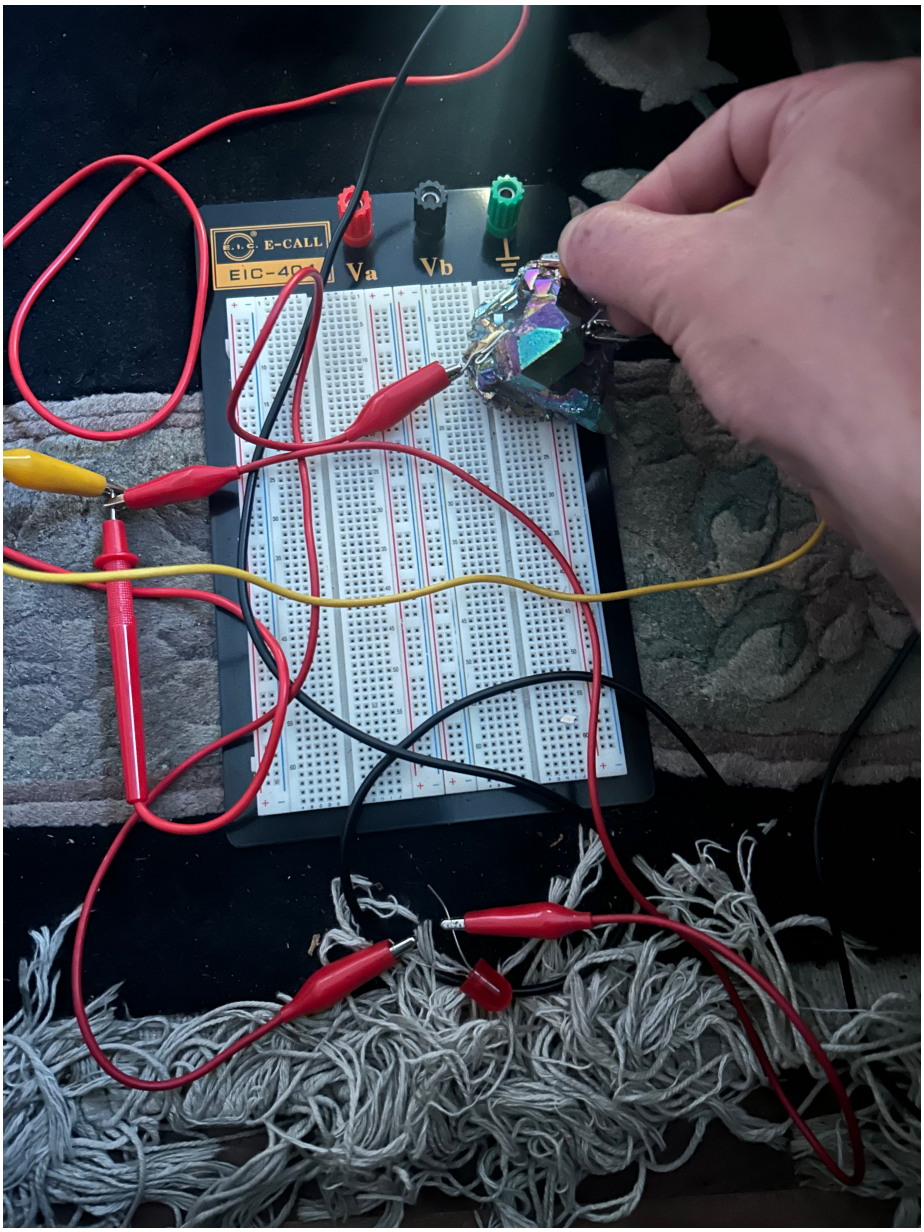

0 5.0 0
0.0 0 0 0
0 0 0.0

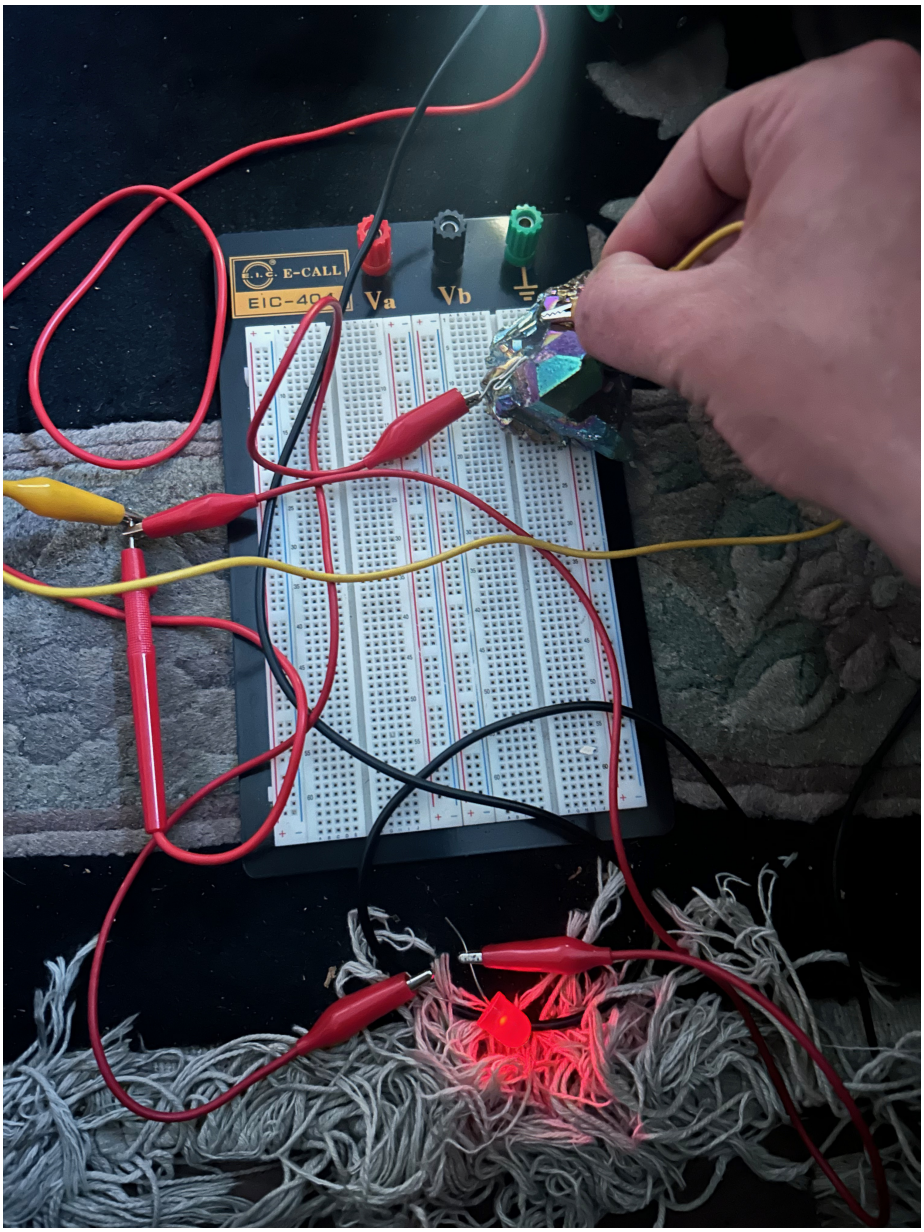


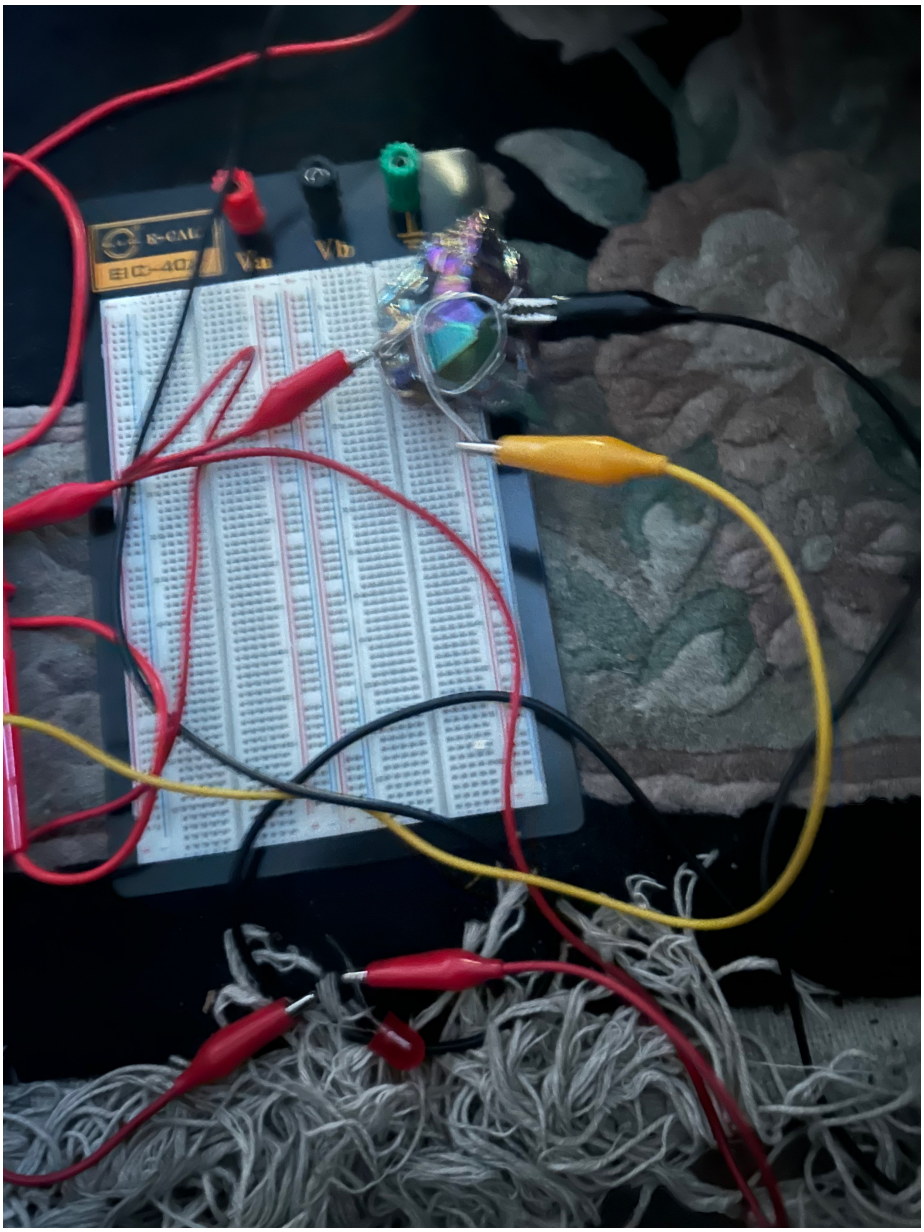

E-CALL
Va
Vb

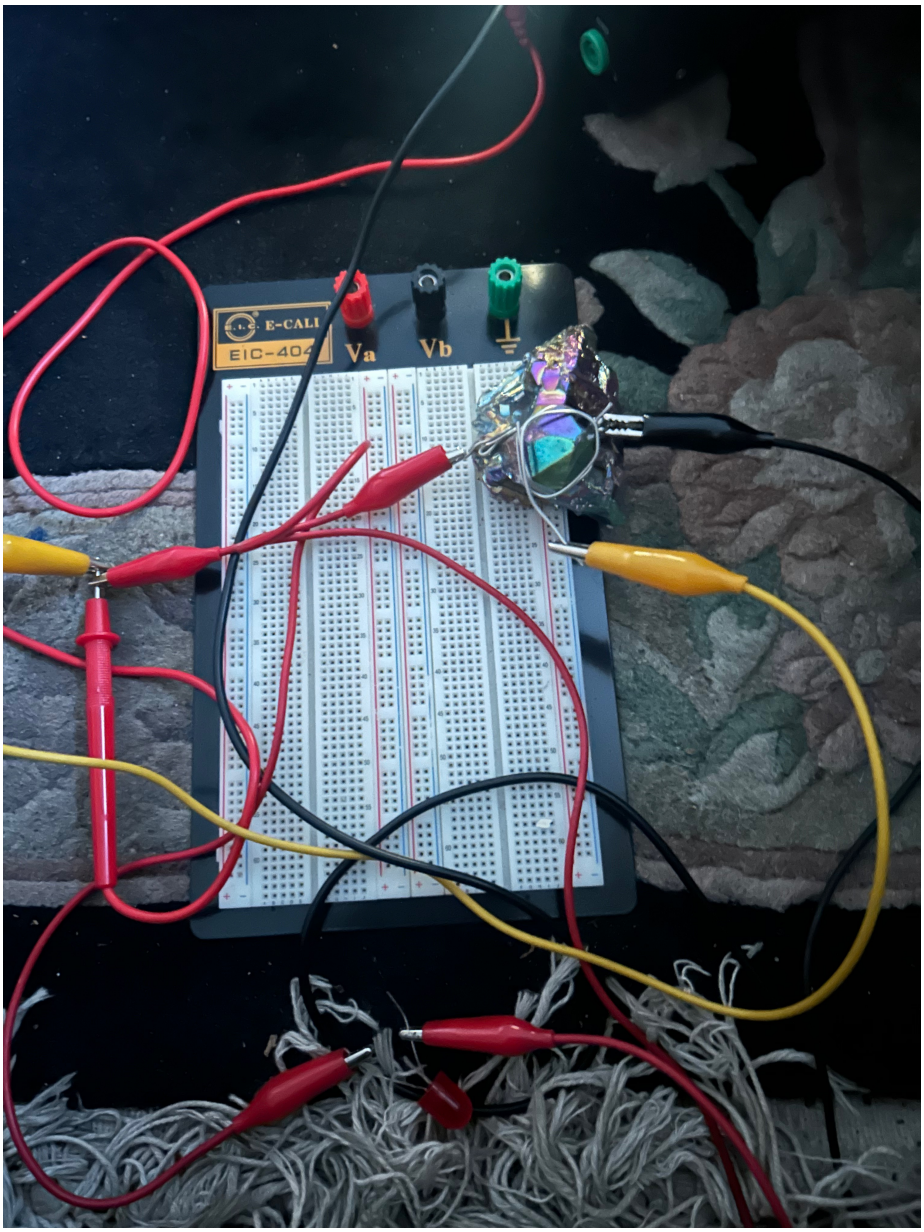

E-CALL
Va
Vb

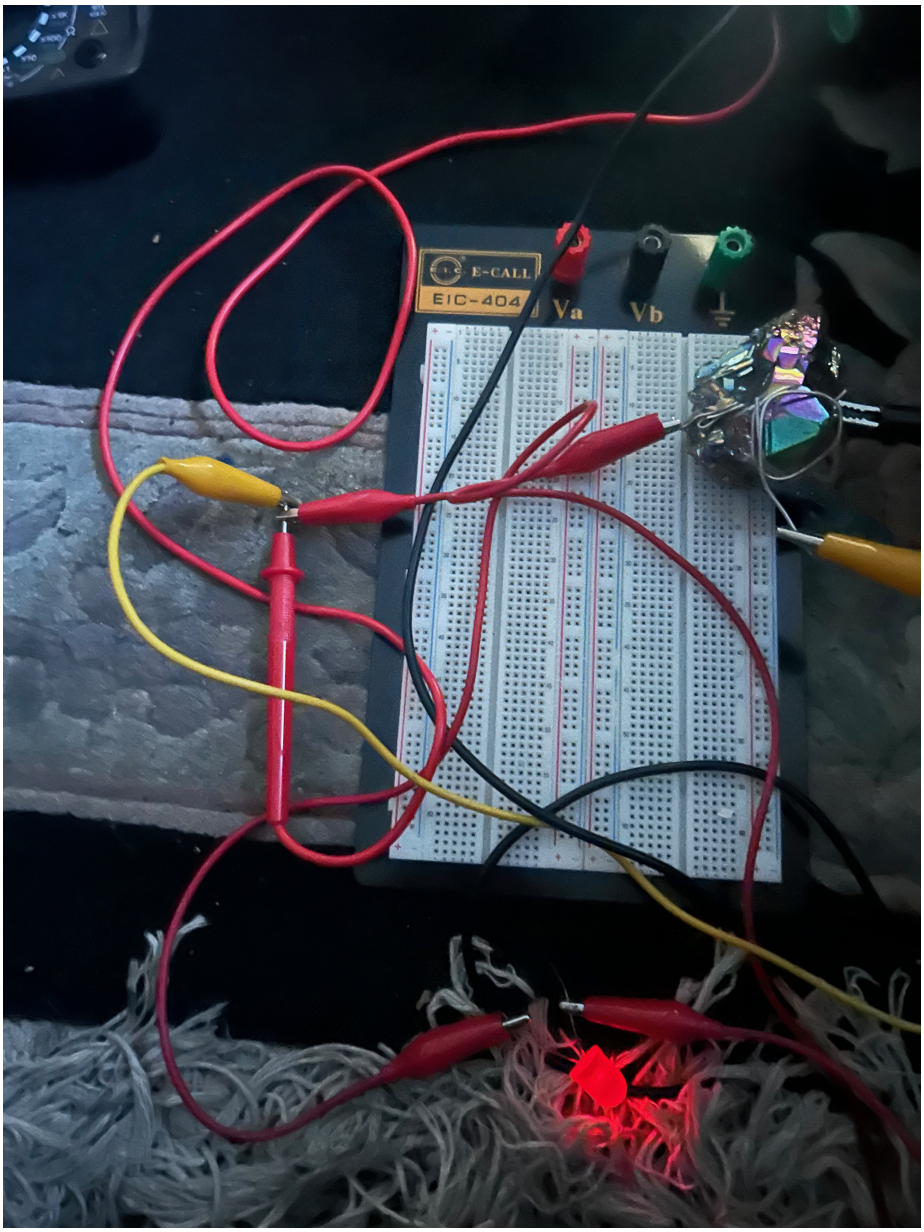

E-CALL
Va
Vb

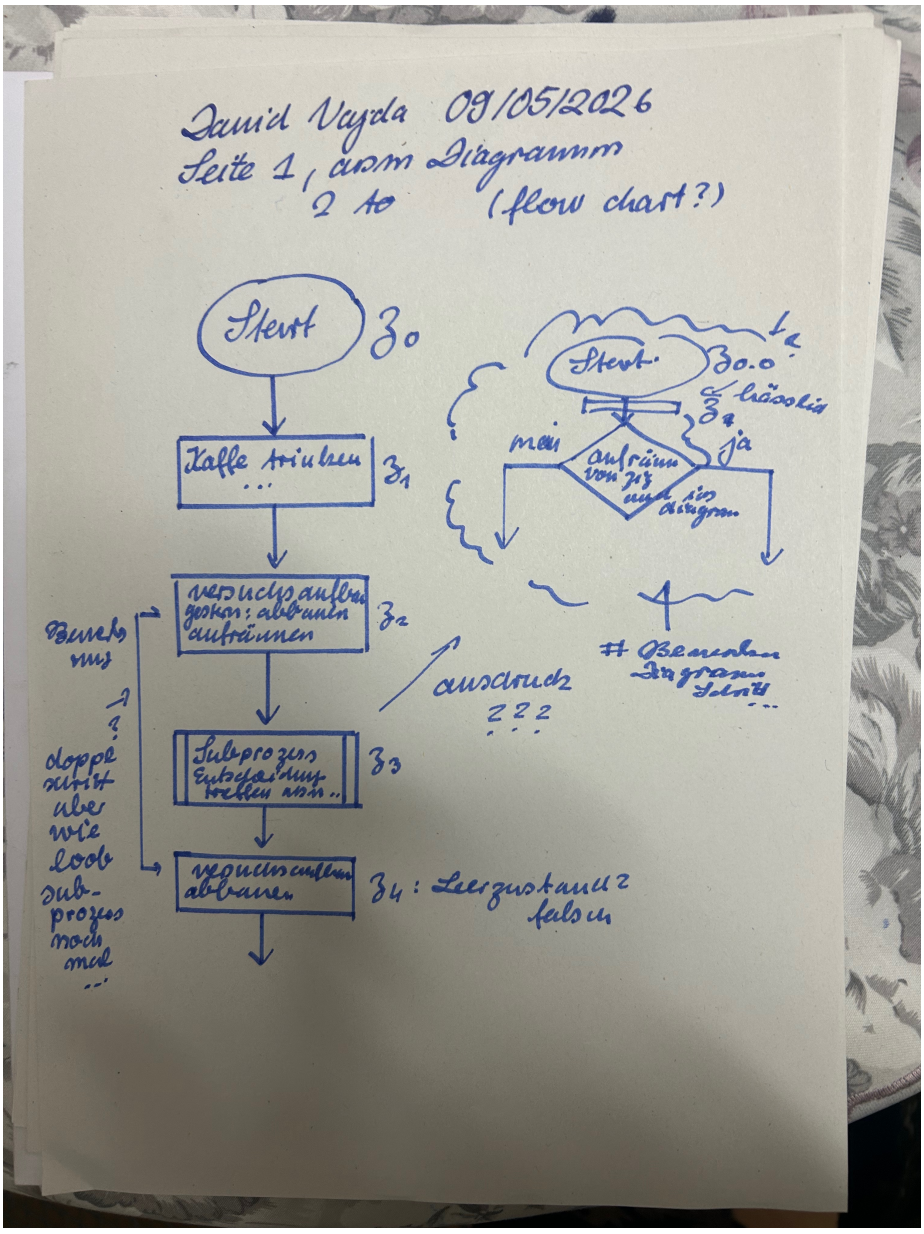

E-CALL
Va
Vb


E-CALL
Va
Vb

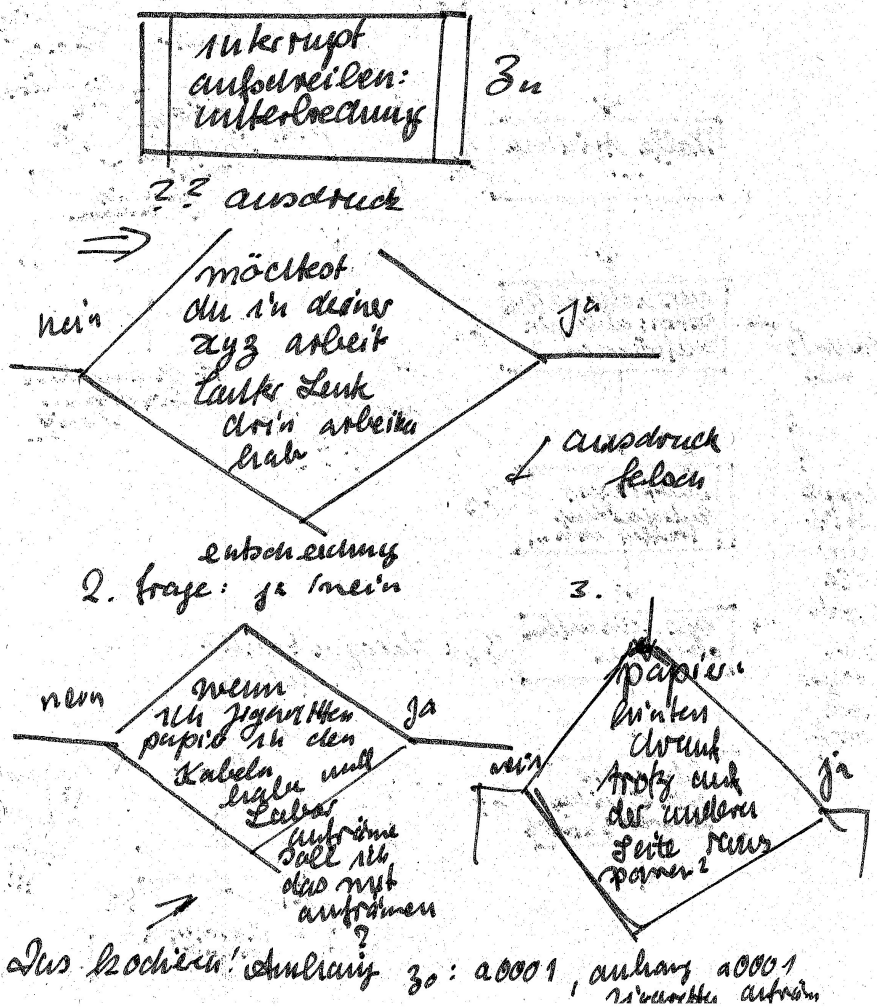

Va
Vb


E-CALI
EIC-404
Va
Vb


E-CALL
EIC-404
Va
Vb


David Vayda 09/05/2026
Seite 1, arm Diagramm
2 to (flow chart?)
Start
Kaffe trinken
...
Subprozess
Entscheidung
treffen usw..
Leerzustand
feiern
ja
nein
Start
Ende
ausdruck
z z z
...

Daniel Vajda, Seite 2
08/05/2026
asm diagramm flow chart
mir fällt auf:



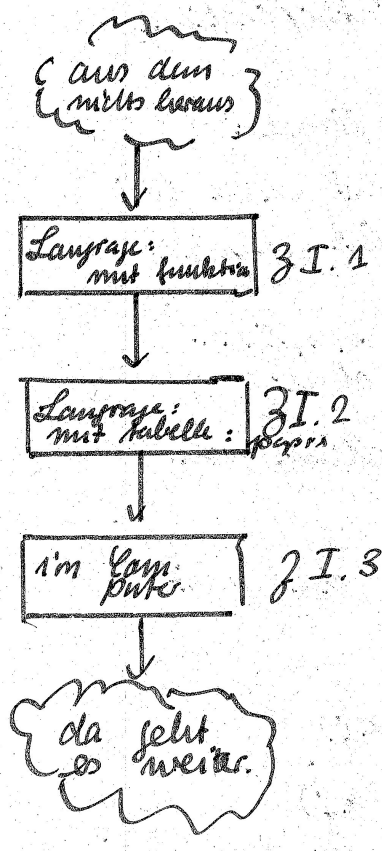
Das brochiert Anfrage zu: a0001, anfrage a0001 ...

David Vcyda (c) 09/05/2026

Seite 4, vom Diagramm...

(flow chart)...

3. Schritte:



Zigarettenpause...

Jamil Vajda (c) 09/05/2026
Seite 5, asm Diagramm...
(flow chart) +
Langrage, Interpolation...

$$l_i(x) = \sum_{\substack{j=0 \\ i \neq j}}^{n} \frac{(x - x_j)}{(x_i - x_j)} =$$

?

$x = (1, 2, 3, 4, 5)$
↑ x?

$$l_1(x) = \frac{(x-2)}{(1-2)} \cdot \frac{(x-3)}{(1-3)} \cdot \frac{(x-4)}{(1-4)} \cdot \frac{(x-5)}{(1-5)}$$

$$= \frac{(x-2)\cdot(x-3)(x-4)(x-5)}{(-1)\cdot(-2)\cdot(-3)\cdot(-4)}$$

$$= \frac{(x-2)\cdot(x-3)\cdot(x-4)\cdot(x-5)}{24}$$

Das andere mit tabelle lösen:
polyname:

Jamid Vajda (c) 09/05/2026

Serie 6, Langrage

$p_2(x) = (x-2)$

$p_3(x) = (x-3)$

$p_4(x) = (x-4)$

$p_5(x) = (x-5)$

$x^1 \ldots x^0$ (seltsame formulierung?



$a_1 x^1 + a_0 x^0$ ↑ identet kein

oder: → Chaos



↳ Reihenfolge im Denken...

← trotzdem !!!

Steinzeit



Hiero Glyphen normiert durcheinander

$p_1(x) = a_{11} x^1 + a_{10} x^0$

$p_2(x) = a_{21} x^1 + a_{20} x^0$

$p_3(x) = a_{31} x^1 + a_{30} x^0$

$p_4(x) = a_{41} x^1 + a_{40} x^0$

oder:

$a_1 x^1 + a_0 \quad | a_1 x_1^1 + a_0 x^0$?

$b_1 x^1 + b_0$

$c_1 x^1 + c_0 \quad \ldots$

$d_1 x^1 + d_0$

tabelle $(1, a_{11}) \quad (0, a_{10})$

$(1, a_{21}) \quad \ldots$

$(1, a_{20}$

prinzip...

David Vajda (c) 09/05/2026
Seite 3, asm Diagramm
+ weiteres flow chart...

im nächsten Schritt kommt:
Langrage interpolation.

freistil ↳ richtiger Name?

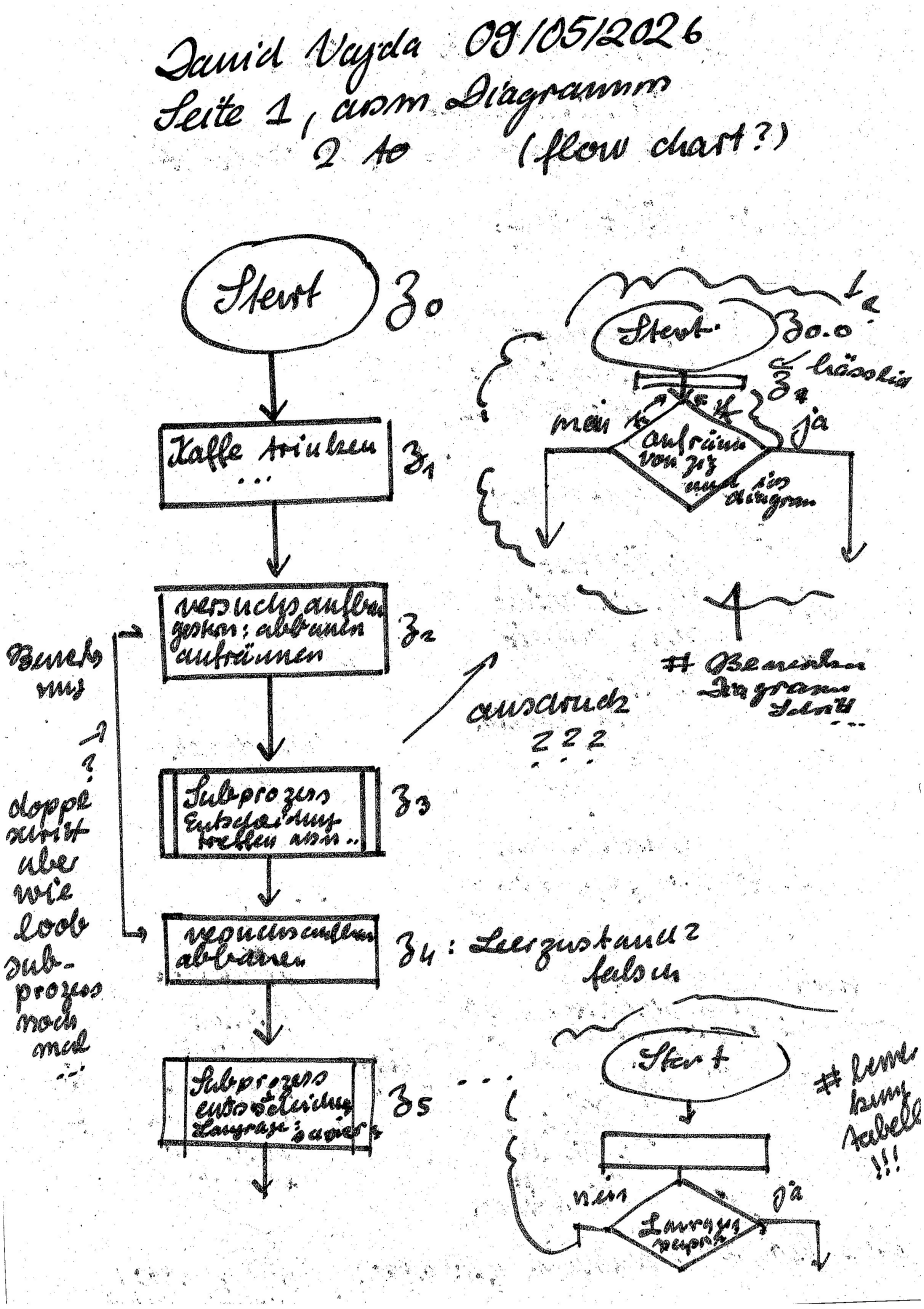
1.) Zum Glück muß man bei
x-en en keine y-ne
erraten...
$x = (1, 2, 3, 4, 5)$...
Das geht auch ohne:

1.1.) freie Darstellung von
polynomen...
D.h.
wir machen das mit
x:
Die formel:

$$\frac{(x - x_j)}{(x_i - x_j)}$$

Das Ding: kann man frei
hinschreiben:
ok: und mit 1....

1.2.) tabelle: keine funktion
wie heißt tabelle sondern tabelle: 1 funktion


David Vajda 09/05/2026
Seite 1, asm Diagramm
2 to (flow chart?)
Start
Z0
Kaffe trinken
Z1
Z2
Subprozess Entscheidung treffen usw..
Z3
Z4 : Leerzustand fehlen
Subprozess endbearbeitung Langsam speichern
Z5
Start
Z0.0
Z4
nein
ja
aufräumen von Z3 und Z4
Gemeinsam Diagramm Start
Tabelle
Langsam speichern
ja
nein

Daniel Vajda (c) 09/05/2026
Satz 8, Lagrange...

	Durcheinander
(1, -4)	
(1, -5)	

	(1,1)	(0,-4)
(1,1)	(1+1, 1·1)	(0+1, 1·(-4))
(0,-5)	(0+1, 1·(-5))	((0+0), (-4)·(-5))

	(1,1)	(0,-4)
(1,1)	(2,1)	(1,-4)
(0,-5)	(1,-5)	(0,-20)

$p_3(x) \cdot p_4(x) = x^2 + x(-9) + x^0 \cdot (-20$

$= x^2 - 9x - 20$

Daniel Vajda (c) 09/05/2026

Seite 9, Lagrage...

$p_2(x) \cdot p_1(x) = x^2 - 5x - 6$

$p_4(x) \cdot p_3(x) = x^2 - 9x - 20$

nächste Tabelle:

	$(2,1)$	$(1,-5)$	$(0,-6)$
$(2,1)$	$(2+2,1\cdot 1)$	$(2+1,(-5)\cdot 1)$	$(2+0,(-6)\cdot 1)$
$(1,-9)$			
$(0,-20)$			

David Vejde (c) 09/05/2026

Seite 1... Langraye...

	(
$(1,0)$	
$(0,-2)$	

↗ Zum Beispiel falsch

	$(1,1)$	$(0,-3)$
$(1,1)$	$(1+1,1\cdot1)$	$(1+0,(-3)\cdot1)$
$(0,-2)$	$(1+0,(-2)\cdot1)$	$(0+0,(-3)\cdot(-2))$

	$(1,1)$	$(0,-3)$
$(1,1)$	$(2,1)$	$(1,-3)$
$(0,-2)$	$(1,-2)$	$(0,-6)$

$p_2(x)\cdot p_3(x) = 1\cdot x^2 + (-2)\cdot x^1 + (-3)\cdot x^1 + (-6)\cdot x^0$

$x^2 + (-5)\cdot x + (-6)$

$= x^2 - 5x - 6$

Probe: Das müßte stimmen...

10.6 next

Dawid Vajda (c) 09/05/2026
Seite 9, Lagrage...

$p_2(x) \cdot p_1(x) = x^2 - 5x - 6$

$p_4(x) \cdot p_3(x) = x^2 - 9x - 20$

nächste tabelle:

	$(2,1)$	$(1,-5)$	$(0,-6)$
$(2,1)$	$(2+2,1\cdot 1)$	$(2+1,(-5)\cdot 1)$	$(2+0,(-6)\cdot 1)$
$(1,-9)$	$(2+1,-9)$	$(1+1,-14)$	$(1+0,-15)$
$(0,-20)$	$(2+0,-19)$	$(1+0,-25)$	$(0,-26)$

	$(2,1)$	$(1,-5)$	$(0,-6)$
$(2,1)$	$(4,1)$	$(3,-5)$	$(2,-6)$
$(1,-9)$	$(3,-9)$	$(2,-14)$	$(1,-15)$
$(0,-26)$	$(2,-19)$	$(1,-25)$	$(0,-26)$

$x^4 - 14 \cdot x^3 + (-19 - 14 - 6) \cdot x^2 + (-15 - 25) \cdot x + (-26)$

$x^4 - 14x^3 - 39x^2 - 40x - 26$

jetzt stimmt dieses Polynom?

David Vajda (c) 09/05/2026
Seite 10, Langrage...

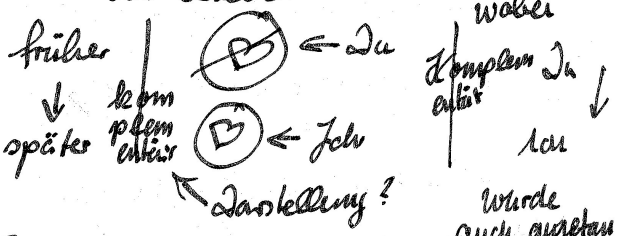
$$l_i(x) = \frac{x^4 - 14x^3 - 39x^2 - 40x - 26}{24}$$

$$= \frac{1}{24}x^4 - \frac{14}{24}x^3 - \frac{39}{24}x^2 - \frac{40}{24}x - \frac{26}{24}$$

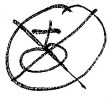
ok: binomialkoeffizient

1.) Darstellung tabelle
≠
tabelle = funktion (siehe Haskitabelle)

↑ vermutung - anderen zu Liebe!
aber genug der Liebe! und anderen



Echt, pech!
So ist es halt!
gute besserung ohne Heilung!

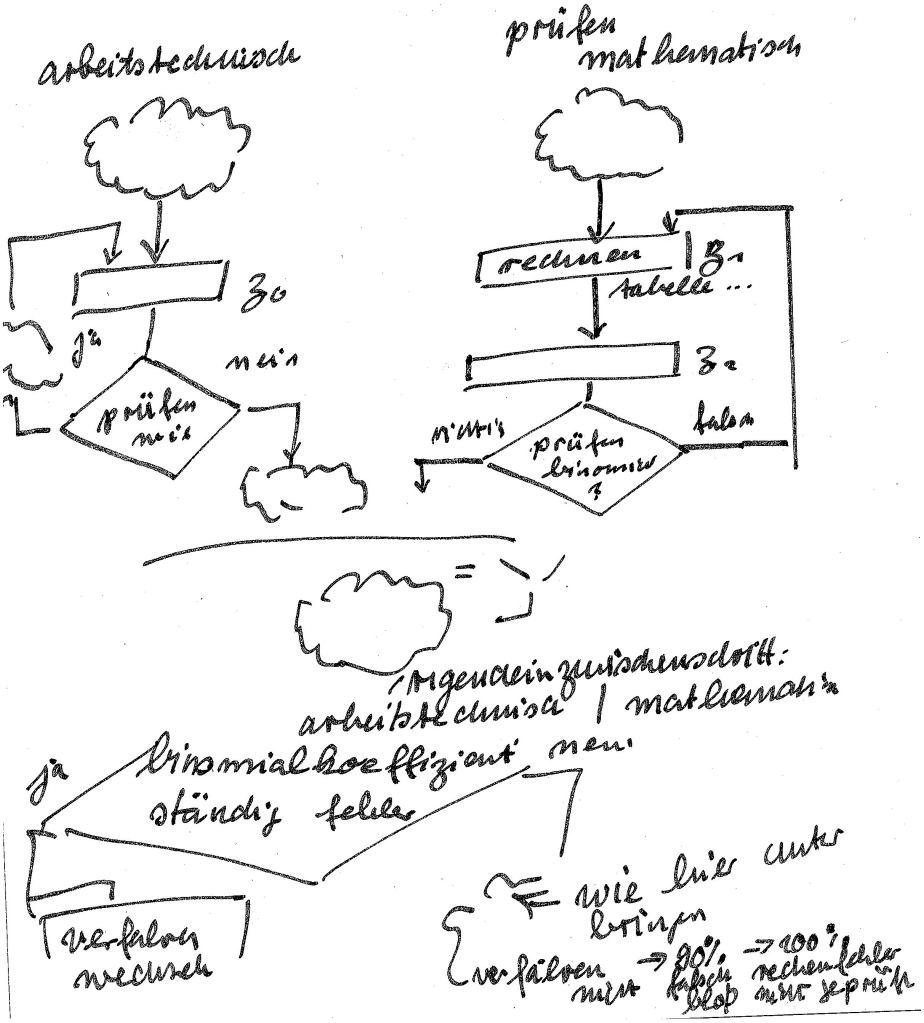


David Vajda (c) 09/05/2026
Seite 12, Langfassung



Daniel Vajda (c) 09/05/2026
Seite 11, Langrage...

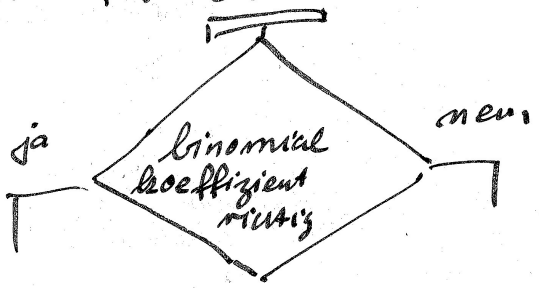
Binomialkoeffizient: prüfung



Daniel Vajda (c) 09/05/2026
Seite 12, Zaungraße ..

binomialkoeffizient?
bei polynom multiplikation

wieder frage:



wahrscheinlich:
nein!

- $(x+b)^m$ z.b. enthält einen
- wie sieht der aus
- wie lautet das verfahren?

wahrscheinlich

√ polynommultiplikation
√ Division

osw nachgucken... jetzt aber
pause
Dann: ttb und nötigen falls!
falsch: mit y - Dann Integral

Dawid Vajda (c) 09/05/2026
Seite 13, Lagrange
Interpolation...

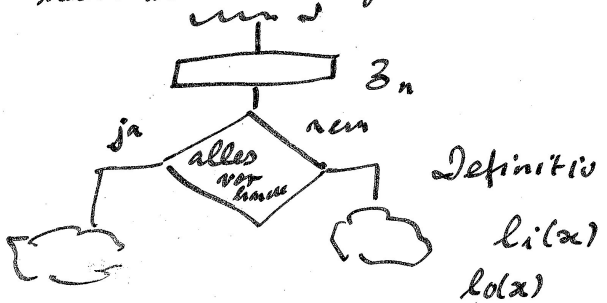
Jetzt aus dem
$l_i(x)$
...
ein polynom mit y-Werten
machen...
Diese waren $y = (2, 8, 9, 5, 9)$
ok...

in ...

$$P(x) = \sum_{i=0}^{n} k_i l_i(x)$$

↑
wobei

halt! haben was vergessen!



Definition

$l_i(x)$

$l_0(x)$
$l_1(x)$
$l_2(x)$
⋮

$\mathcal{L} = (l_0(x), l_1(x), l_2(x), \ldots)$

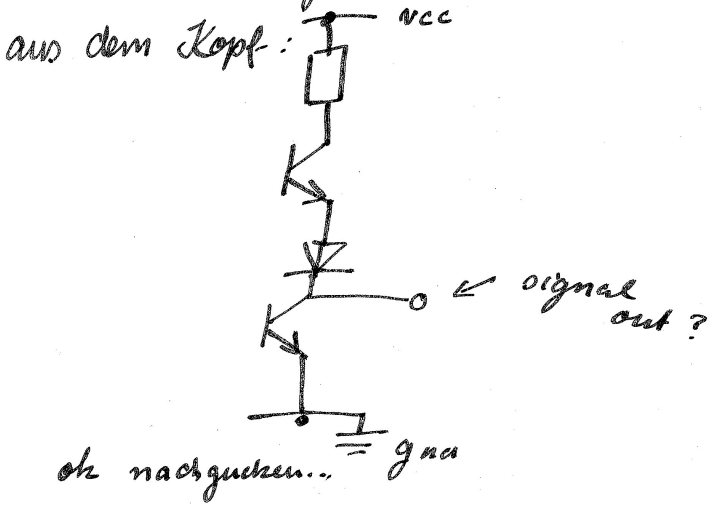
Dawid Vejda (c) 09/05/2026
polynominterpolation, Seite 15

$l_1(x) =$
$l_2(x) =$
$l_3(x) =$
$l_4(x) =$

pause! nur f_i ist nachher y...

oh: jetzt lieber: ATL - zur selbst erhaltung!

aus dem Kopf:



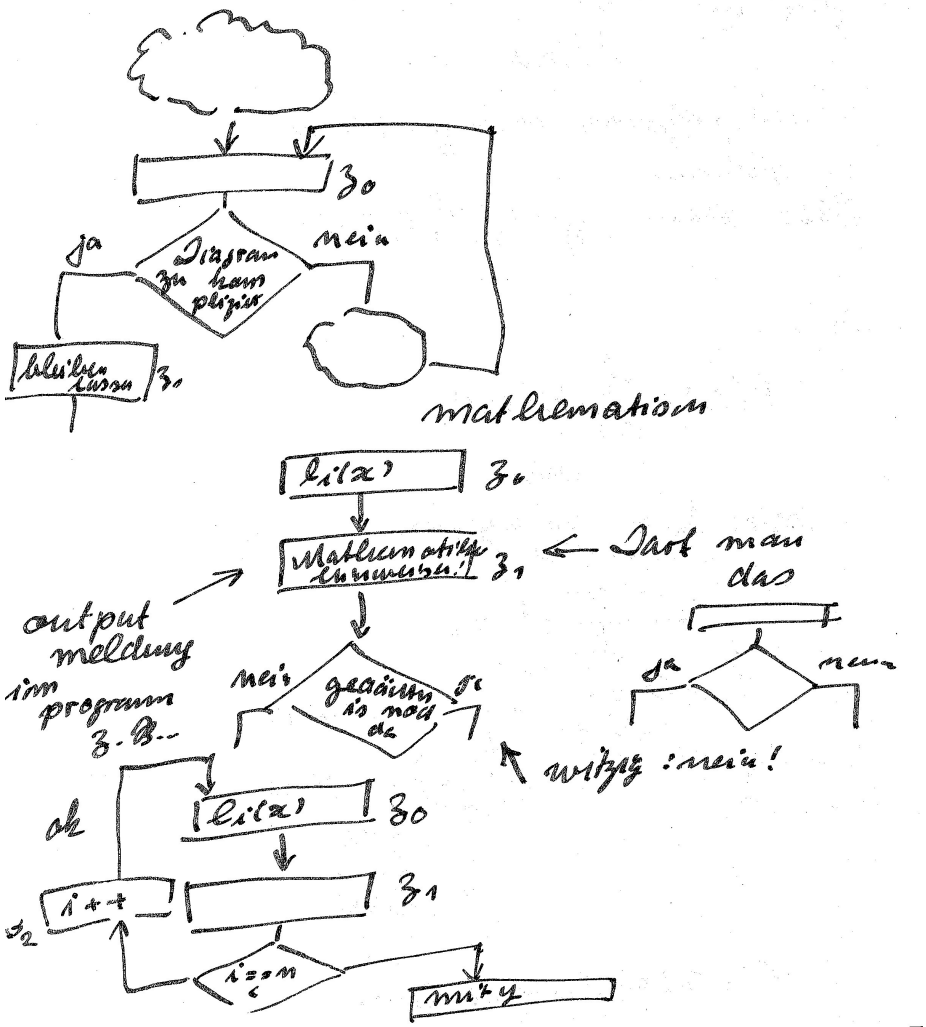
oh nachgucken...

David Vejda (c) 09105/2026

polynominterpolation...

Langrage...

Seite 14



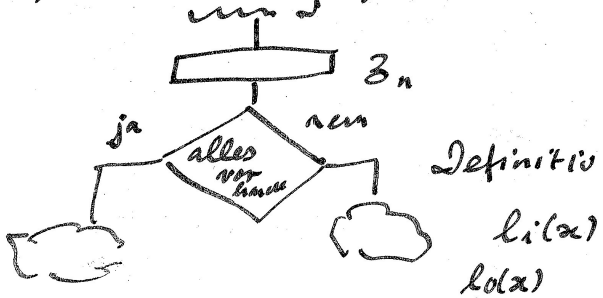
Zuvid Vajda (c) 09/05/2026
Seite 13, Langraye
Interpolation...

Jetzt aus dem
$l_i(x)$
...
ein polynom mit y-Werten
machen...
Diese waren $y = (2, 8, 9, 5, 9)$
ok...

in ...

$$P(x) = \sum_{i=0}^{n} k_i l_i(x)$$
↑
wobei

halt! Koeffizienten was vergessen!



$$L = (l_0(x), l_1(x), l_2(x), \ldots)$$

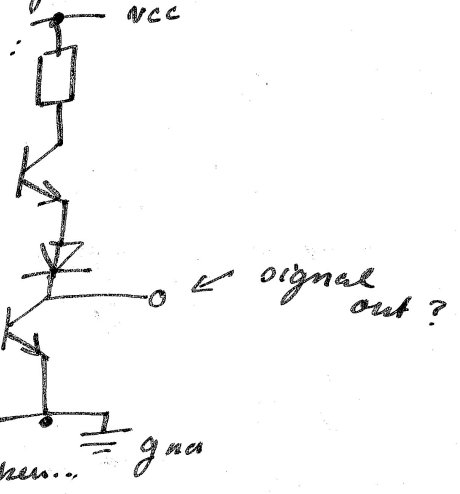
Jamid Vajda (c) 09/05/2026
polynominterpolation, Seite 15

$l_1(x) =$
$l_2(x) =$
$l_3(x) =$
$l_4(x) =$

pause! nur f_i ist nachher y...

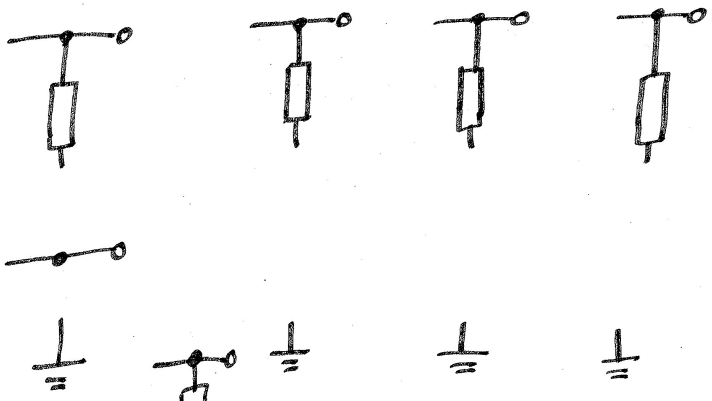
oh: jetzt lieber: ATC - zur selbst erhaltung!

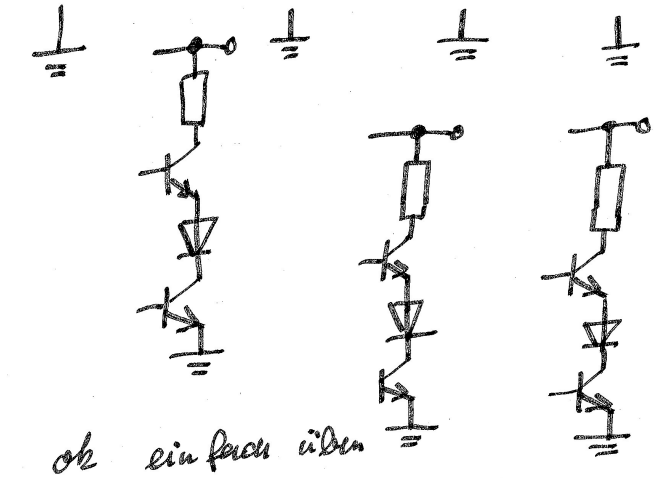
aus dem Kopf:



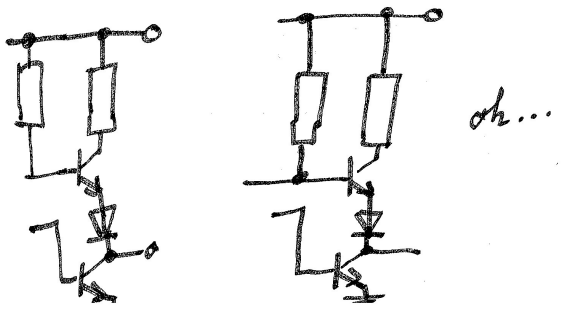
oh nachgucken...

David Vajda (c) 09105l2026
Seite 17 / ATL...



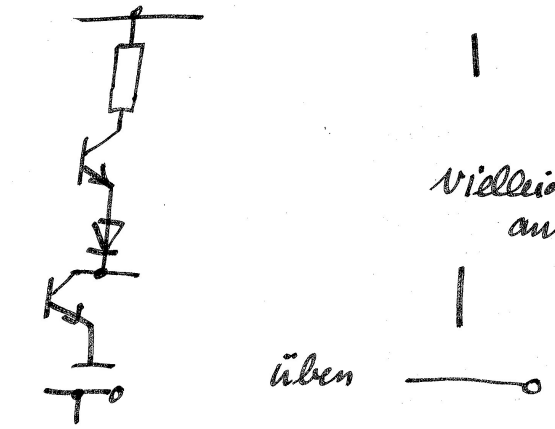


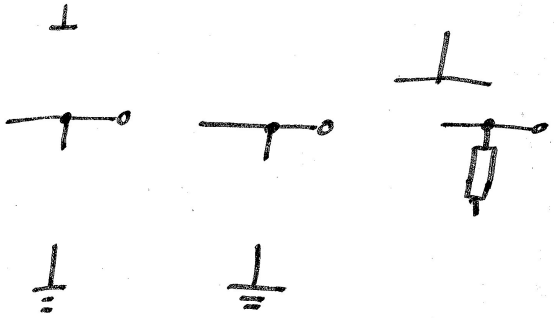
oh ein fetzen üben



oh...

Dunid Vojda (c) 09/05/2026
Seite 16 polynom interpolation...
Atl...

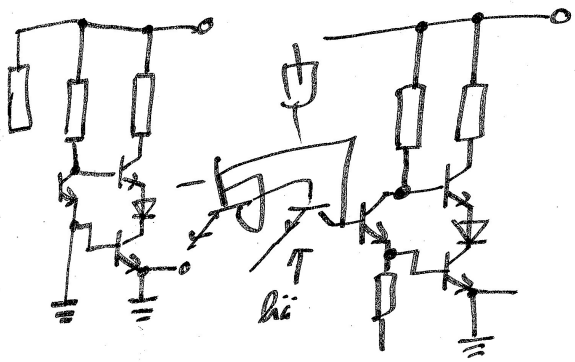




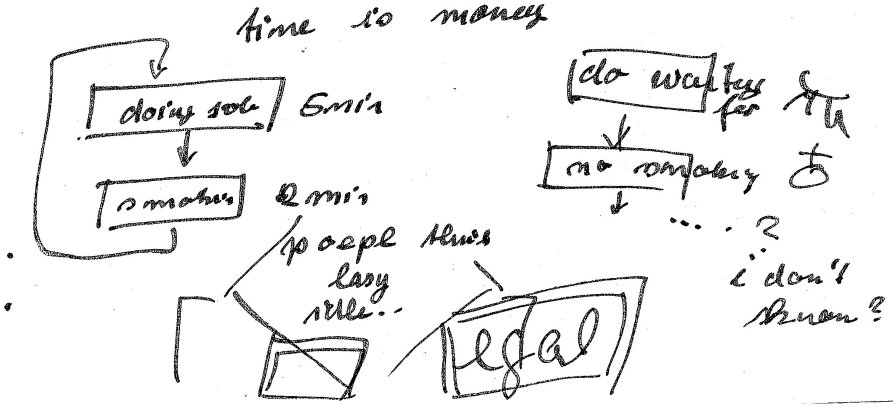


↑ Hinweis : echo("...: wie ausdrücken")
print echo ... Hinweis!

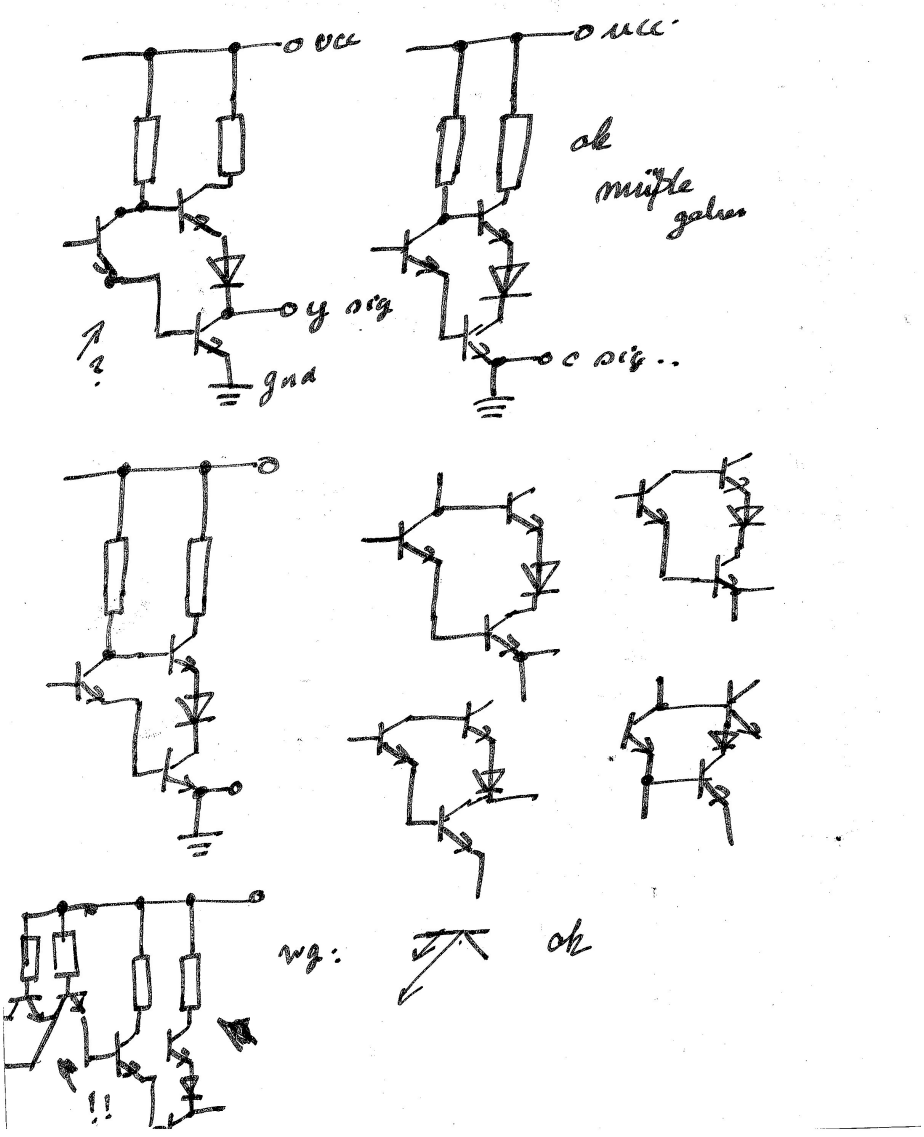
David Vajda (c) 09/05/2026
Seite 21, ttl gatte...



oh...
↑ industrial std.
from Bach is the
days it turk?
time is money!
oh: nachdem die Sache so schön
tut teilen wir das zeitfenster ein.

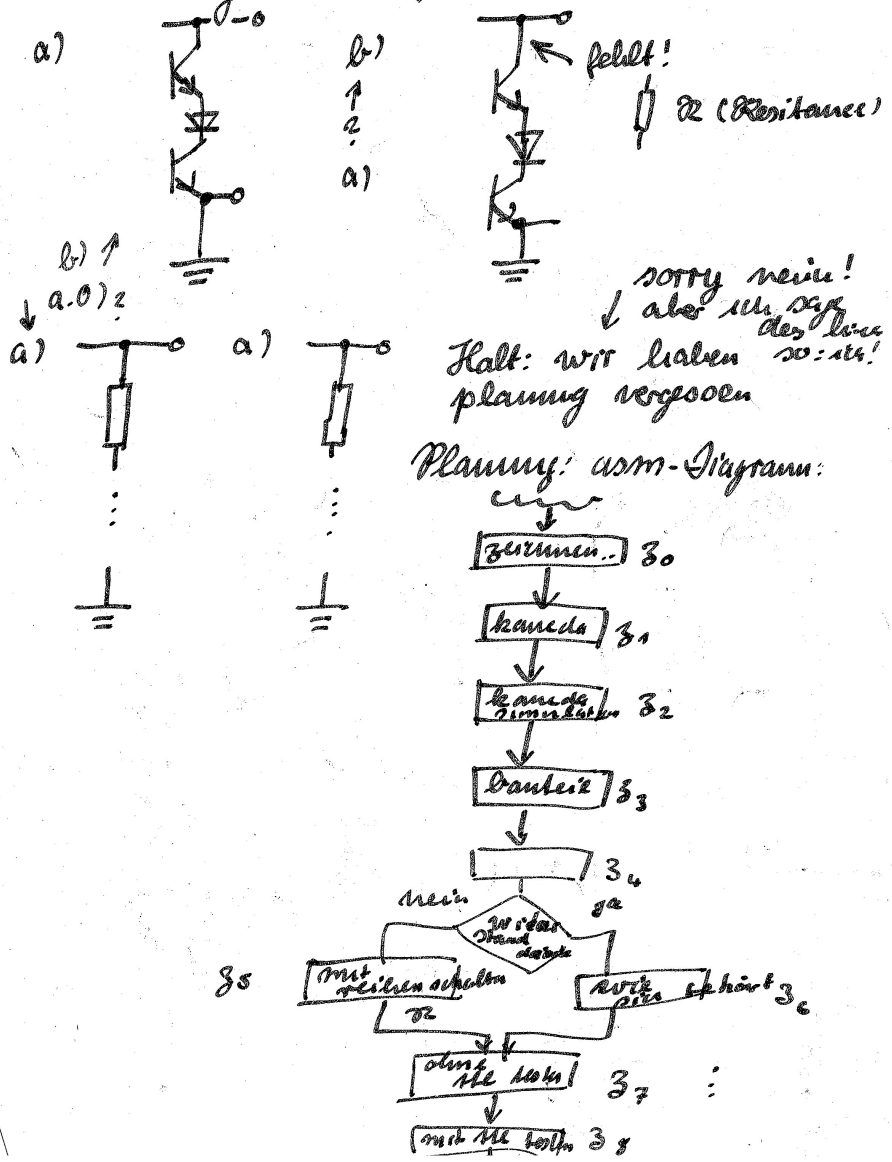


Jamal Vajda (c) 09/05/2026
Seite 20, tHl gatter...

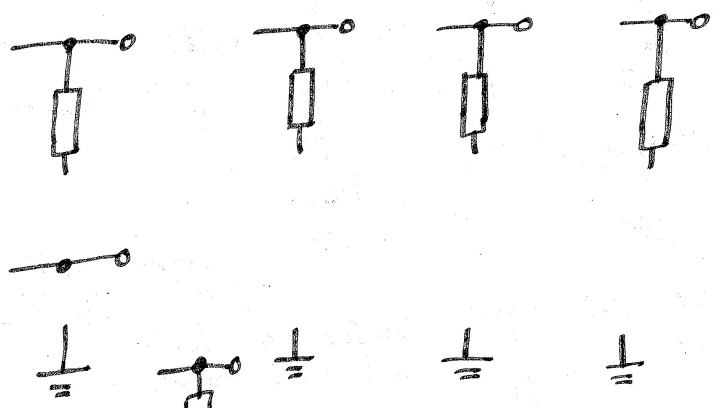


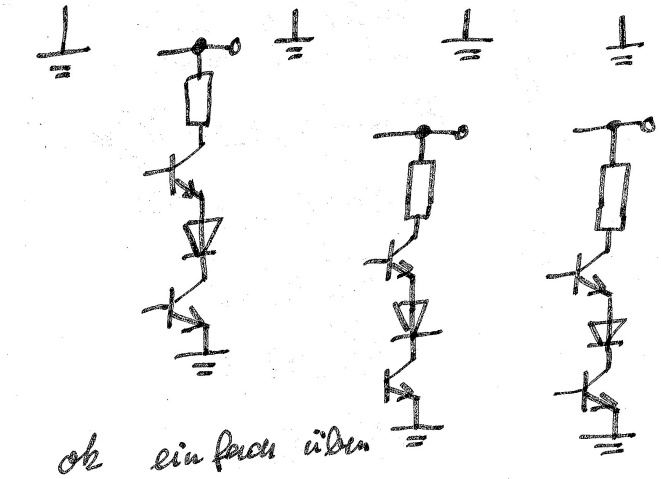
Daniel Vagda (c) 09/05/2026

Hl gatter übung, Seite 18

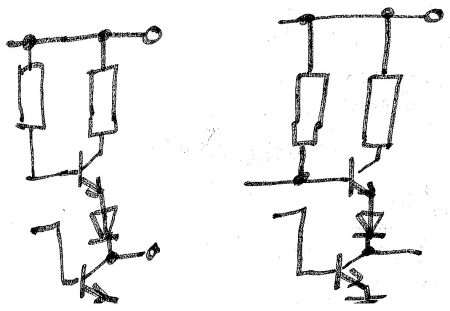


David Vajda (c) 09105l2026
Seite 17 / Alle...



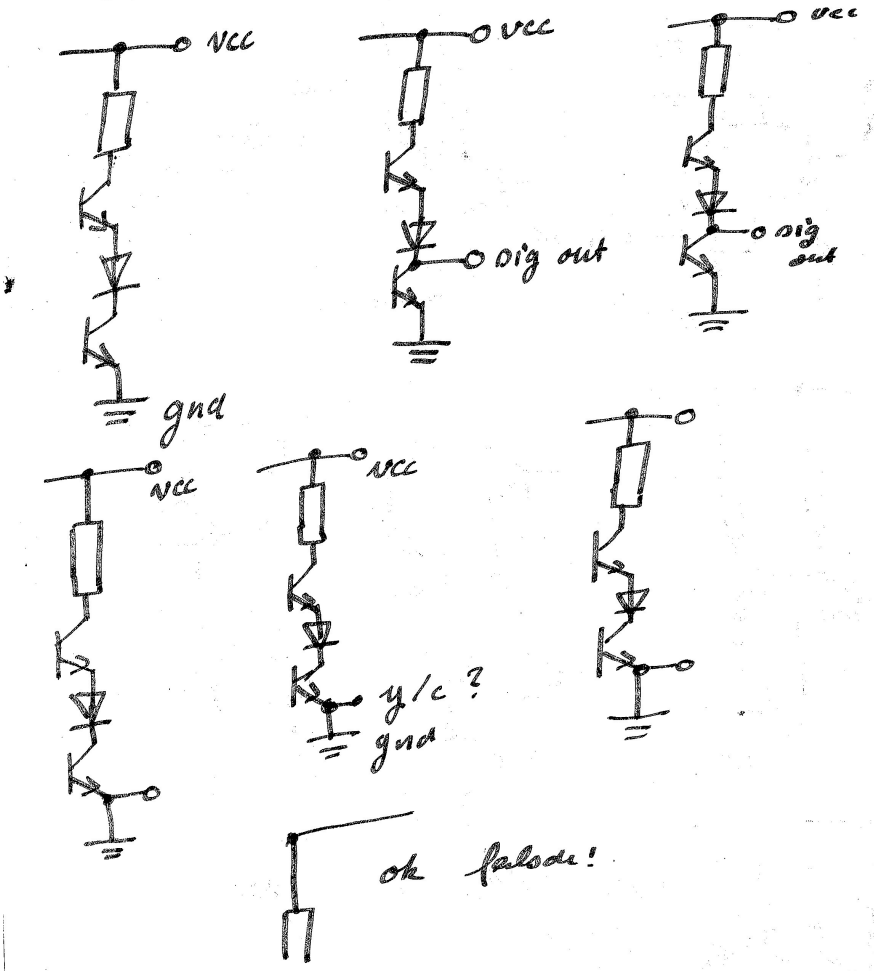


oh ein paar üben

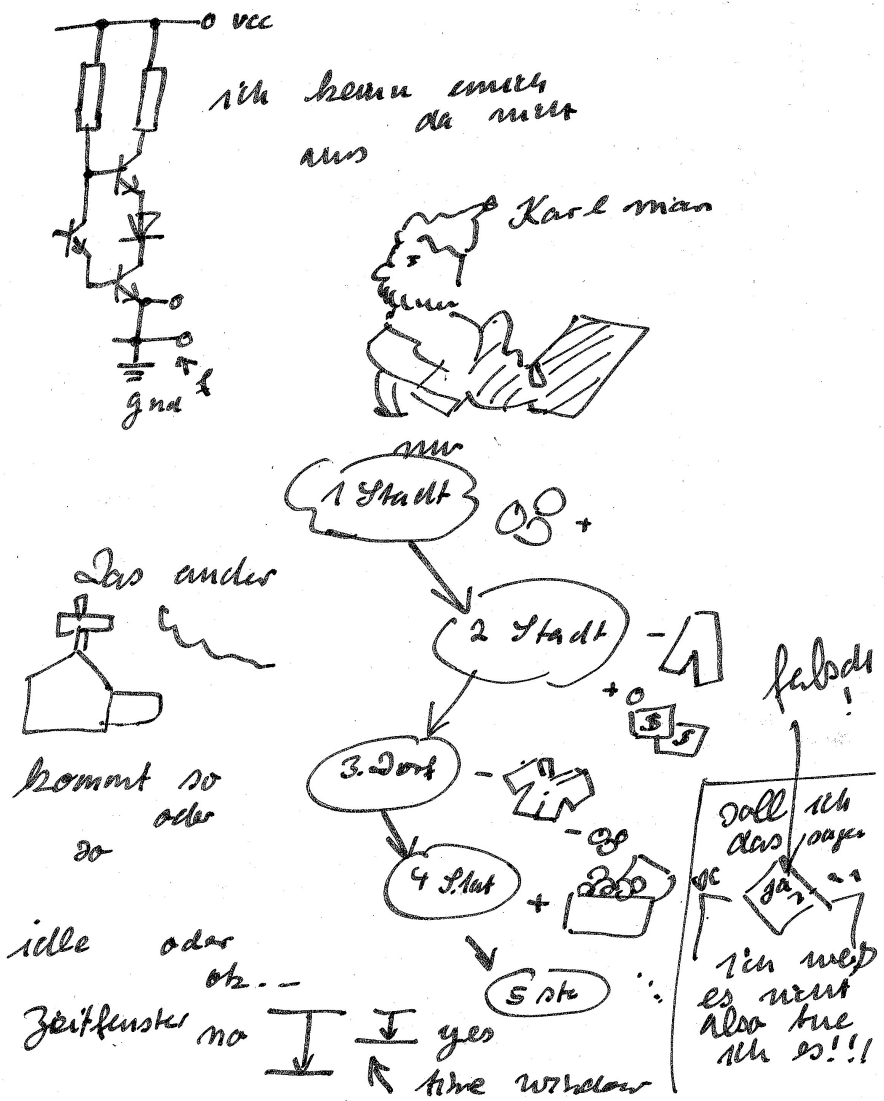


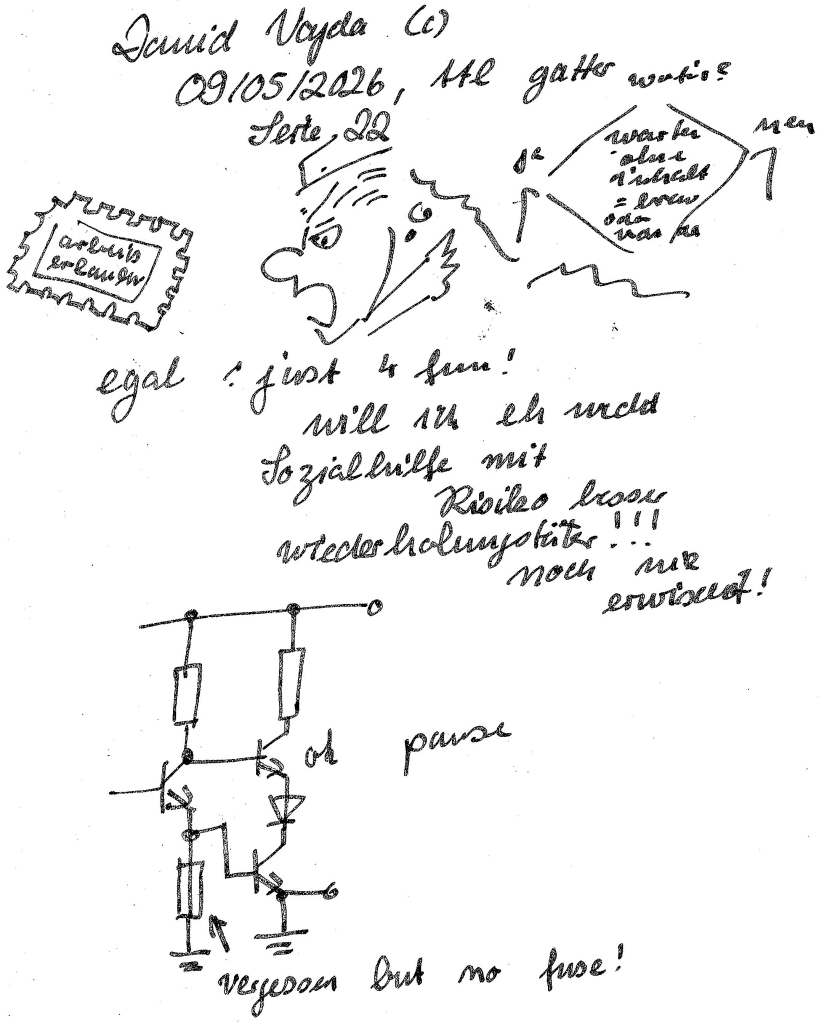
oh...

Daniel Vajda (c) 09/05/2026
ATL gatter... Seite 19..
ok: gegentaktstufe üben: endstufe...

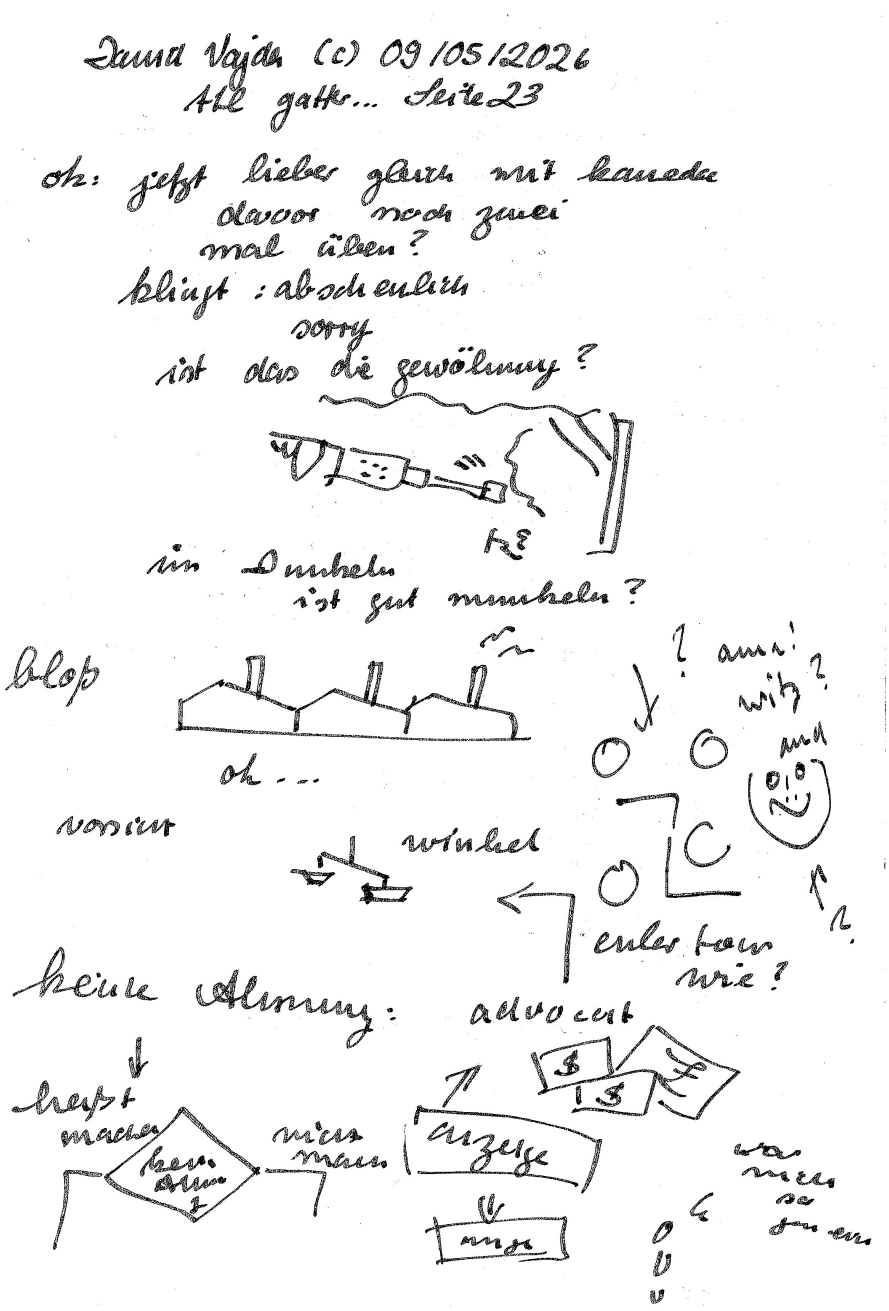


David Vagda (c) 09/05/2026
Atl gatter übung.. Seite 21

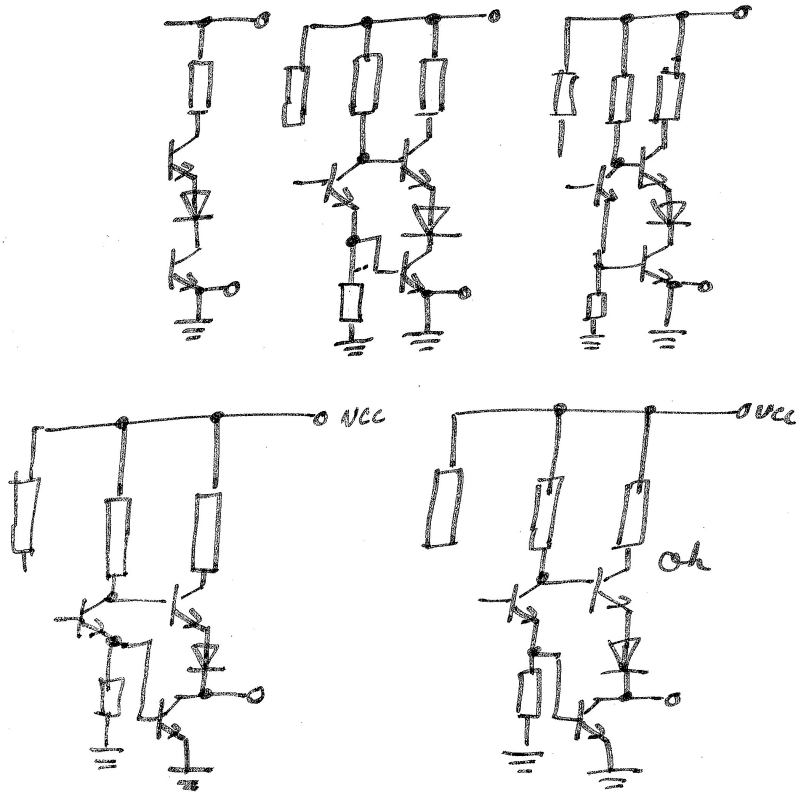



Dennid Voyda (c)
09/05/2026, 11h gatte wurde?
Seite 22
arbeits erlaubnis
egal : just 4 fun!
will ich ehrlich
Sozialhilfe mit
Risiko bross
wiederholungstäter!!!
noch nix
erwischt!
oh pause
vergessen but no fuse!

Dana Vajda (c) 09/05/2026
All gatte... Seite 23



Damit Najda (J) 09/05/2026
Atl galtes... Seite 24



pause machher weiter...

David Vayda (c) 09/05/2026
Atl gatter, ..., Seite 24

okz: vorgehen:

1.) ich schreibe es per Hand auf...
noch mal: (a) unsere Gatter: (b) unsere: asm Chart diagramm... floatingt

2.) also:
a) ich übe das Atl gatter per hand: transistor-transisten Logik
b) ich übe es genauso in caneda: einfach als übung
c) caneda hat glaube ich:
+++ illustrierte und die Weisheit übe gatter illustrierte ≙ Klugscheißerei interessieren mich nur anderswoanders also sonst +++ abtria und contemplativa?

(20) (50) ich sage mal so!

+++ versichert Doppelagent +++
Liebe Mädchen: Reden ist nicht alles fühlt mit zu unwissenheit
⇒ Das hier ist meines: Wir ein

David Vejda (c) 09/05/2026
ALL gatter... Seite 25

e Dritte: diese Leute sind im
ALL gatter Schleifen nicht ich!
und wenn



Dann: ich bin auch ein Mann!
und:



Hin oder Her

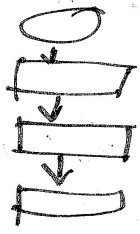


Uhr einstellbar

Ying und Yang
Tag und Nacht

Also vorsicht!
und jetzt weiter:
ordentliche übung!

ich bin ich!
1.) mit hand
2.) caneda
3.) caneda simulation



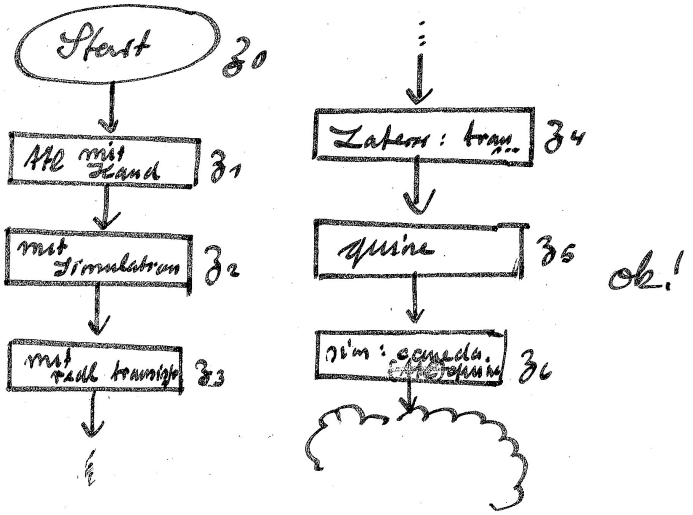
... gehen ...

Daniel Vajda (c) 09/05/2026

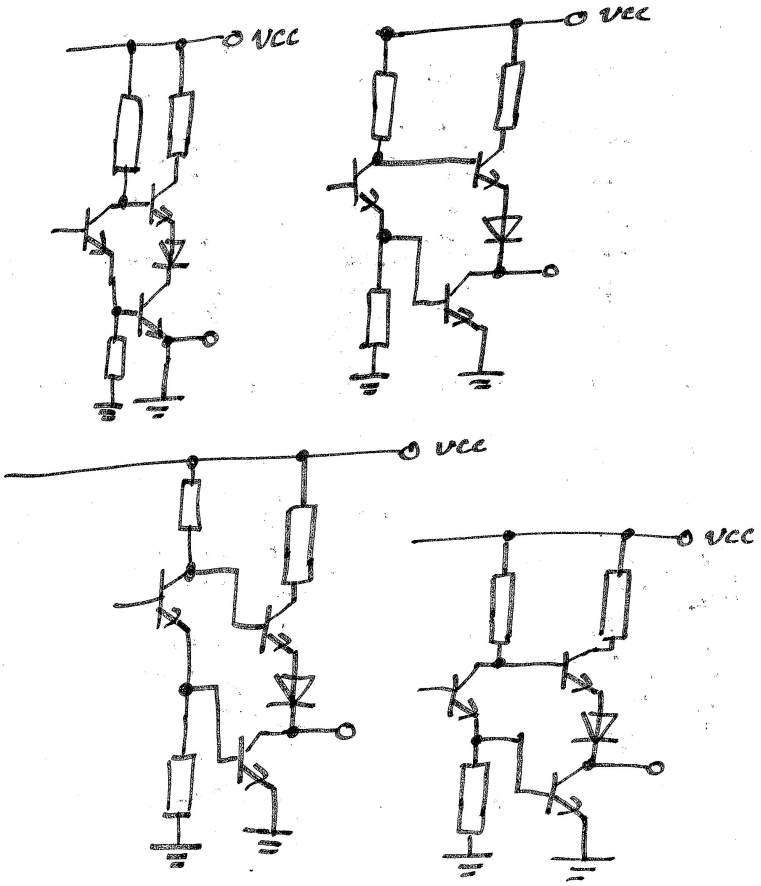
ttl gatter: Seite 27

So also:

1.) mit Hand
2.) mit caneda
3.) mit caneda als Simulation
4.) dann mit realen Transistoren
5.) ok: wenn das geschehen ist: noch übung mit Latch: transistor...
6.) Dann kommt: duf und zwar mit quine und allem was dazu gehört
7.) ttl simulieren in caneda...

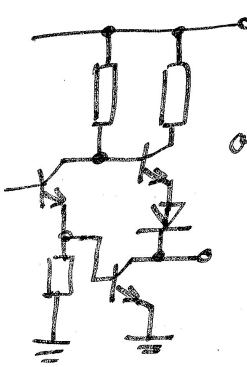


Jamid Vajda (c)
09/05/2026, Seite 28
Atl gatter übung...



ok: hier war die übung...

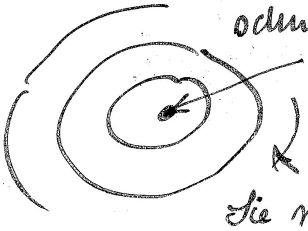
Daniel 09/05/2026
Vajda Attila gatte... Seite 29



ok das war noch eine
jetzt eine pa...

Was ist die Steuererklärung
daje?

1.) es ist keine Steuererklärung
wenn man lügt!
2.) ich habe mir sagen lassen
daß jeder seine eigene
Steuererklärung macht!
3.) wie sah die erste Steuererklärung
der Welt denn aus
4.) Konzentrische Kreise:
Entschuldigung ich
schwärme... ??



Das ist der
Teil der macht
seine ... ja
De und viele weitere
Sie müssen... also.

David Vajda (c)
09/05/2026 11€+
Steuerklärung... Seite 30

a) pause:
weil ich enttäusche?
⇒ Das muß man!
weil man langsam aber
stetig kann

5.) Die einen:
können die anderen
müßen aber die die
können müßen auch wenn
es wieder so weit ist



6.) Warum ist das Steuerrelevant?
a) weil ich lerne aber produziere...
b) aber wie: verkaufe ich es?
nein: somit nicht illegal?
nein: Handel ist illegal.
ich produziere wissen vorraus
7.) ich sage aber mal so:
Es gibt so und so viele eifersüchtige
Watuten auf dieser Erde:
Rumänien + Bulgarien + Ukraine +
überall +...

10.7 next

David Vayda (c) 09/05/2026
Atl + Steuererklärung,
Seite 32

10.) Die Sache braucht ihre Zeit:
ich weiß alles...
nein! ich lerne technik...
ich werde wissen wie
es ausgeht?
nein!

11.) Sage aber Wahrheit:
Liebes: Steueramt
+ Liebes Sozialamt
+ Liebes Arbeitsministerium:
vielleicht (a) haben sie
mich verletzt
√ich bin in der
Lage: das zu tun
was ich tue: aber nicht 10.000
Masken auf zu stellen...

12.) aber wichtiges kann abhanden:
√ Deutschland kommen!
ich finde seltsam
√ gefahr: Sicherheitsrisiko

13.) ich Abilde in Deutschland!
Gefahr größer: Leute schleppen
Wissen weg!

Zumi Vajda (c) 09/05/2026
Alle gatter... Seite 33

Gucken Sie mal:
Vater tot?

14.) Ich sage meiner war nett.
und andere?
andere sitter und
aber meiner schon!

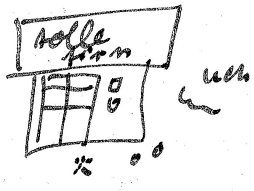
also ich passe auf!
irgendwann trennen
sich Vater und Sohn!

weil
a) jemand fragt: will Sohn
Vater los werden, aber
niemand: Vater: Sohn
b) Es gibt so und so viele
Leute. Die sagen:
Sie leben in der Welt..
neu..
andere wissen davon:
aber: Man sieht sich
nicht
c) für mich ist er tot:
Das heißt: ich will mein
geld
d) weder so noch so:
Das Geld teilweise aus
Deutschland

Dawid Vayda (c) 09/05/2026
Atl gatter, Seite 31

Die machen aggressiv!
und können?
Nein: Du glaubst! Das falsch!

Nächster Punkt...
8.) als ich meiner Mutter Zigaretten
geld entwendete als kleiner
Junge: Da hätte ich sagen können:
Hallo ich rauche - 5 DM!
Aber ich tat es nicht ⇒ folg
kriminell
9.) Guck mal dritte!
ich bleibe in Sozialhilfe
weil ich nichts besseres vor
habe!
überlege mal Arbeitserlaubnis:
Dann tolle firma:



und da?

Daniel Vayda 09/05/2026
Alle gatte... Seite 34

d) geld aus Deutschland
"Steuer in Deutschland"?
noch besser:
geld wegschleppen
e) ich will aber niemanden-
allein glücklich...
gehe Deutschen family nicht
auf den...
f) Noch besser: Vater tot:
ja / nein..
ich mag vergiften mich:
bin in Deutschland..
bleib: es war nett!
Das ist der unterschied
todesurteil unterzeichne ich nicht!
ich sage was ist?
egal: für mich
g) lebt noch? Dann sage ich:
aber auch nicht ewig!
h) am Ende alles ein Erlebnis!
und dann?
15.) man muß das Erbe annehmen-
wenn man es tut und das ist
besser: Warum: Weil man weiß
nicht was passiert. Denkt: Angst!

Jamal Vajar (c)
09/05/2026 Alt zalte...
Seite 35

wenn man sein Erbe nicht
annimmt:

Dann passiert folgendes, möglich
1.) Deutschland ist
geprellt worden...
2.) Es können Sachen passieren,
die wir nicht kennen..

vor längerer Zeit lief vor mir ein
Mann und jammerte: Er sei um
40.000 € o. 80.000 € geprellt ~~be~~
worden
1.) Jugend/moral sorgen
haben die: Die es annehmen
2.) Der Rest finanzielle...
und wer weiß was noch alles
passiert?
1.) man denkt man kriegt
kein geld: nur 0,00€
2.) verblickt noch eines auf
die Waage...
ich weiß es nicht. Man erzählt den
Leuten von Jugenden: Schönen Leben:
ordnung. Wer es bevolgt: kriegt eins auf
die Nase...

Daniel Vajda (c) 09/10.5.2026
Hl Gatte ... Seite 36

Sie kriegen vielleicht...

Das ist das...

16.) Sie sagen: Das ist ein Fehler!
Ihre ganze Einstellung mit
Tugend!
Sie arbeiten aus Tugend
Sollte dieser Herr noch:
soll er sich selbst verteidigen!
aber für mich ist es so!

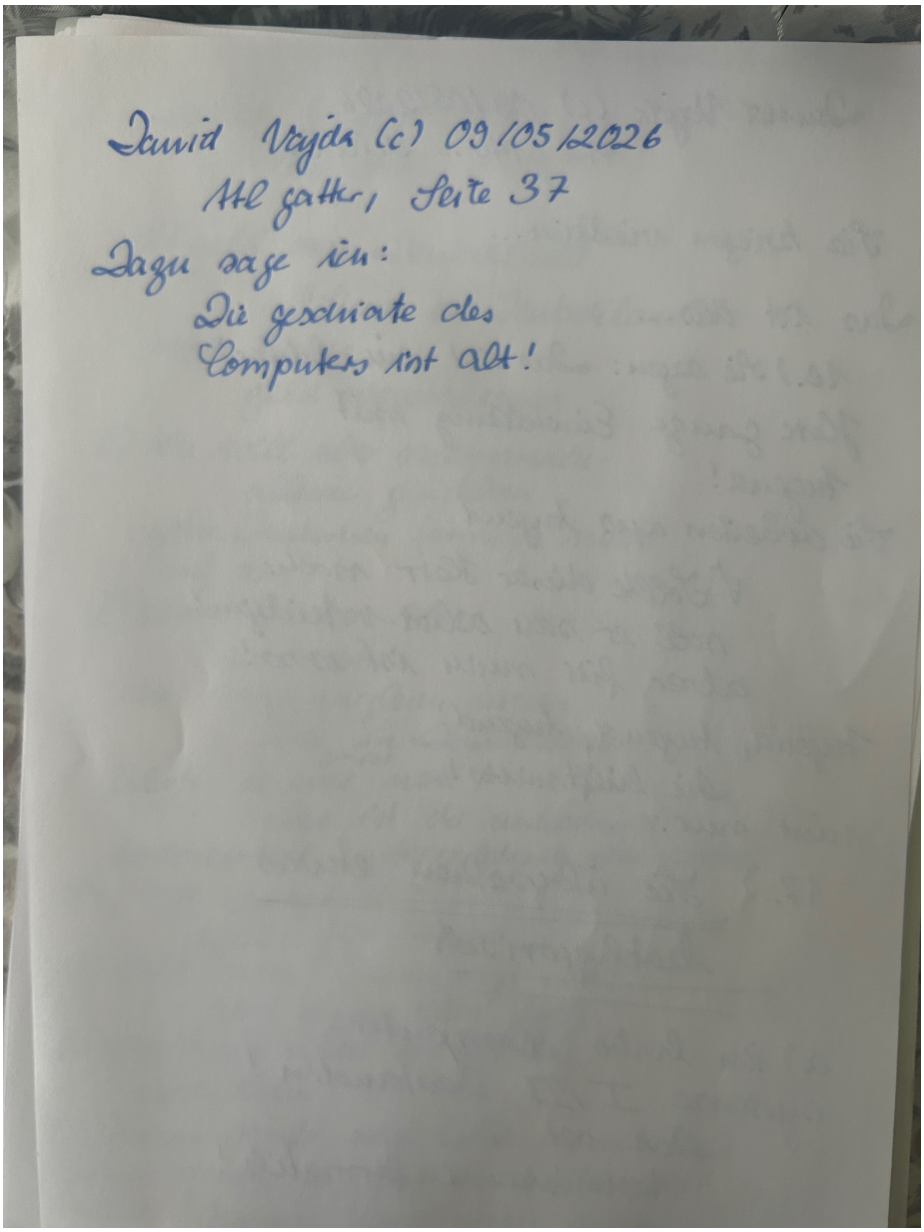
Tugend, Tugend, Tugend!
Die hilft nichts!
nicht nur:

17.) Sie übersehen etwas
kategorisch

a) ich habe Computer-
systeme I/II bestanden!
Das ist
technische Informatik!

Jamia Vajda 09/05/2026
Alle gatte... Seite 34

d) geld aus Deutschland
"Steuer in Deutschland"?
noch besser:
geld wegschleppen
e) ich will aber niemanden-
allein gleichlassen...
gehe Deutschen family nicht
auf den...
f) Noch besser: Vater tot:
ja / mein..
ich mag vergessen nicht:
bin in Deutschland..
bleib: es war nett!
Das ist der unterschied
todesurteil unterzeichne ich nicht!
ich sage was ist?
egal: für mich
g) lebt noch? Dann sage ich:
aber auch nicht ewig!
h) am Ende alles eine Bildung!
und dann?
15.) man muß das Erbe annehmen-
wenn man es tut und das ist
besser: Warum: Weil man weiß
nicht was passiert. Leicht: Angst!


David Vajda (c) 09/05/2026
Alt gatter, Seite 37
Dazu sage ich:
Die geschichte des
Computers ist alt!

Jamal Najar (c)
09/05/2026 Alt Zatte...
Seite 35

wenn man sein Erbe nicht annimmt:

Dann passiert folgendes, möglich

1.) Deutschland ist geprellt worden...

2.) Es können Sachen passieren, die vor nicht kenne..

vor längerer Zeit lief vor mir ein Mann und jammerte: Er sei um 40.000 € o. 80.000 € geprellt ~~to~~ worden

1.) Jugend / moral sorgen haben die: Die es annehmen

2.) Der Rest finanzielle...

und wer weiß was noch alles passiert?

1.) man denkt man kriegt kein geld: nur 0,00€

2.) verliert noch eines auf die Nase...

ich weiß es nicht. Man erzählt den Leuten von Jugenden: Schönen Leben: ordnung. Wer es befolgt: kriegt eins auf die Nase...

Daniel Vajda (c) 09/05/2026
Alt gatter, Seite 37

Dazu sage ich:
Die geschichte des
Computers ist alt!
Sie ist sehr alt!
und schon jetzt fürchte
ich man dabei sind teile
abhanden gekommen...
Also: Computer so haben kein
patentrezept...
a) mips 32 ist quelloffen
b) arm
c) atmel preisen produkte
online an ...
Sie sagen:
true was der Lehrer sagt
ich sage:
inhaltlich: ja...
aber a) Der Mensch ist faul
Lernmaßnahmen haben
oft Ziele...
b) Der Lehrer selber...
und so weiter: atmel ist selber
schuld. Und was soll das?

Jamil Vajda (c) 09/05/2026
11l gatte... Seite 38

ok: Dann geht die Sache weiter:
keine einzige aufgabe
wurde abgeschrieben!
ich liebe die selber entwickelt!
weiß aber worum es geht!

Latex
bash
sind quelloffen!
ok!
am Ende weiß keiner mehr was...
und ich sage!
Hoppla: technische
Informatik ist... irgendwas.
blop braucht traurig: und morgen
kaufe Computer ein!
blop wäre schön wenn alle
freuden und Sorgen heilen!
und entläuschungen!
mir ist das egal: blop!
technische Informatik: Labor ist
wohl aufwändiger als man meint!
18.) ich sage nicht andere sollen das
nicht! nur: ich meine prüfung
habe ich. ok. ich sage nicht

10.8 next

Jamil Vajda (c) 09/05/2026
Ath gatha ... Seite 39

dass andere mehr können oder
ich mehr sonst getan hätte.
Bloß jetzt bin ich sicher.'

18.) Wenn Irrtümer tun sich
vielleicht auf ...

a) wg. Sonne: am Ende
jedes glaubt der ist ein
Dieb? mein!

b) Computersysteme 1/2
es gibt viele Irrtümer am Ende
woanders:

jemand hört:

Denkt ach so ...
meine ich aber nicht.

(Uhw. ... legal
nur: Die Aufgaben selbst gemacht?
ja!
nächstes:

ich habe prüfung.
ob ich weiß es.
Ich will produzieren?
Dann sage ich
ob. weiß ...

Durch Vajda (c) 09/10/2026
Hl Gatte ... Seite 36

Sie kriegen vielleicht...

Das ist das...

16.) Sie sagen: Das ist ein fehler!
Ihre ganze Einstellung mit
Tugend!
Sie arbeiten aus Tugend
V Sollte dieser Herr noch:
soll es sich selbst verteidigen!
aber für mich ist es so!

Tugend, Tugend, Tugend!
Die hilft nichts!
nicht nur:

17.) Sie übersehen etwas
kategorisch

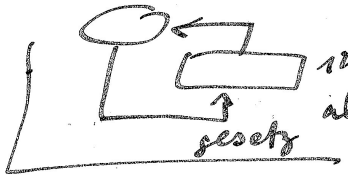
a) ich habe Computer-
systeme I/II bestanden!
Das ist
technische informatik!

Jamie Vagen 09/05/2026
Hl gatte... Seite 34

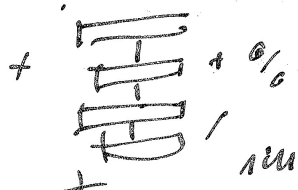
d) geld aus Deutschland
"Steuer in Deutschland"?
noch besser:
geld wegschleppen
e) ich will aber niemanden-
allein glücklich...
gehe Deutschen family nicht
auf den...
f) Noch besser: Vater tot:
ja / nein..
ich mag vergessen nicht:
bin in Deutschland..
bleib: es war nett!
Das ist der unterschied
todesurteil unterzeichne ich nicht!
ich sage was ist?
egal: für mich
g) lebt noch? Dann sage ich:
aber auch nicht ewig!
h) am Ende alles eine Erlösung!
und dann?
15.) man muß das Erbe annehmen-
wenn man es tut und das ist
besser: Warum: Weil man weiß
nicht was passiert. Denkt: Augen!

Daniel Vayda (c) 09/05/2026
All gatte... Seite 40

Es gibt schlimmer probleme:
unfähigkeit ist gift!
hier sehe ich das eine:
ich tue nichts...
Der Kopf auch nicht
sieht voll gut aus.
Dann Endlosschleife wg. gesetz!



ich darf nicht
aber ok!



ich tue:
kapiere ich worauf es hinausläuft?
b) gegenfrage: beim ersten weg?
Dann passieren sachen!
Hätte ich mal!
ich kapiere nicht! ich weiß auch
jetzt nicht!

Druid Vajda 09/05/2026, Seite 41
Ah... gatts...

Dann sage ich mir:
ich rede aber nicht mit
a) Investition
+++ Danke für Ihre Spendensammlung! +++
b) mit professor
c) ich gehe gerne wo hin
II Die feststellungen von Dr. der Medizin sind für mich altmodisch!
Es ist zwar: psychologie dicke richtig!
Danke für info...
aber doch nicht so!
fehlinfos: jemand meint ich bin
a) psychopath
b) räuber
c) abschreiber...
aber ich denke Steuerfahndung.
Zwar richtig! nur!
Man dachte anders!
Dann weiter: ich brauche mit prof zu reden. Abregnen... ok. ich verstehe Industrie so: ich sage: ein papier.
ein Millionär?! Sofort entfernt

Journal Vojda (c) 09/05/2026
Ah gotter... Seite 42

ich möchte niemanden bremsen
können!
ich verstehe Linux heute
nicht gut
aber man geht da hin?
Heute gibt es Internet!
und jetzt?
nicht nur: selbst das!
Computer gibt es lange!
Sehr lange!
Doch manchmal kommt
was abhanden
mips 32...
ist quelloffen
ok: bot gut und dann aufhören!
technische Informatik nicht mehr
3 Säulen..
bot: ok - aufhören...
me: wenn ich das mit der
Steuer mache:
Zurück zum Ende...

Jamil Vajda 09/05/2026
Alt gatte... Seite 43

Die aufgaben habe ich gemacht!
Die Bauteile
Programmiergeräte
...
schon vorher.
Computer noch da...
na ja: Systemupdate!
oh: ist gut alles da
Pfandldosen?

Steuererklärung:
a) genauso wie Computer:
gehe ich gerne überall
hin?
nein
b) Erklärung
c) Dazwischen: Sozialamt +
... alles elendige Menschen
Na ja: wenn alle so elendig
sind überlebt man manchmal
alles... Dazu: alle die irgendwas
tun wollen haben probleme...
b) Erklärung: wenn dann
[illegible] man später. Ein

David Hayden 09/05/2026
Seite 44, 11 Zeilen

dann: ok: ich soll zeug
zurückzahlen:
unlogisch

a) geld klebt an der Straße?
ich soll es nicht mitnehmen?
mit dem geld stimmt was nicht!
wenn geld nicht anzieht hat es...

b) zurückzahlen: Steuer
1.) unangekündigte pünktlichkeit

a) angeblich gab es früher bravo
Wirtschaftsbetrug.
Wegewegen: wegen Immobilie?
Wein wg. dem Lies: früher hatten
alle schönes Buch
Mathematik geht nicht
mit beweisen...
Ich vermute manch Lies macht
deswegen: Werk der Wissenschaft:
weil es verhaftet es als letzte
Hinterlassenschaft an Menschheit

David Vogel (c) 09/05/2026
Alle gatte, Seite 45

nein: um meine
geschäftsbücher.
ich habe auch alles
bis zur Sozialhilfe waren
alle Daten jeder von drin...
Dann nicht mehr?
warum?
Eben deswegen. Dann sagt man:
hier fehlt was. Das ist wichtig!
ist teil von Erklärung!
ohne Spott: weil ich
angst kriegte.
teil der Erklärung: wo man
vielleicht früher... oft vielleicht
anders... nicht nur zum guten
spitze verlauf von Wissenschaft?
am anfang eher so?
als ich Kind war: hatte ich
alles:
aber vor allem richtiges?
Dann sage ich: geschäftsbuch
+ Erklärung...

Daniel Vojda 09/05/2026

Seite 46, 1tl gatte...

nur warum im netz?

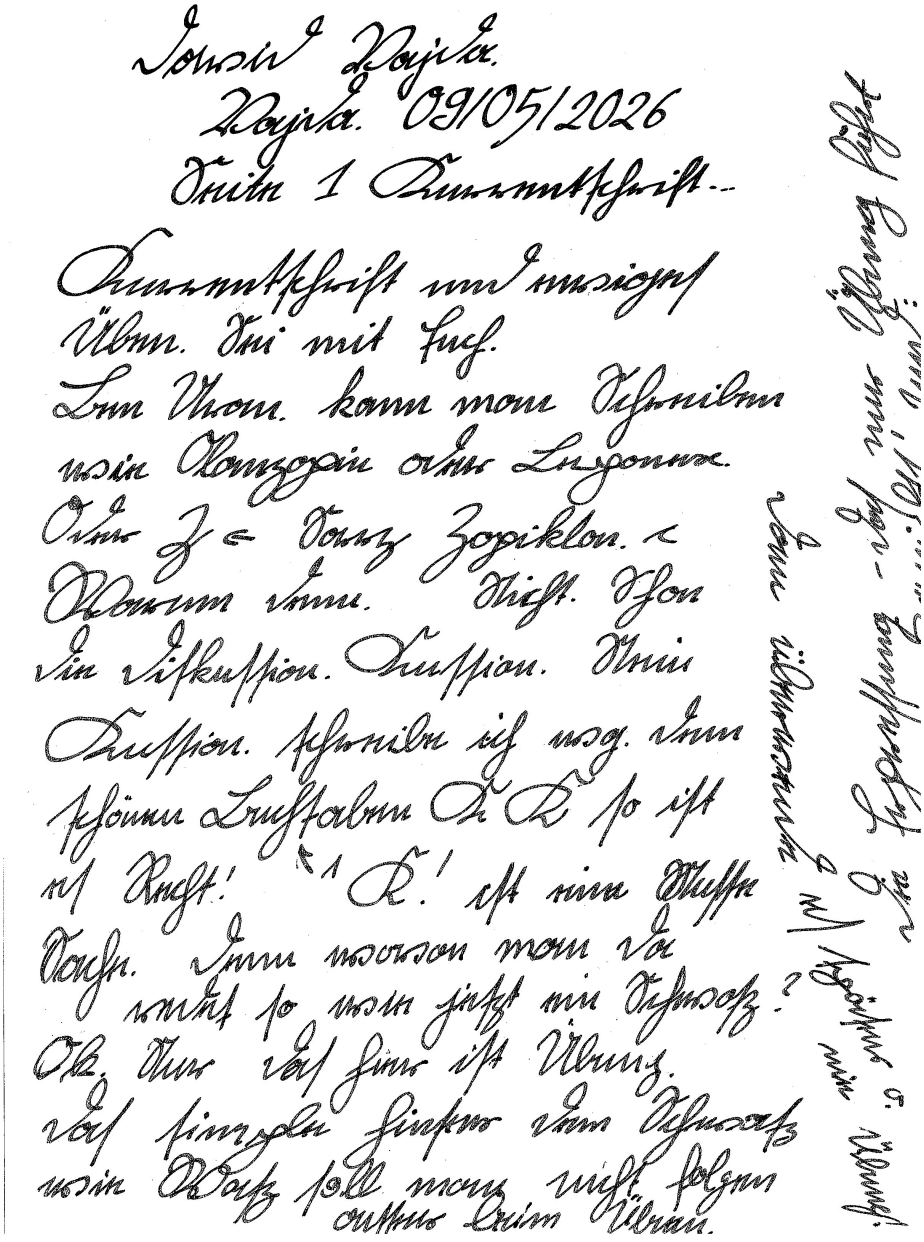
1.) weil ich angeber bin
– logisch

2.) als ich ... als Leute
geheim wg. nicht... den..
Die Leute dachten

Durch geheim schafft man geheim
Es weiß ja am Ende alles

2x geheim auch nicht...

und oder: Dann wissen es andere
(Zum wen...)
usw...





Copy Add to Informed Delivery

22:24 19

Latest Update

Your item departed a transfer airport in FRANKFURT, FRANKFURT, GERMANY on September 4, 2026 at 10:38 am. The item is currently in transit to the destination.

Get More Out of USPS Tracking:

USPS Tracking Plus®

International Transit

Departed

Frankfurt, GERMANY
September 4, 2026 10:38 AM

Departed

New York City, UNITED STATES
September 3, 2026 6:50 PM

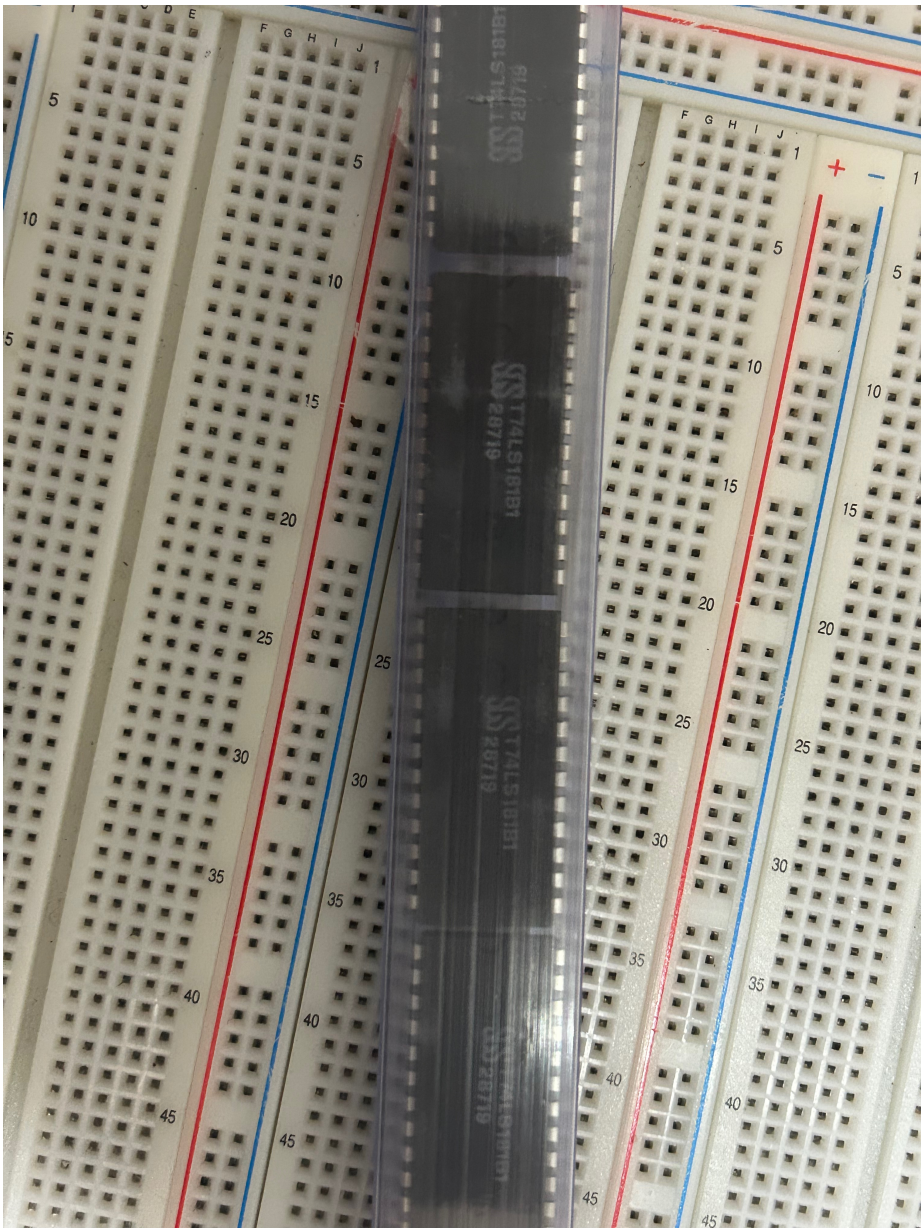
See All Tracking History

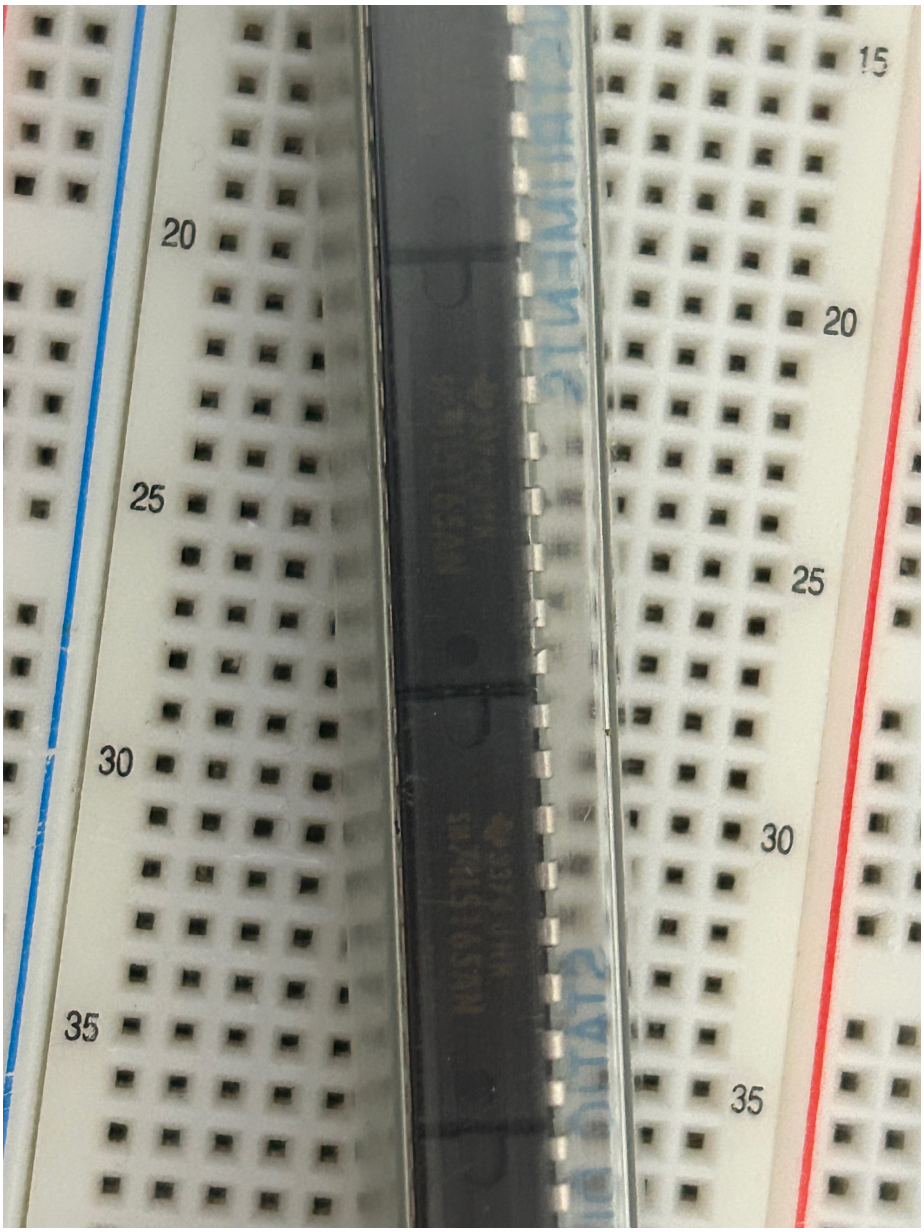
What Do USPS Tracking Statuses Mean?

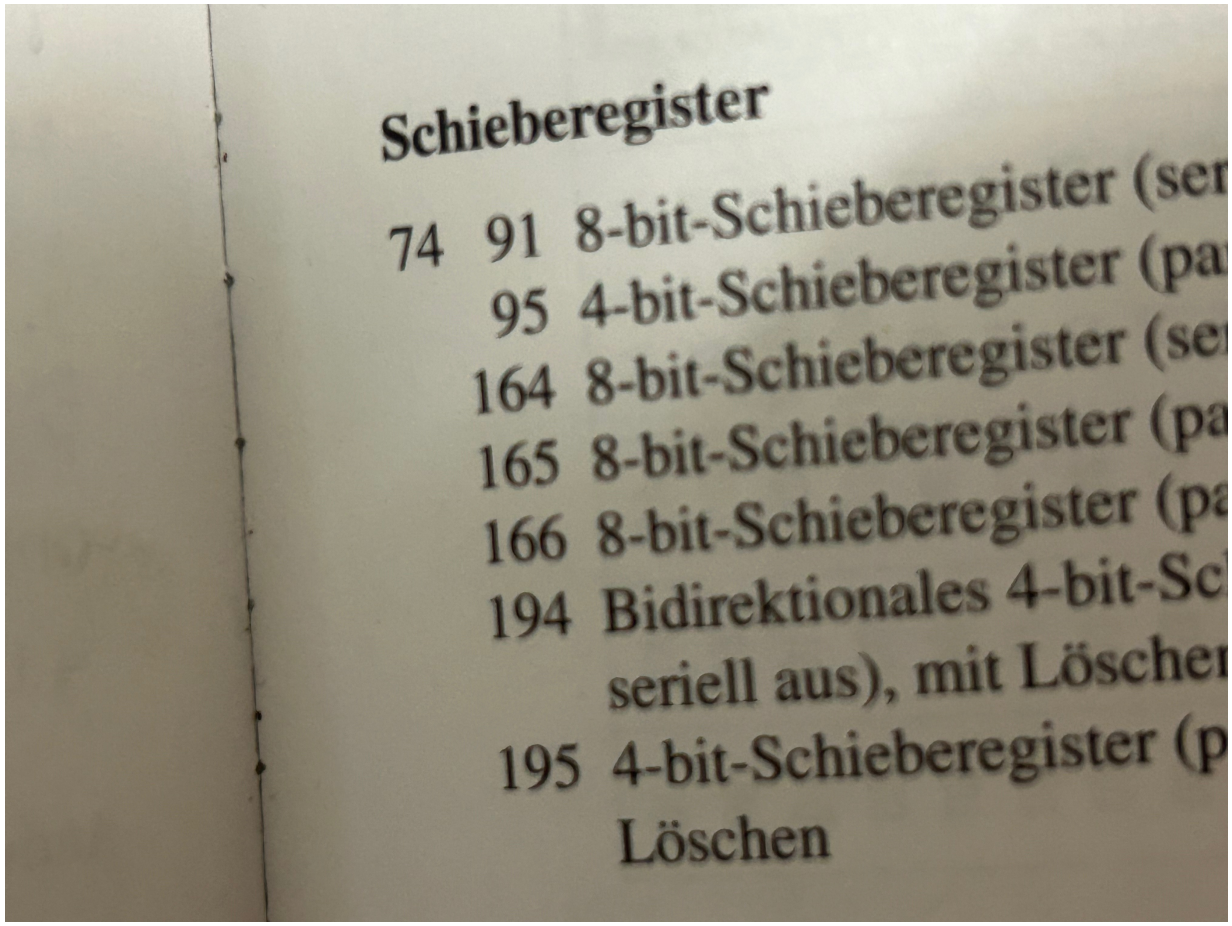
Text & Email Updates

USPS Tracking Plus®

EIC-408

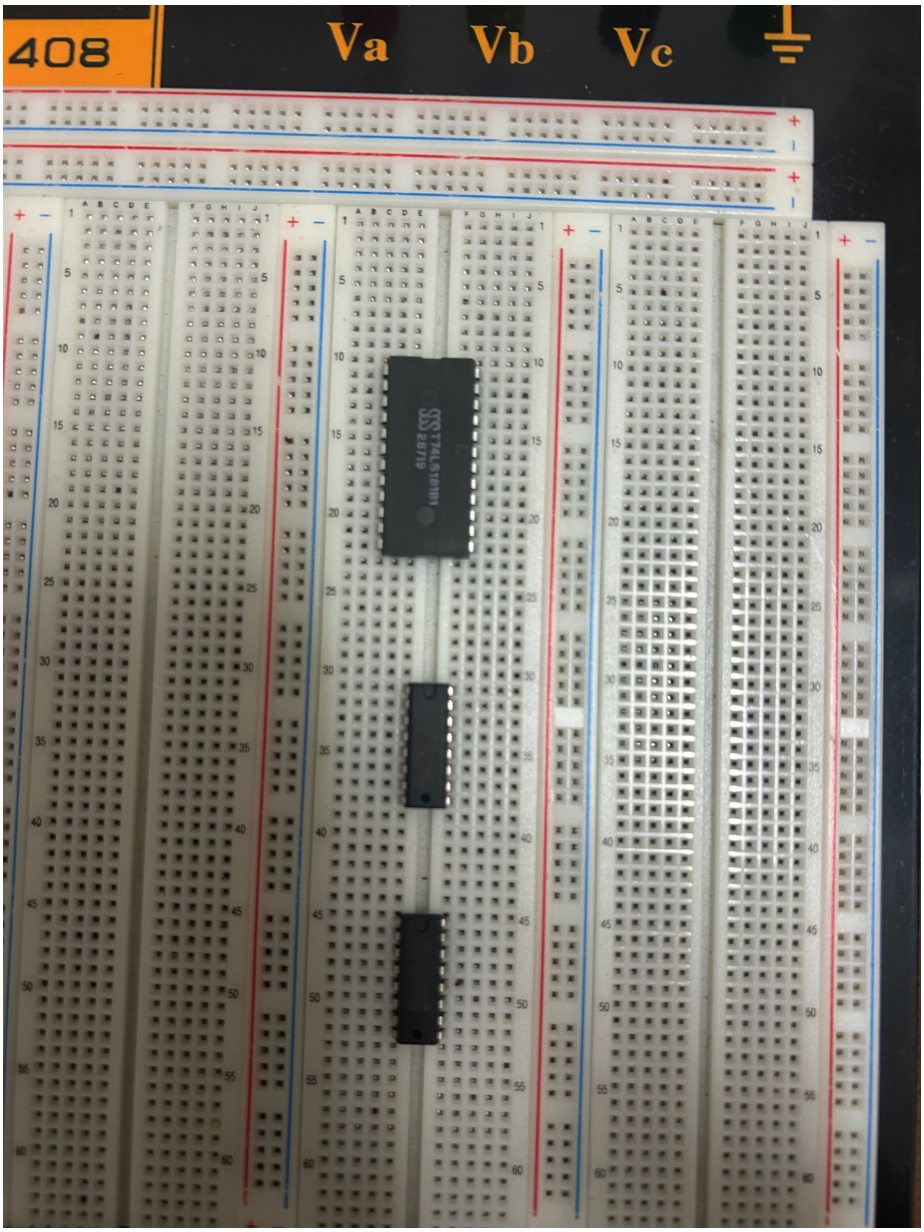


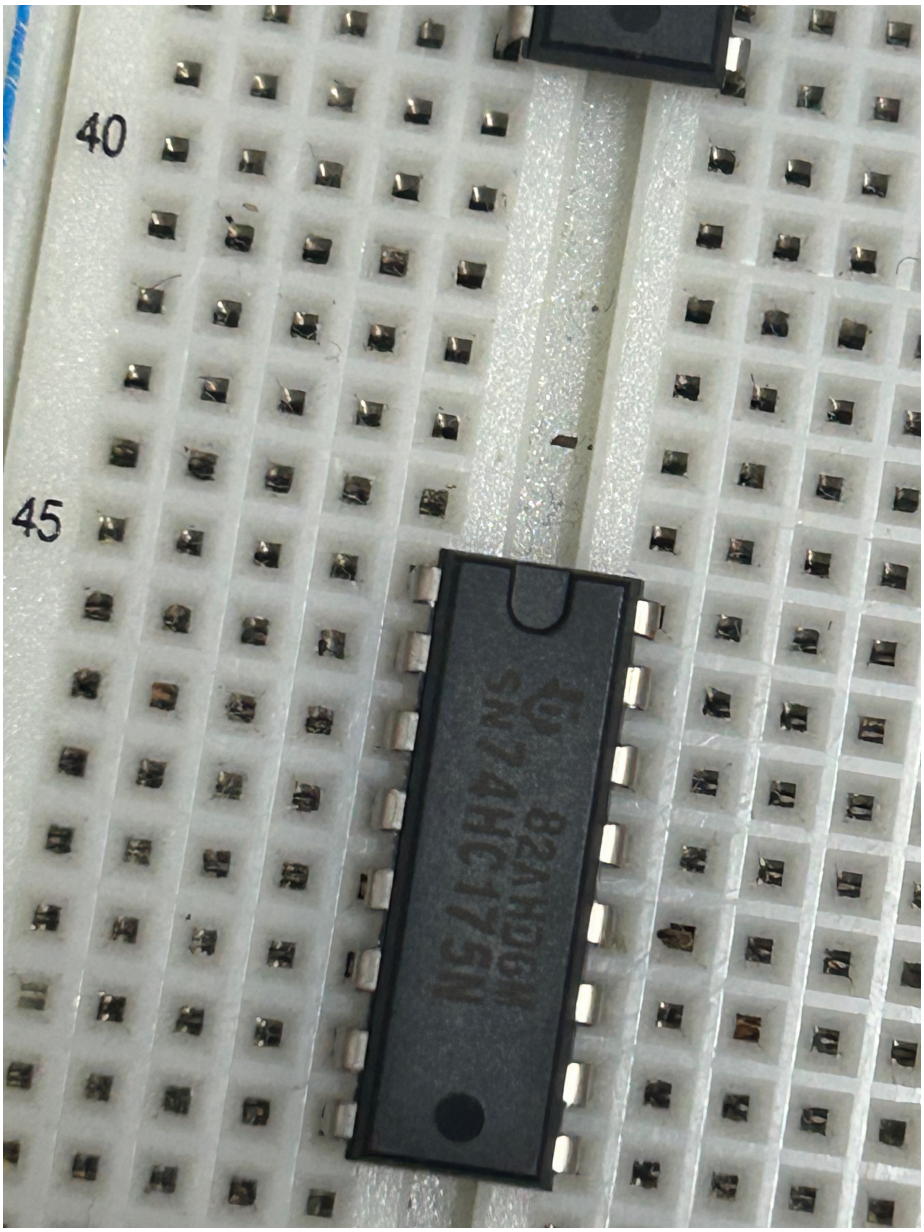

15
20
25
30
35
20
25
30
35



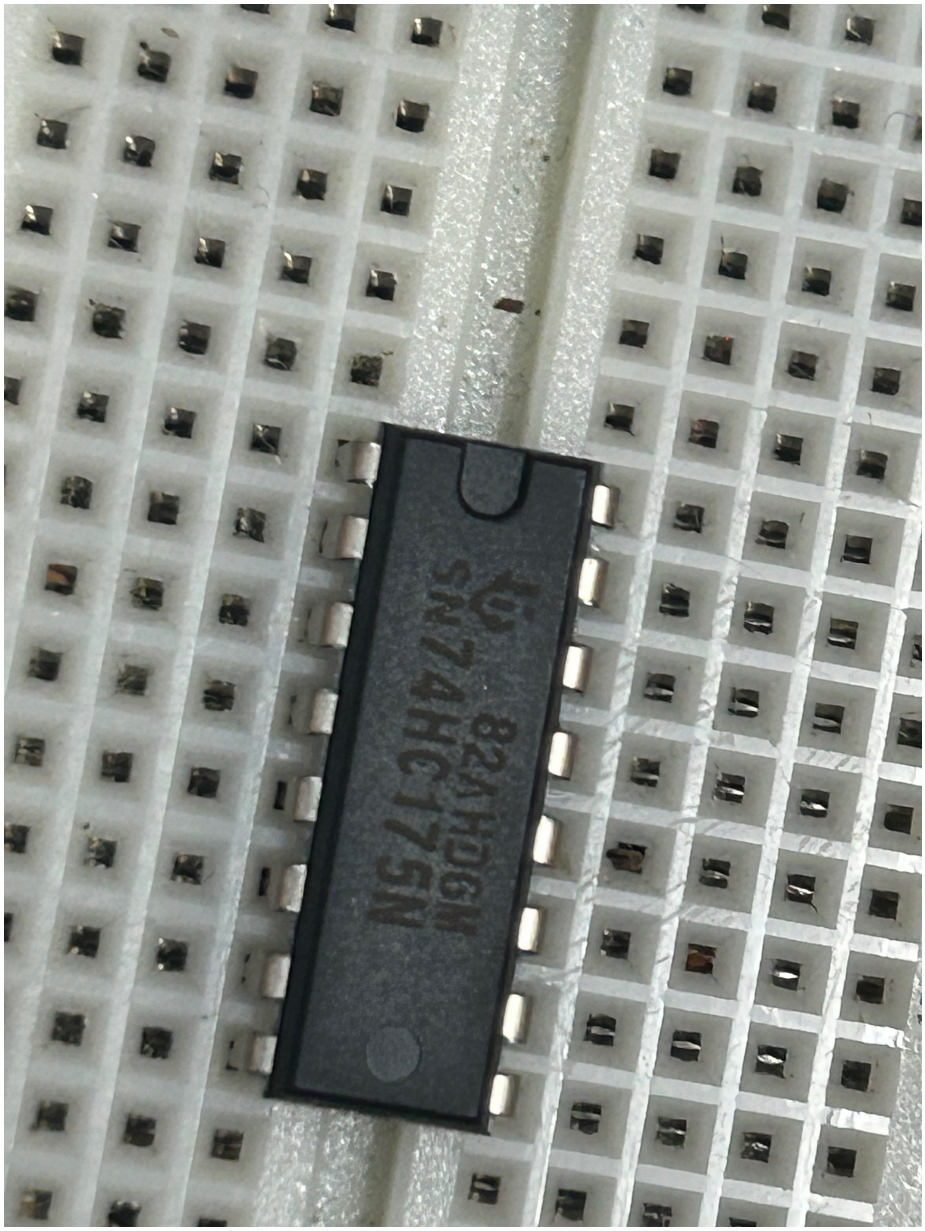
Schieberegister

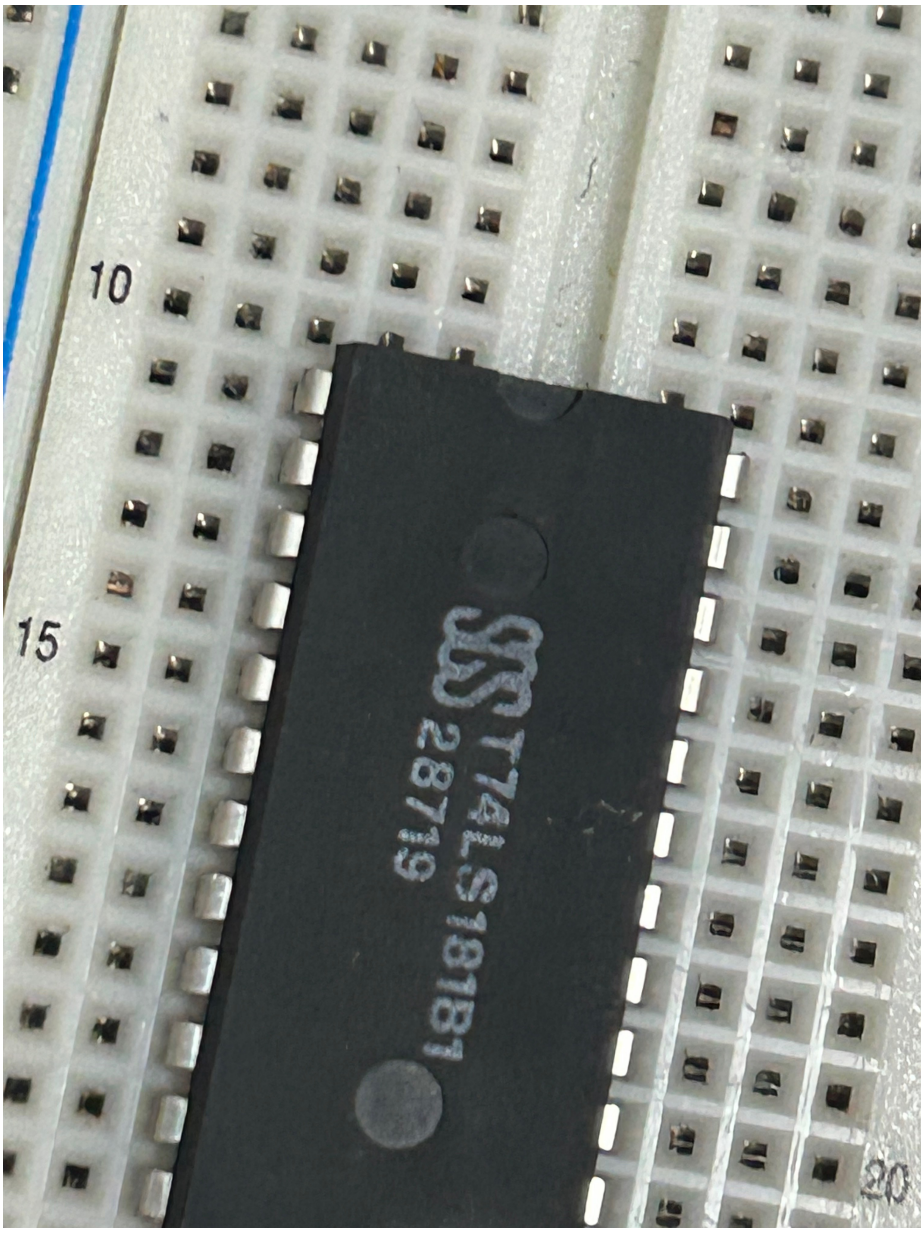
74 91 8-bit-Schieberegister (ser
95 4-bit-Schieberegister (pa
164 8-bit-Schieberegister (se
165 8-bit-Schieberegister (pa
166 8-bit-Schieberegister (pa
194 Bidirektionales 4-bit-Sc
seriell aus), mit Löscher
195 4-bit-Schieberegister (p
Löschen


408
Va
Vb
Vc


40
45
SN74HC174N
M9GHV28

10.9 next




10
15
20
SN74LS181N
2879

23:00

Cours Electronique et I...

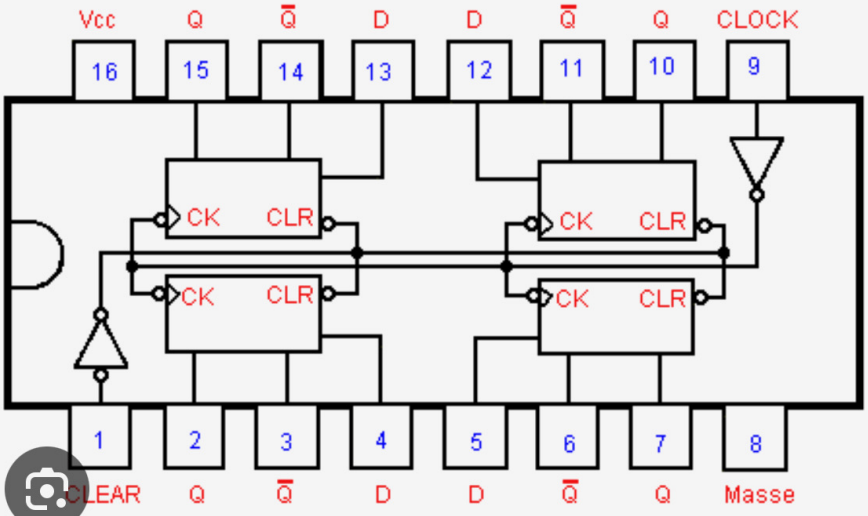


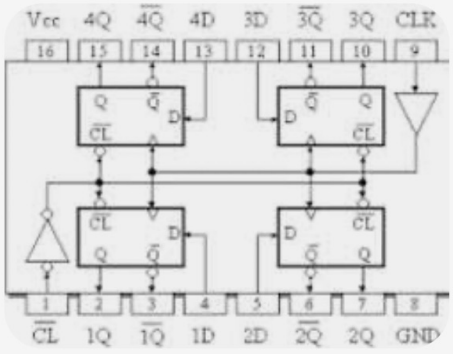
Fig. 51. - Brochage du circuit intégré 74175.

Synchronous D and JK flip-flops available in Integrated...

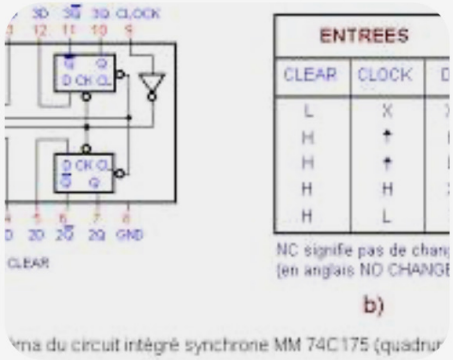
Aufrufen ›

Teilen

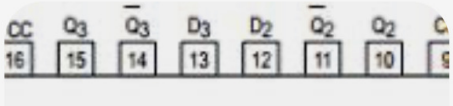
Speichern

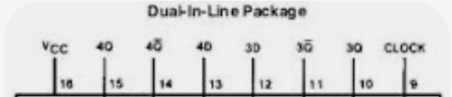


74175 IC Quad D-type flip-flo...
www.amazon.eg



Study of the Series Offset...
Cours Electronique et...



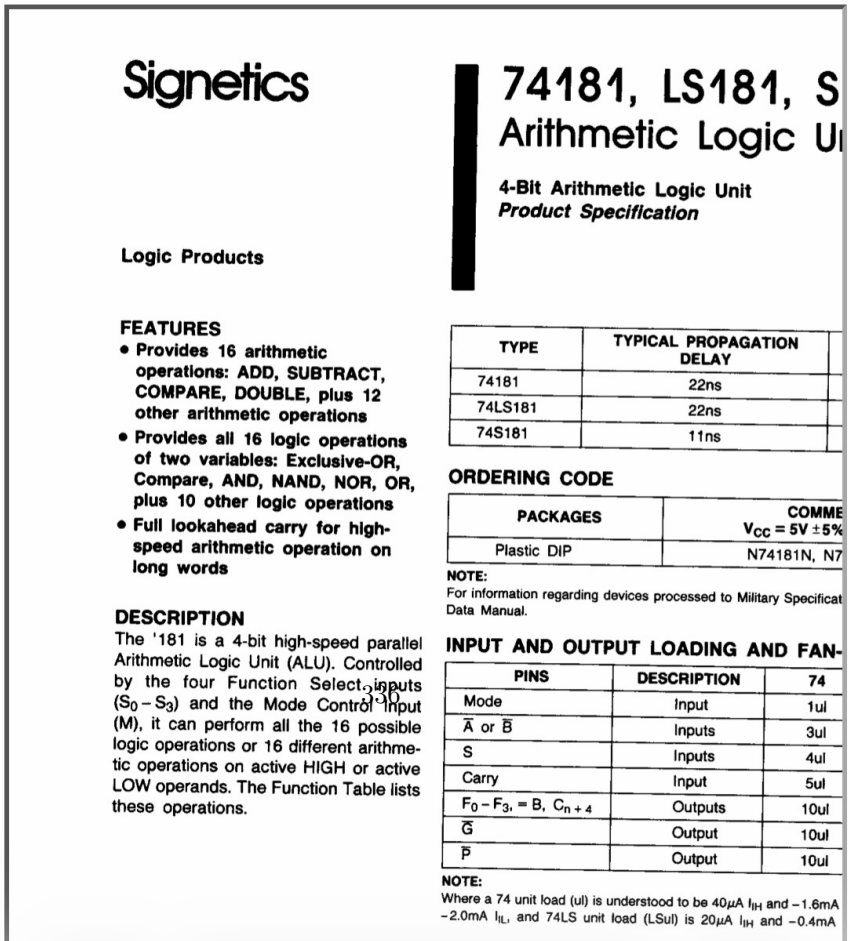


74181 Datasheet Download

74181 datasheet details, specifications, and related parts are available on the Main page: 74181 Datasheet Details

Part No.	74181
Manufacturer	Philips Semiconductors
Description	4 Bit Arithmetic Logic Unit
Size	488.90 KB

Download 74181 Datasheet PDF



Signetics

Logic Products

74181, LS181, S
Arithmetic Logic U

4-Bit Arithmetic Logic Unit
Product Specification

FEATURES
- Provides 16 arithmetic operations: ADD, SUBTRACT, COMPARE, DOUBLE, plus 12 other arithmetic operations
- Provides all 16 logic operations of two variables: Exclusive-OR, Compare, AND, NAND, NOR, OR, plus 10 other logic operations
- Full lookahead carry for high-speed arithmetic operation on long words

DESCRIPTION
The '181 is a 4-bit high-speed parallel Arithmetic Logic Unit (ALU). Controlled by the four Function Select inputs ($S_0 - S_3$) and the Mode Control input (M), it can perform all the 16 possible logic operations or 16 different arithmetic operations on active HIGH or active LOW operands. The Function Table lists these operations.

TYPE	TYPICAL PROPAGATION DELAY
74181	22ns
74LS181	22ns
74S181	11ns

ORDERING CODE

PACKAGES	COMME $V_{CC} = 5V \pm 5\%$
Plastic DIP	N74181N, N7

NOTE:
For information regarding devices processed to Military Specificat Data Manual.

INPUT AND OUTPUT LOADING AND FAN-

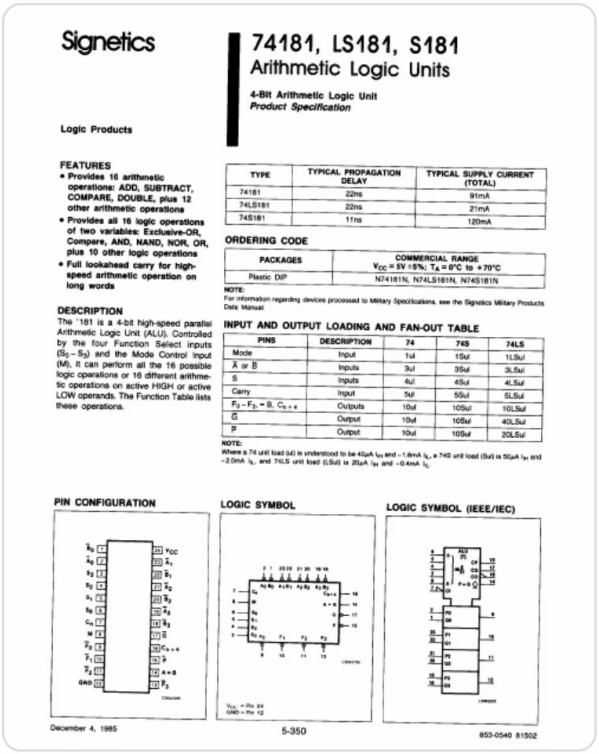
PINS	DESCRIPTION	74
Mode	Input	1ul
$\overline{A}$ or $\overline{B}$	Inputs	3ul
S	Inputs	4ul
Carry	Input	5ul
$F_0 - F_3$, = B, C_{n+4}	Outputs	10ul
$\overline{G}$	Output	10ul
$\overline{P}$	Output	10ul

NOTE:
Where a 74 unit load (ul) is understood to be 40μA I_{IH} and −1.6mA −2.0mA I_{IL}, and 74LS unit load (LSul) is 20μA I_{IH} and −0.4mA

23:00

74181 Datasheet | Philips Semiconductors

74181 is 4 Bit Arithmetic Logic Unit manufactured by Philips Semiconductors.



Signetics

74181, LS181, S181
Arithmetic Logic Units

4-Bit Arithmetic Logic Unit
Product Specification

Logic Products

FEATURES
- Provides 16 arithmetic operations: ADD, SUBTRACT, COMPARE, DOUBLE, plus 12 other arithmetic operations
- Provides all 16 logic operations of two variables: Exclusive-OR, Compare, AND, NAND, NOR, OR, plus 10 other logic operations
- Full lookahead carry for high-speed arithmetic operation on long words

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74181	22ns	91mA
74LS181	22ns	21mA
74S181	11ns	120mA

ORDERING CODE

PACKAGES	COMMERCIAL RANGE V_{CC} = 5V ±5%; T_A = 0°C to +70°C
Plastic DIP	N74181N, N74LS181N, N74S181N

NOTE:
For information regarding devices processed to Military Specifications, see the Signetics Military Products Data Manual.

DESCRIPTION
The '181 is a 4-bit high-speed parallel Arithmetic Logic Unit (ALU). Controlled by the four Function Select inputs (S_0 – S_3) and the Mode Control input (M), it can perform all the 16 possible logic operations or 16 different arithmetic operations on active HIGH or active LOW operands. The Function Table lists these operations.

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

PINS	DESCRIPTION	74	74S	74LS
Mode	Input	1ul	1Sul	1LSul
$\overline{A}$ or $\overline{B}$	Inputs	3ul	3Sul	3LSul
S	Inputs	4ul	4Sul	4LSul
Carry	Input	5ul	5Sul	5LSul
F_0 – F_3, $\overline{G}$, C_{n+4}	Outputs	10ul	10Sul	10LSul
$\overline{G}$	Output	10ul	10Sul	40LSul
$\overline{P}$	Output	10ul	10Sul	20LSul

NOTE:
Where a 74 unit load (ul) is understood to be 40μA I_{IH} and −1.6mA I_{IL}, a 74S unit load (Sul) is 50μA I_{IH} and −2.0mA I_{IL}, and 74LS unit load (LSul) is 20μA I_{IH} and −0.4mA I_{IL}.

PIN CONFIGURATION

LOGIC SYMBOL

LOGIC SYMBOL (IEEE/IEC)

December 4, 1985 5-350 853-0540 81502

Part No.: **74181**

Brand: PHILIPS

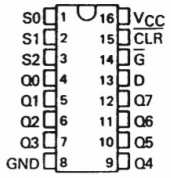
Philips Semiconductors

Unit Price: 16.19 USD

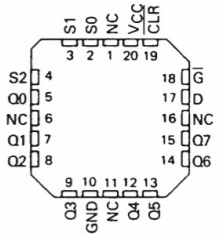
Buy: Stock & Price

- 8-Bit Parallel-Out Storage Register Performs Serial-to-Parallel Conversion with Storage
- Asynchronous Parallel Clear
- Active High Decoder
- Enable/Disable Input Simplified Expansion
- Expandable for N-Bit Applications
- Four District Functional Modes
- Package Options Include Ceramic Chip Carriers and Flat Packages in Addition to Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

SN54259, SN54LS259B . . . J OR W PACKAGE
SN74259 . . . N PACKAGE
SN74LS259B . . . D OR N PACKAGE
(TOP VIEW)



SN54LS259B . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

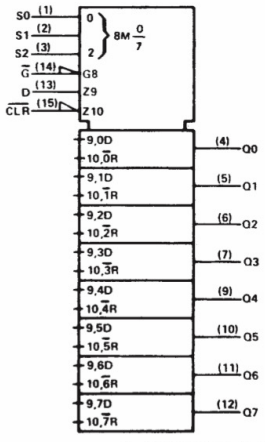
description

These 8-bit addressable latches are designed for general purpose storage applications in digital systems. Specific uses include working registers, serial-holding registers, and active-high decoders or demultiplexers. They are multifunctional devices capable of storing single-line data in eight addressable latches, and being a 1-of-8 decoder or demultiplexer with active-high outputs.

Four distinct modes of operation are selectable by controlling the clear ($\overline{CLR}$) and enable ($\overline{G}$) inputs as enumerated in the function table. In the addressable-latch mode, data at the data-in terminal is written into the addressed latch. The addressed latch will follow the data input with all unaddressed latches remaining in their previous states. In the memory mode, all latches remain in their previous states and are unaffected by the data or address inputs. To eliminate the possiblity of entering erroneous data in the latches, enable $\overline{G}$ should be held high (inactive) while the address lines are changing. In the 1-of-8 decoding or demultiplexing mode, the addressed output will follow the level of the D input with all other outputs low. In the clear mode, all outputs are low and unaffected by the address and data inputs.

The SN54259 and SN54LS259B are characterized for operation over the full military temperature range of −55°C to 125°C. The SN74259 and SN74LS259B are characterized for operation from 0°C to 70°C.

logic symbol†



†This symbol is in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

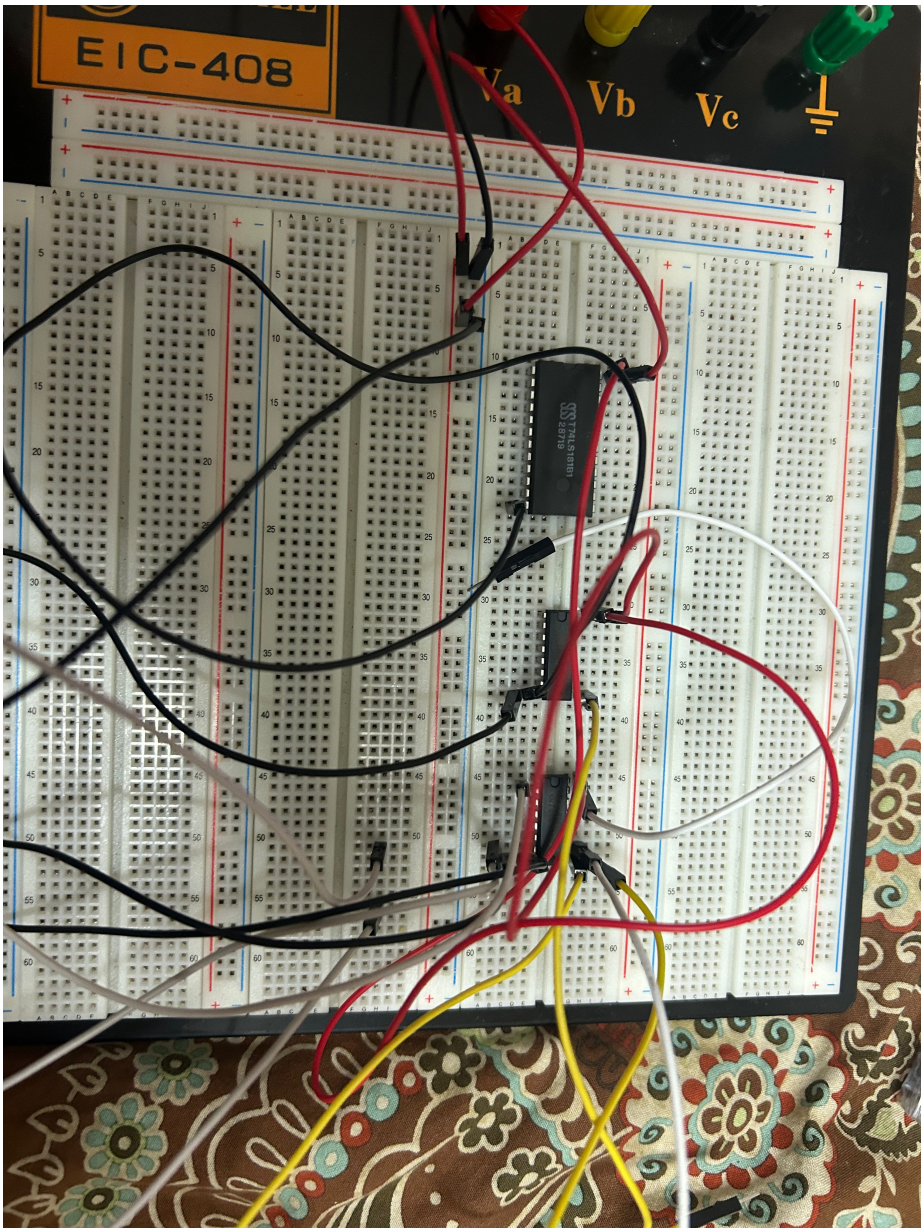
Copyright © 1988, Texas Instruments Incorporat

FUNCTION TABLE

	OUTPUT OF EACH	

LATCH SELECTION TABLE

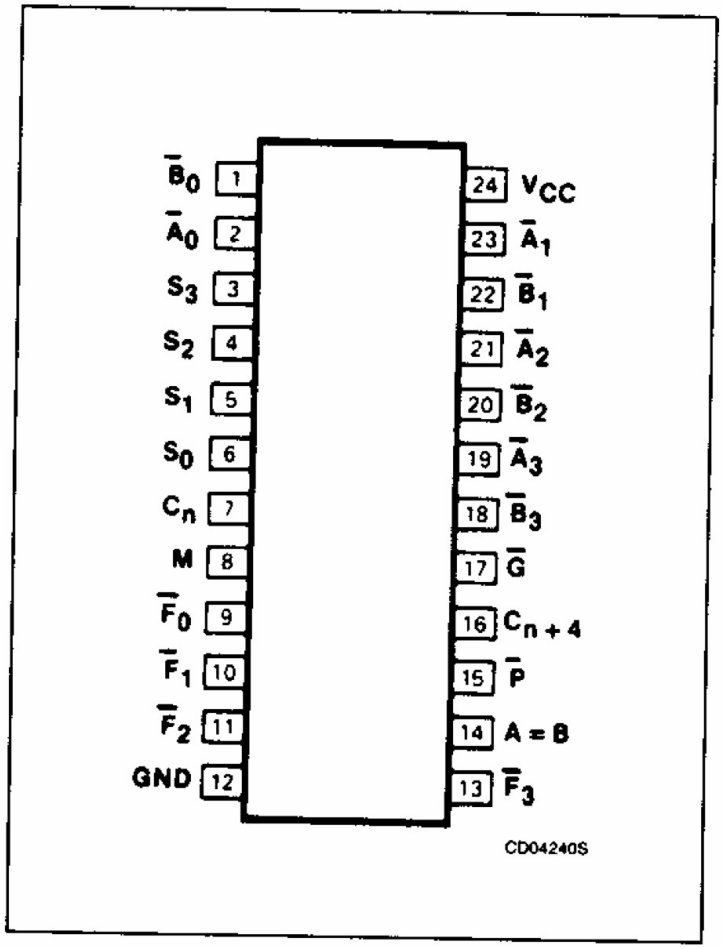
SELECT INPUTS	LATCH


EIC-408
Va
Vb
Vc

23:04

NOTI
Wher
−2.0r

PIN CONFIGURATION

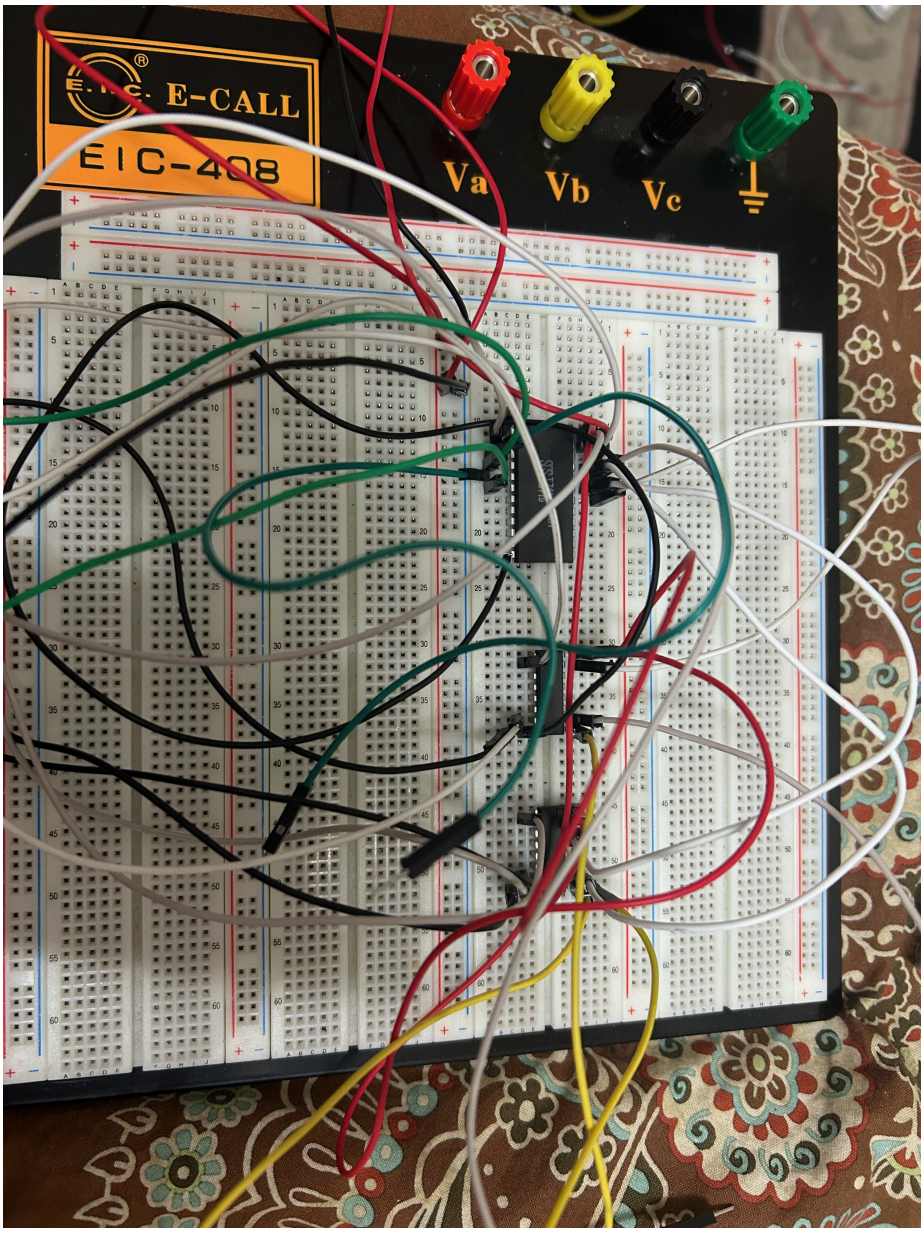


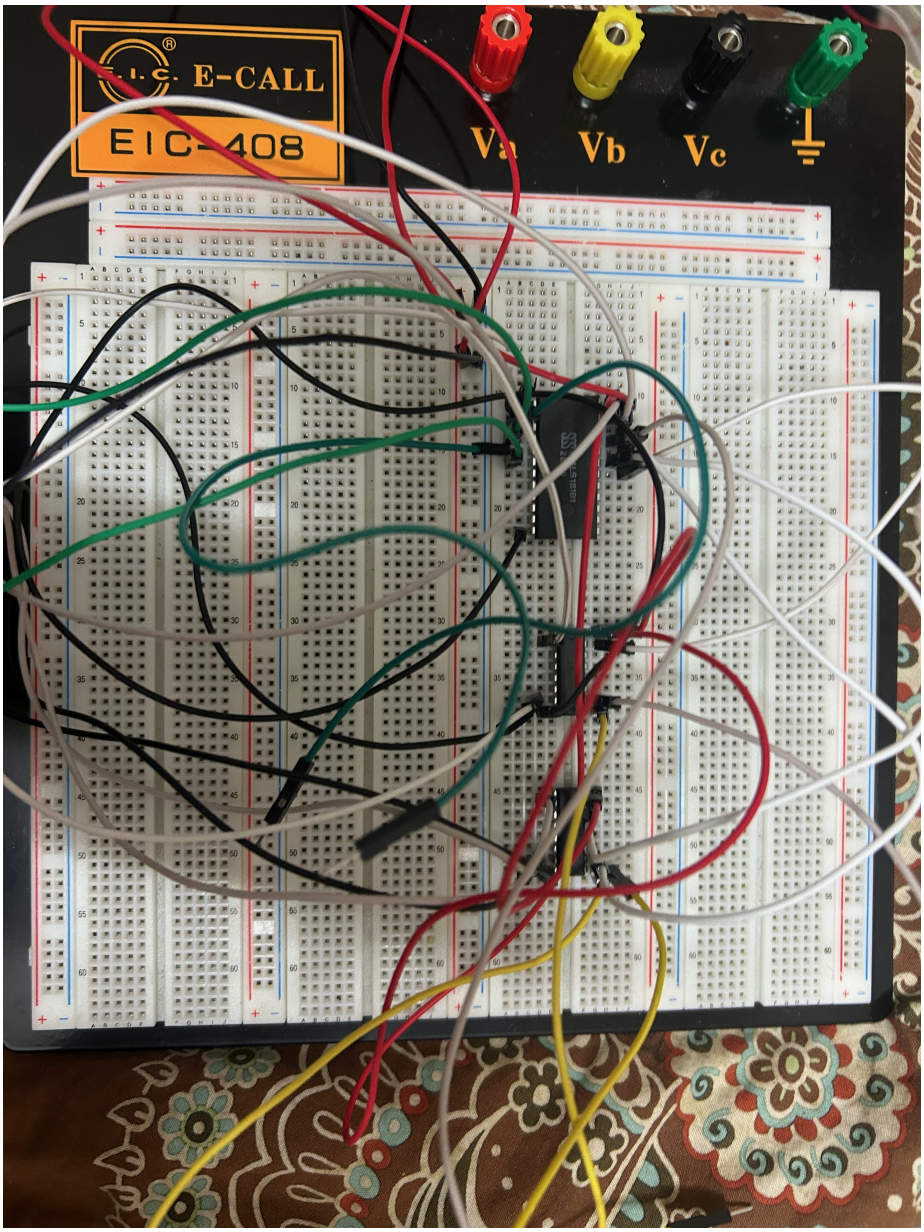
LOC

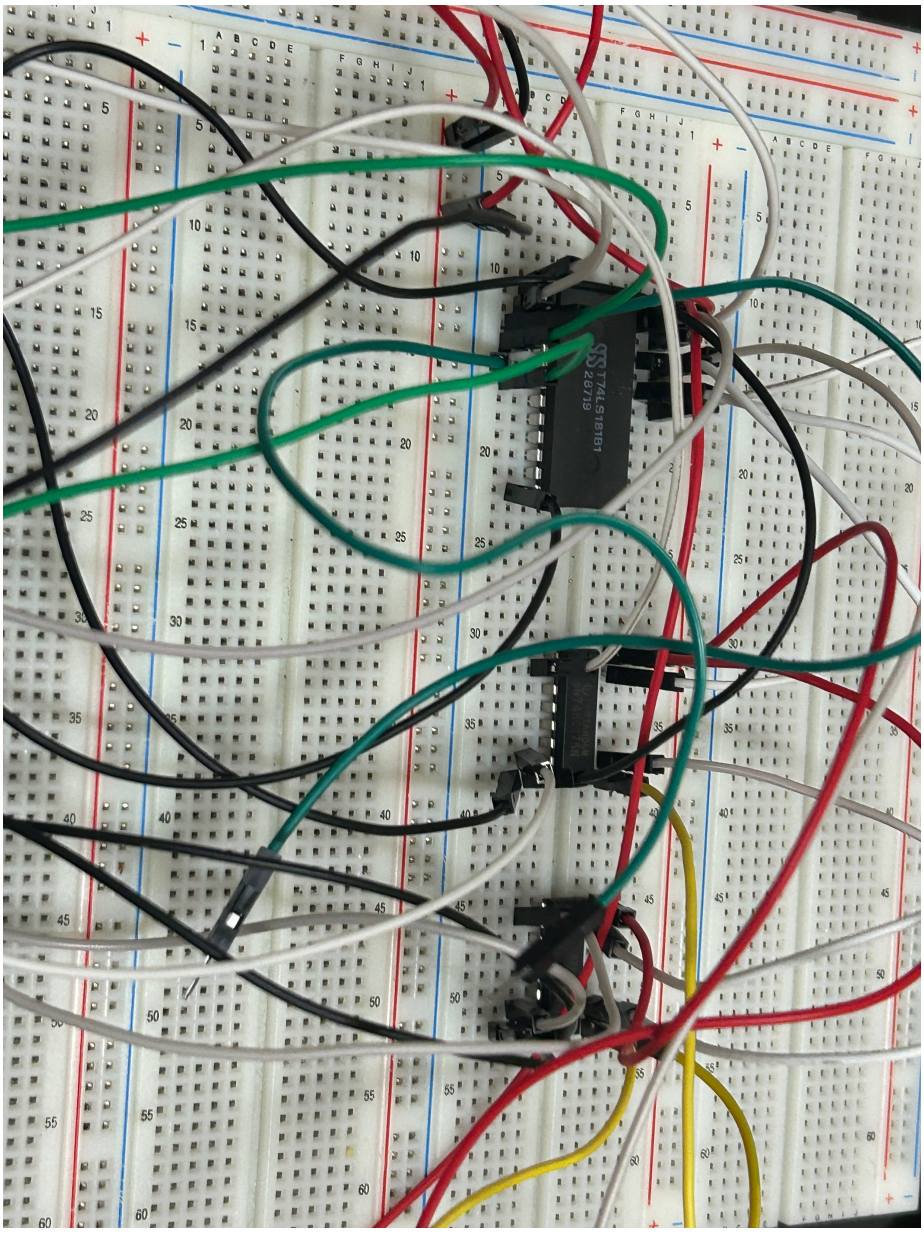
7
8
6
5
4
3

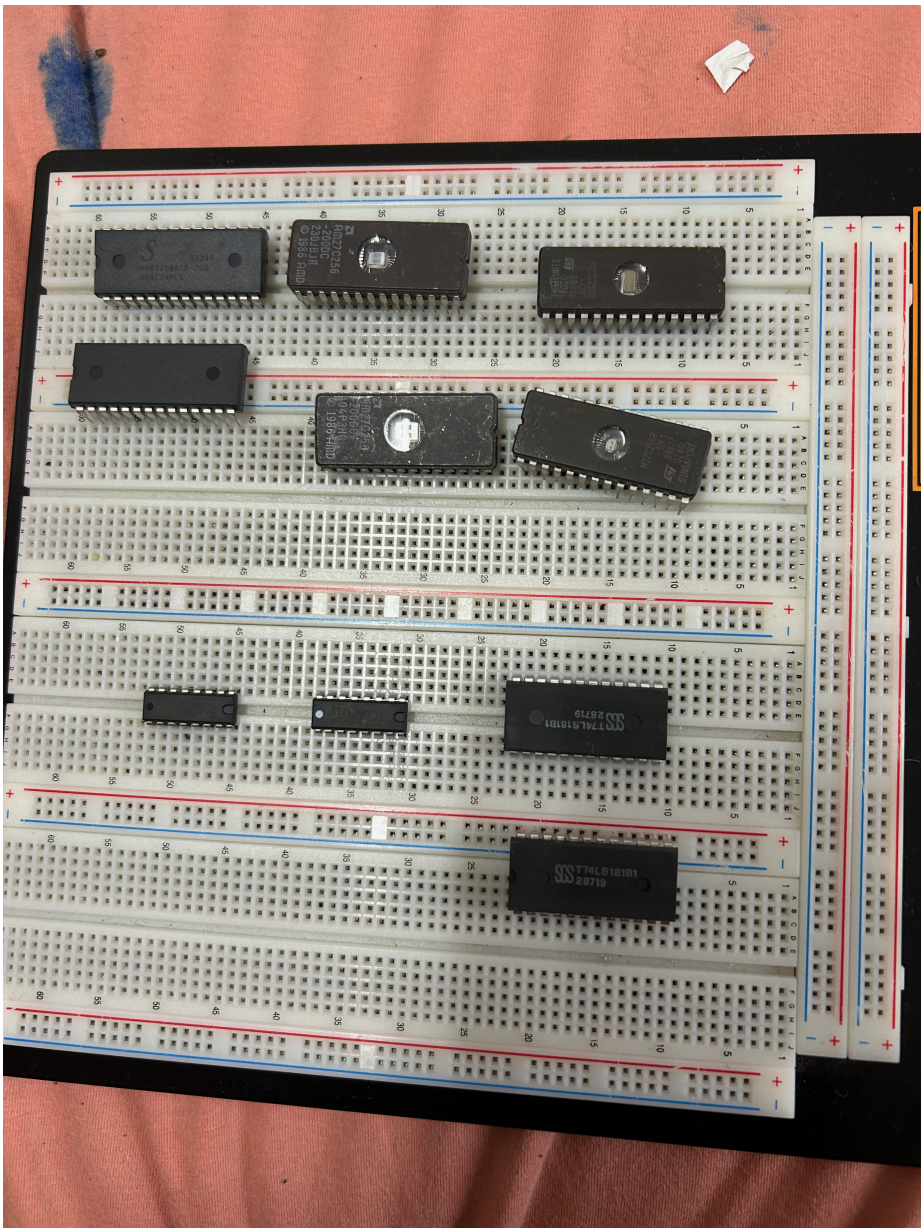
V_{CC}
GND

December 4, 1985

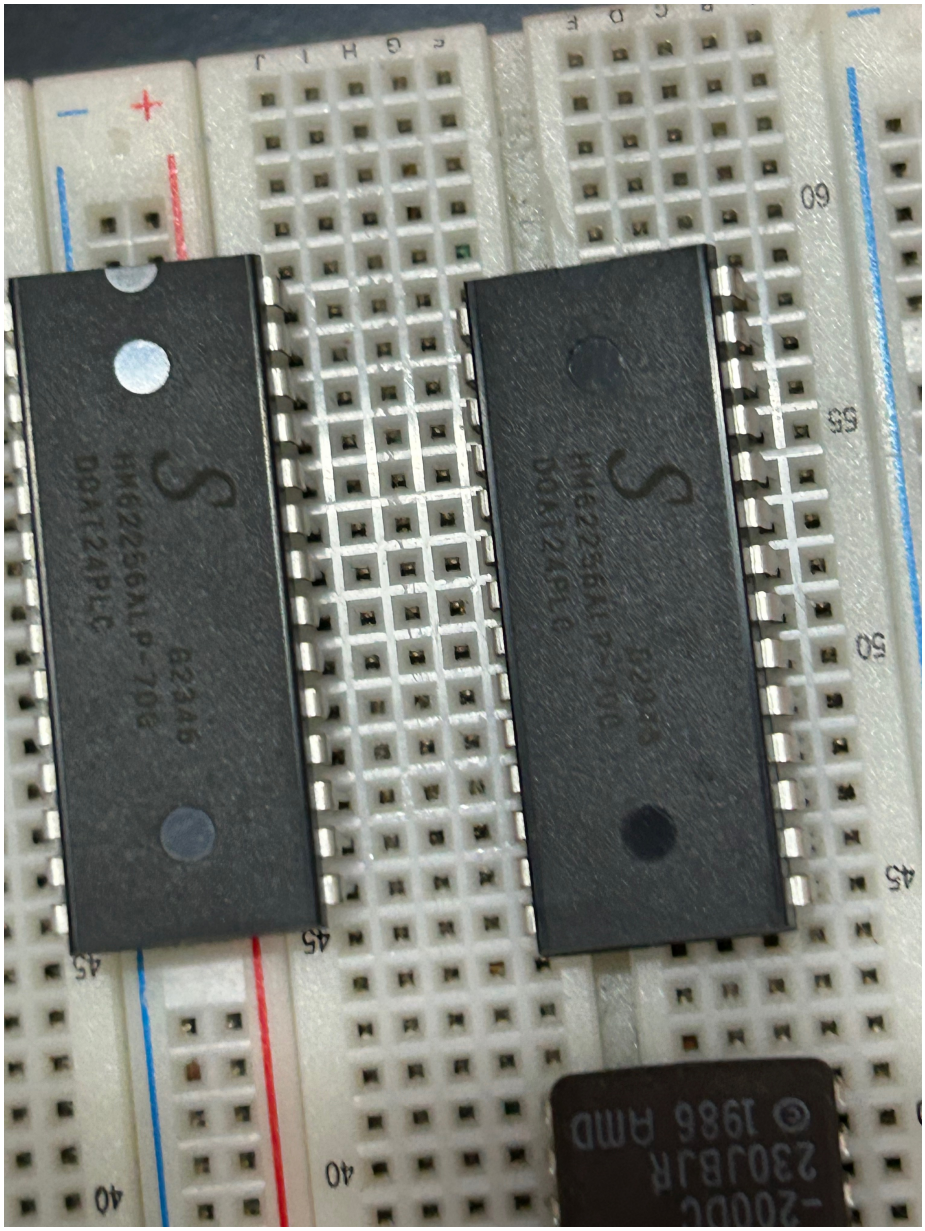

E-CALL
EIC-408
Va
Vb
Vc

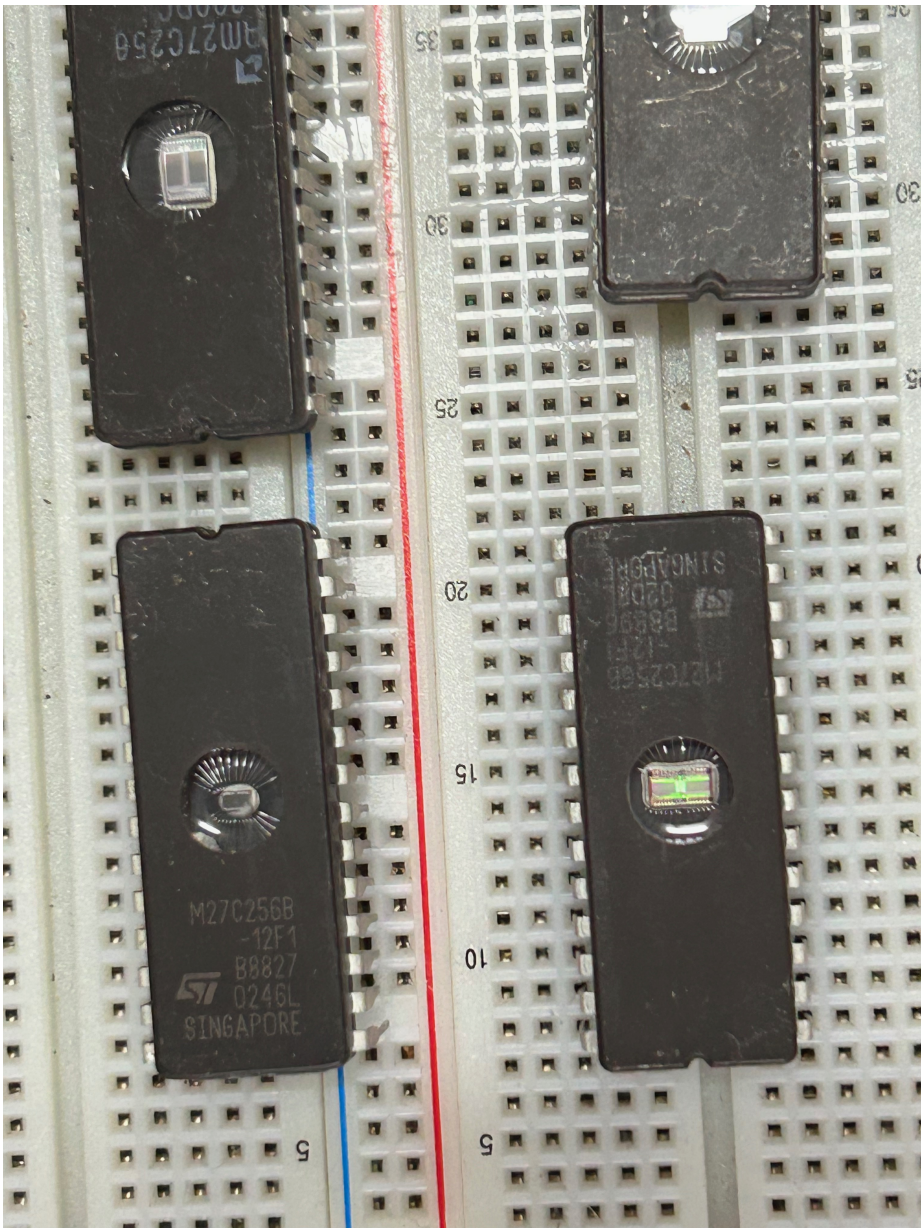

E-CALL
EIC-408
Va
Vb
Vc

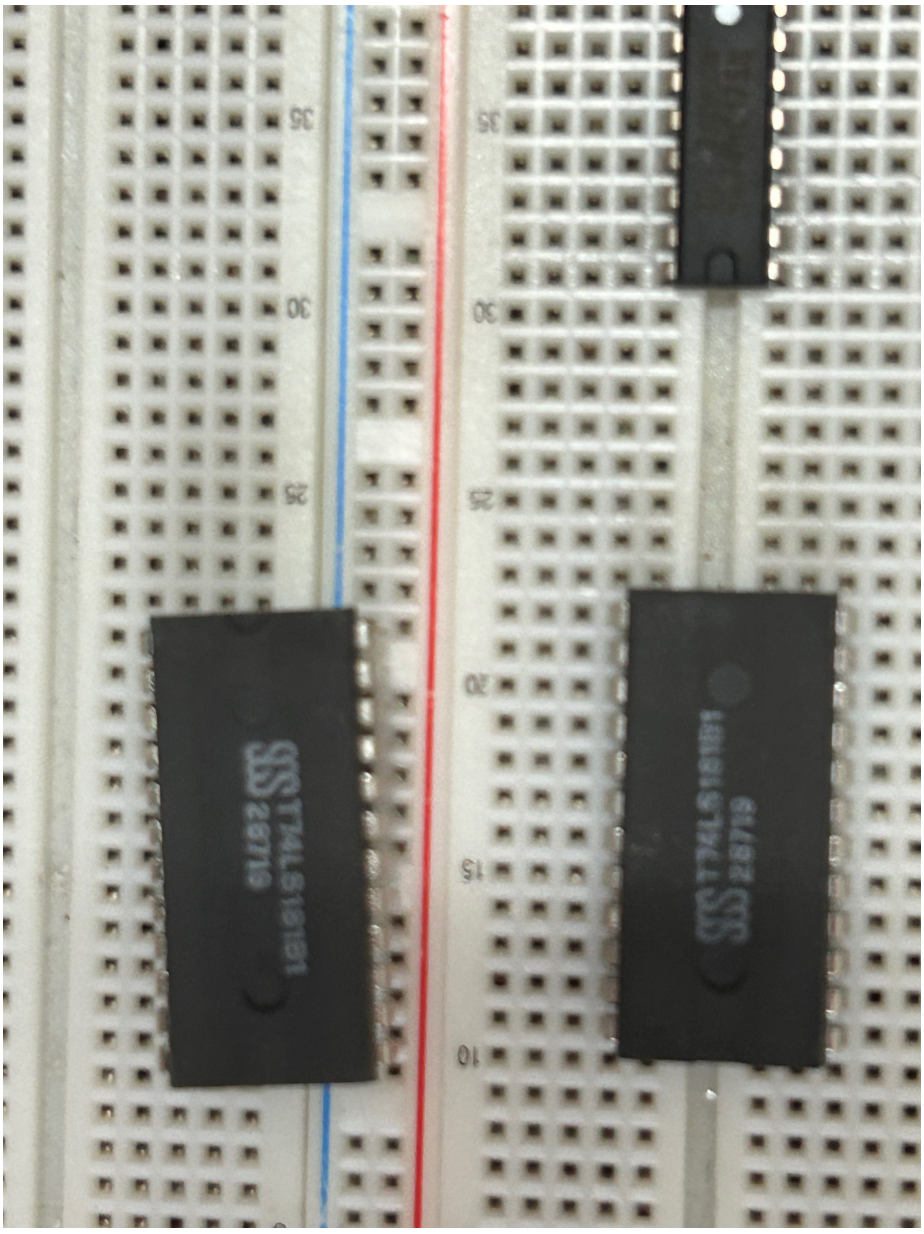

74LS161
2671J

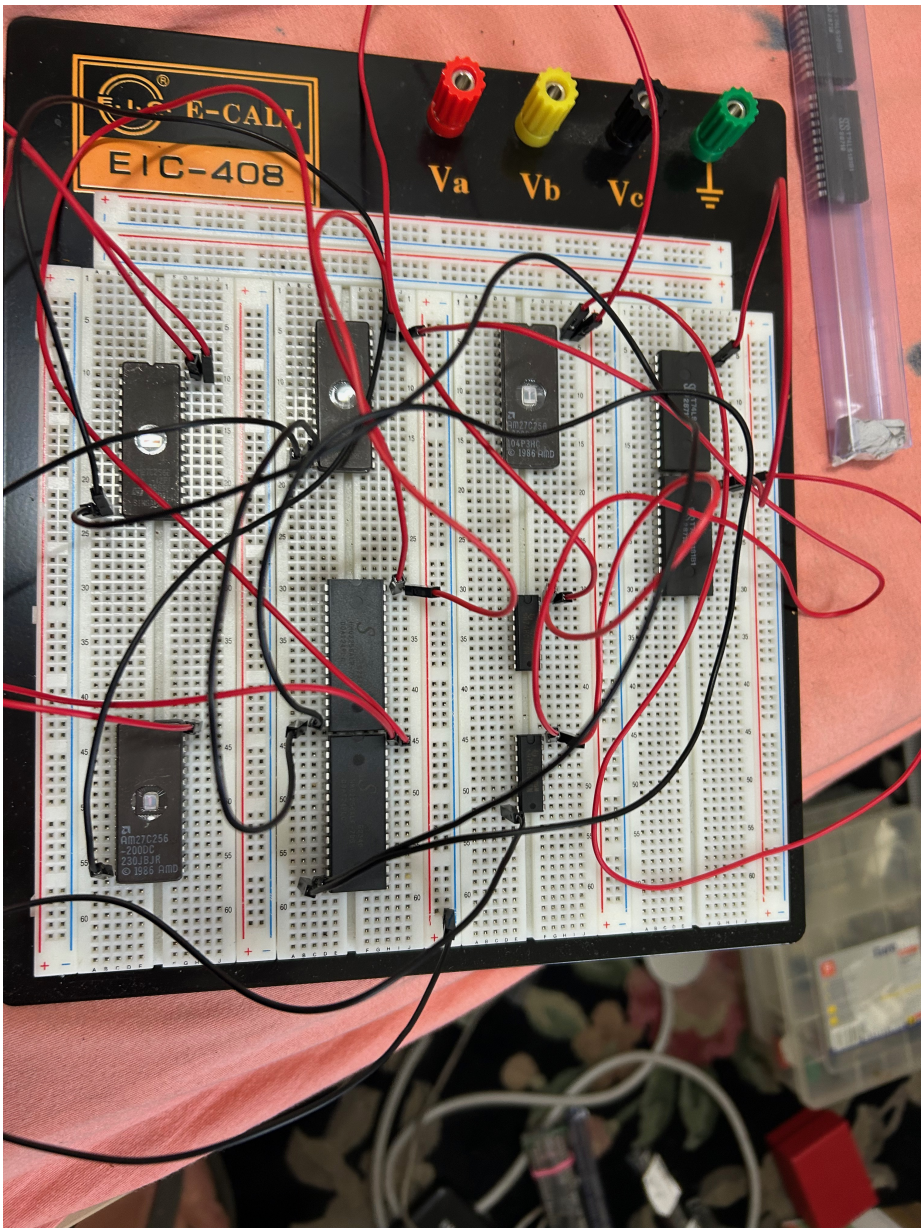

T74LS181B1
28719

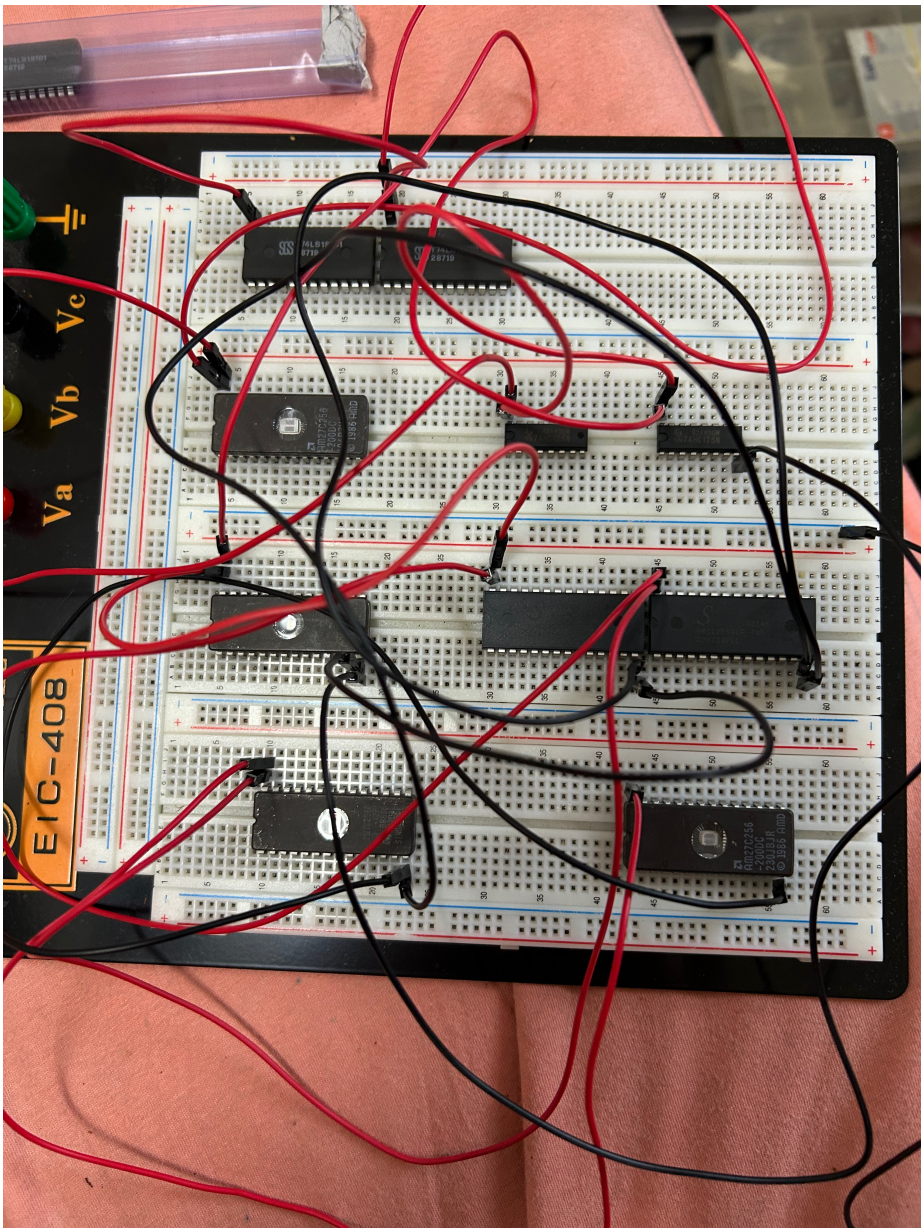
10.10 next

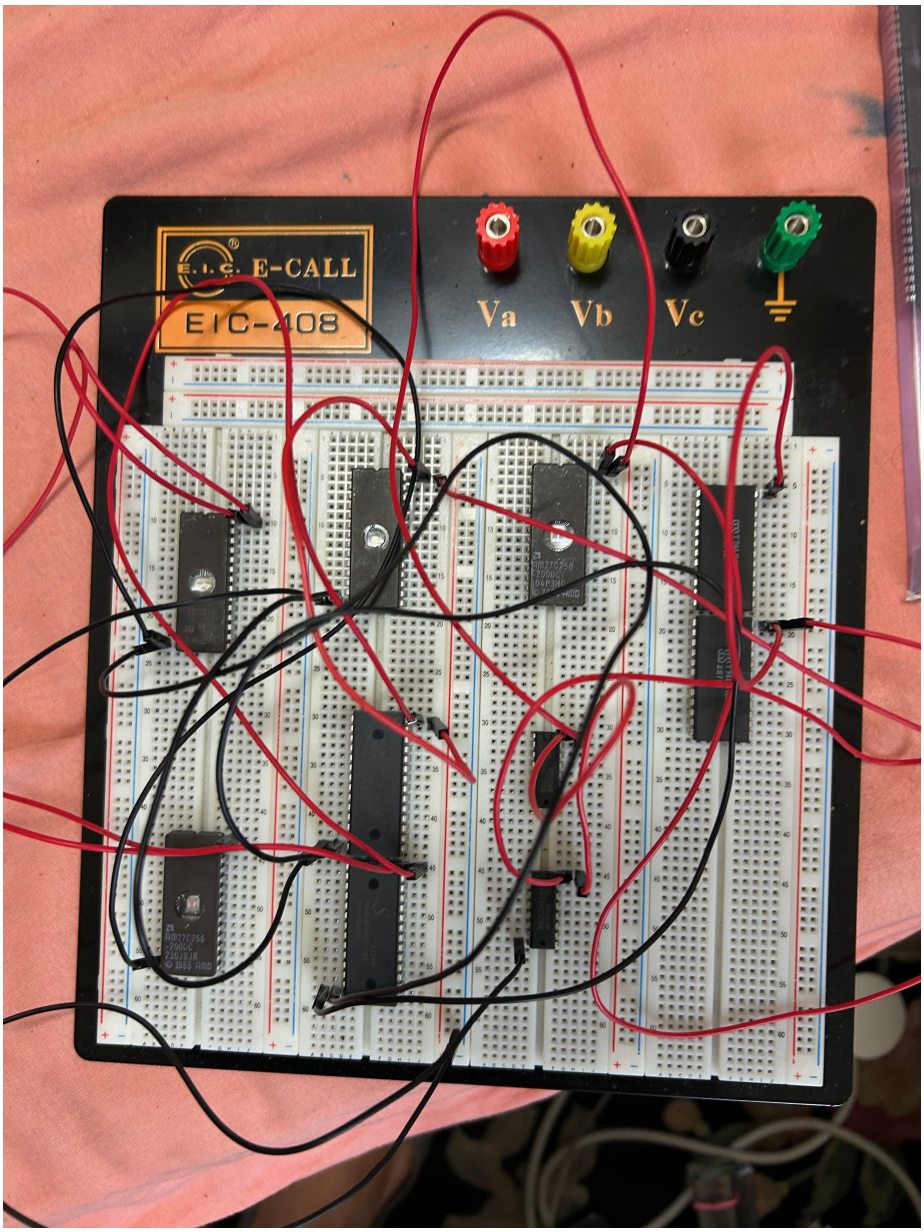



M27C256B
-12F1
B8827
0246L
SINGAPORE


35
30
25
20
15
10
SN74LS161N1
SN74LS161N1


E-CALL
EIC-408
Va
Vb
Vc
Am27C256
-200DC
238JBJR
© 1986 AMD


Vc
Vb
Va
EIC-408


E.I.C. E-CALL
EIC-408
Va
Vb
Vc