

Im Fernsehen sehen Sie im Vordergrund, wie man einen Computer baut. Wir zeigen Ihnen die Hintergründe.

Verknüpfen und Vergleichen

Zum NDR-„Klein“-Computer

In diesem Monat werden die Folgen „Verknüpfungen“, „Gleich riecht er“, „Geschafft: er schwingt“ und „Der Nichtstu-Befehl“ ausgestrahlt. Es wird gleich zu Anfang recht forsch vorgegangen und ein Bogen von einfachen logischen Verknüpfungen bis zur fertigen Computerplatine gezogen. Aber Sie wollen ja möglichst schnell mit dem Computer arbeiten, also muß auch möglichst schnell ein Computer zur Verfügung stehen.

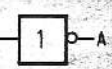



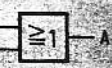







Die Bauanleitungen, Schalt- und Bestückungspläne finden Sie alle im Sonderheft unserer Schwesterzeitschrift mc: „Mikrocomputer Schritt für Schritt“. Daher können wir uns auf aktuelle Probleme stürzen und auf Ihre Fragen eingehen. Übrigens, wenn Sie Fragen oder Tips haben, schreiben Sie uns doch (Franzis-Verlag, ELO-Redaktion, Postfach 37 01 20, 8000 München 37). Tips und Fragen von allgemeinem Interesse nehmen wir in die Artikelreihe auf.

Verknüpfungen

Es kann gar nicht oft genug gesagt werden: Arbeiten Sie beim Aufbau der Schaltungen und Bausätze in aller Ruhe, langsam und sorgfältig. Wie wir aus der Ausstrahlung beim NDR wissen, sind fast alle Fehler beim Aufbau darauf zurückzuführen, daß nicht sorgfältig gelötet wurde – und es wäre doch schade, wenn die ganze Computerei an einer „kalten Lötstelle“ scheitern würde. Computer sind eigentlich ganz einfach aufgebaut: Aus Logik-Gliedern, die es ihm möglich machen, Entscheidungen zu treffen, und aus Speicherelementen, den Flipflops, in denen Informationen aufgebaut werden können. Der Trick an der ganzen Sache ist, daß eine Unmenge von Logik-Gliedern und Flipflops auf kleinstem Raum untergebracht werden können und dann einen Mikroprozessor bilden.

In Bild 1 sind einige Logik-Glieder gezeigt. Wie schon gesagt, wollen wir in der ELO nicht noch einmal die Bauanleitungen aus dem mc-Sonderheft oder dem Buch von R. D. Klein: „Mikrocomputer selbstgebaut“ bringen, sondern die Fernsehreihe durch zusätzliche Informationen unterstützen. Mit den Logik-Gliedern läßt sich schon eine ganze Menge anfangen, zum Beispiel eine „Entscheidungsmaschine“, wie sie in Bild 2 gezeigt ist. Mit einigen von ihnen läßt sich feststellen, ob sich eine mögliche Verhelichung auch lohnt. Leider ist die Schaltung nicht ganz vollständig – wenn die Angebetete nicht will, hilft der ganze „Heiratscomputer“ nichts. Eine wichtige Position nehmen auch die Speicher, die Flipflops, ein. Das Ergebnis eines Gatterbergwerks, wie Sie es in Bild 2 gesehen haben, ändert sich sofort, wenn Sie die Signale an den Eingängen

Bild 1: Das sind einige Logikglieder mit den neuen und den alten Symbolen. Nicht immer sind die „neuen“ glücklich gewählt.

Neue Norm	Alte Norm	Beispiel	Wahrheitstafel																
E  A		7404	<table border="1"> <tr><td>E</td><td>A</td></tr> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </table>	E	A	0	1	1	0	NICHT-Glied									
E	A																		
0	1																		
1	0																		
E ₁  E ₂ A		7408	<table border="1"> <tr><td>E₁</td><td>E₂</td><td>A</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	E ₁	E ₂	A	0	0	0	0	1	0	1	0	0	1	1	1	UND-Glied
E ₁	E ₂	A																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	1																	
E ₁  E ₂ A		7432	<table border="1"> <tr><td>E₁</td><td>E₂</td><td>A</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	E ₁	E ₂	A	0	0	0	0	1	1	1	0	1	1	1	1	ODER-Glied
E ₁	E ₂	A																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	1																	
E ₁  E ₂ A		7486	<table border="1"> <tr><td>E₁</td><td>E₂</td><td>A</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	E ₁	E ₂	A	0	0	0	0	1	1	1	0	1	1	1	0	EXCLUSIV-ODER-Glied
E ₁	E ₂	A																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	0																	
E ₁  E ₂ A		7400	<table border="1"> <tr><td>E₁</td><td>E₂</td><td>A</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	E ₁	E ₂	A	0	0	1	0	1	1	1	0	1	1	1	0	NAND-Glied (NICHT-UND)
E ₁	E ₂	A																	
0	0	1																	
0	1	1																	
1	0	1																	
1	1	0																	
E ₁  E ₂ A		7402	<table border="1"> <tr><td>E₁</td><td>E₂</td><td>A</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	E ₁	E ₂	A	0	0	0	0	1	0	1	0	0	1	1	0	NOR-Glied (NICHT-ODER)
E ₁	E ₂	A																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	0																	

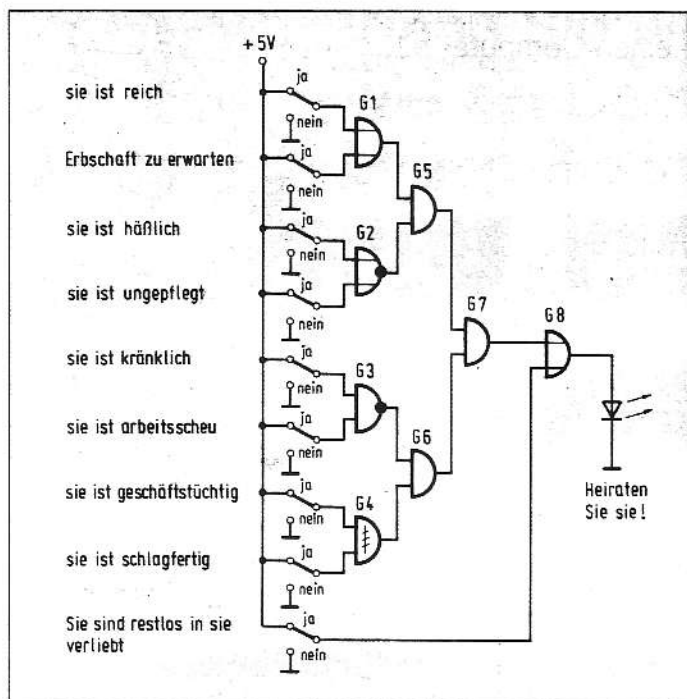


Bild 2: Falls Sie dieser „Entscheidungs-
maschine“ zu sehr
vertrauen, sind Sie
selbst schuld.

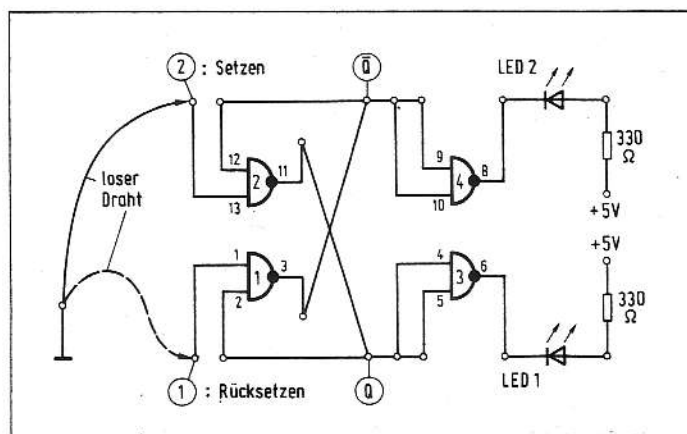


Bild 3: Bei diesem
RC-Flipflop sehen
Sie gleich, auf wel-
cher Seite es steht.

ändern. Sie können es nicht für später aufbewahren – Sie können nichts speichern. Zum Speichern brauchen wir ein Element, das Informationen auch dann behält, wenn die Eingangsdaten nicht mehr da sind. Bauen Sie doch gleich mal aus einem 7400 ein Flipflop auf, wie es in Bild 3 gezeigt ist. Die Logik-Glieder 1 und 2 bilden das eigentliche Flipflop, die beiden anderen Gatter werden nur für die Anzeige gebraucht. Die beiden Gatter sind übers Kreuz verbunden. Wie flippt nun das Flop? Beim Einschalten leuchtet eine der beiden Leuchtdioden rein zufällig. Wenn Sie nun den Draht an den Eingang 1 legen, dann leuchtet in jedem Fall LED 2. Sie haben das Flipflop zurückgesetzt, denn der Ausgang Q, der „Normalausgang“, ist auf 0. Der invertierte Ausgang liegt auf 1. Wenn beide Dioden leuchten oder keine, dann stimmt bei der Schaltung etwas nicht. Wenn Sie jetzt den Draht vom Eingang 1 wegnehmen, geschieht gar nichts. Das Flipflop hat sich den

Zustand gemerkt. Wenn Sie nun mit dem Draht an den Eingang 2 gehen, dann „kippt das Flipflop“ in den gesetzten Zustand, LED 1 leuchtet und LED 2 geht aus. Weiteres Arbeiten an diesem Eingang bleibt ohne Auswirkung auf die Schaltung. Was Sie nun haben, ist ein RS-Flipflop. RS kommt von Reset-Set, englisch für Rücksetzen-Setzen. Bitte überlegen Sie einmal, wie dieses Verhalten durch die Rückkopplung zustande kommt und was passiert, wenn beide Eingänge gleichzeitig auf 0 gelegt werden. Mit einem RS-Flipflop kann man schon eine ganze Menge anfangen, zum Beispiel einen Schalter entprellen. Mechanische Schalter und Taster sind für Logik-Glieder und Computer denkbar schlecht geeignet, weil sie beim Öffnen und Schließen ein wenig klappern und so mehrere 0-1-Impulse liefern. Wie Sie selbst ausprobiert haben, kann man auf dem Eingang eines RS-Flipflops beliebig oft klappern, ohne daß sich nach dem ersten Mal etwas ändert. Bild 4 zeigt

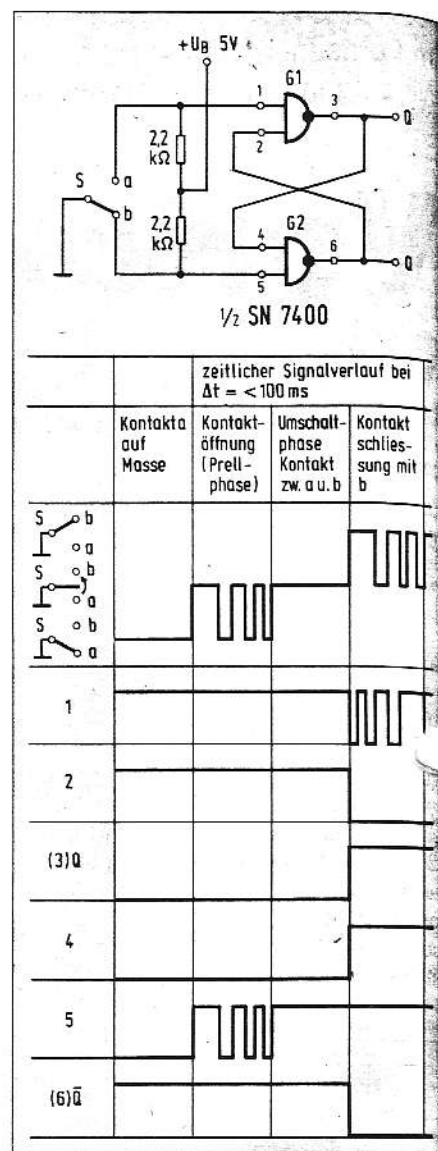


Bild 4: So einfach läßt es sich „entprellen“. Darunter der zeitliche Signalverlauf.

eine solche Entprellschaltung. Später werden Sie sehen, daß man auch per Programm entprellen kann.

Flipflops aller Art

Wie Sie in der Folge „Geschafft: er schwingt“ hören (werden), braucht ein Mikroprozessor einen Takt zum Arbeiten, was nichts anderes bedeutet, als daß bestimmte Aktionen nur zu einem bestimmten Zeitpunkt erfolgen sollen. So ist es nur dann sinnvoll, Daten aus dem Speicher eines Computers in den Prozessor zu holen, wenn der Speicher sie auch richtig bereitstellt. Es ist nämlich so, daß zwischen dem Signal des Prozessors (der CPU) an den Speicher: „Lesen“ oder „Schreiben“ und der Bereitschaft des Speichers eine kurze Zeit vergeht (ca. 100 ns...300 ns). Daten, die in dieser Zeit an den Dateneingängen oder -ausgängen anliegen, sind falsch. Wenn man

dem RS-Flipflop zwei Logik-Glieder vor die Eingänge setzt, wie **Bild 5** zeigt, dann kann sich im Flipflop nur etwas tun, wenn der Takt-Eingang T auf 1 liegt, und so ist das T-Flipflop entstanden. Manchmal braucht man auch ein Flipflop, das keine zwei Eingänge besitzt, wie wir sie beim RS-Flipflop haben. Denken Sie nur an den Datenspeicher beim Computer, wo es für jedes Bit nur eine Datenleitung gibt, auf der entweder 0 oder 1 liegt. Ein „Datenspeicher-Flipflop“, wie es sich mit vielen Kollegen in

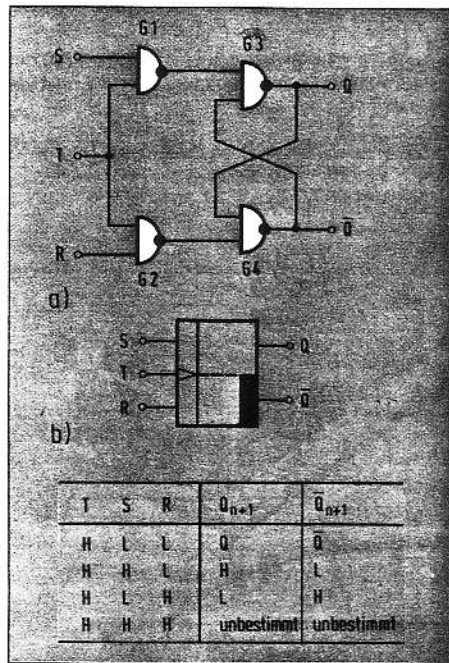
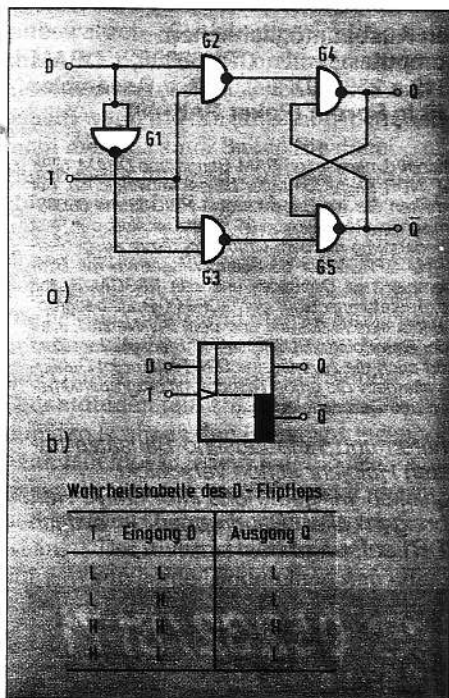


Bild 5: Beweist Takt – das T-Flipflop.

Bild 6: Das letzte der „Statischen“, das D-Flipflop.



einem RAM-Baustein befindet, kann man erhalten, wenn man das T-Flipflop um ein Gatter erweitert, wie es in **Bild 6** gezeigt wird. Damit kennen Sie nun alle statischen Flipflops. „Statisch“ bedeutet, daß das Setzen oder Speichern, wenn überhaupt, dann nur von dem logischen Wert eines Takteingangs abhängt. Für Zähler und Schieberegister (was das ist, kommt gleich) braucht man aber Flipflops, die nur auf den Wechsel eines Signals, nicht aber auf den Zustand reagieren. Man möchte ein Flipflop, das beim ersten Tastendruck auf 1 kippt, beim zweiten auf 0, beim dritten wieder auf 1 und so weiter. Mit solchen dynamischen Flipflops kann man dann zählen.

Bei dynamischen Flipflops macht man es wie in den Firmen, man setzt dem eigentlichen Speicher einen Boß vor die Nase. Darum heißen sie auch Master-Slave-Flipflops (das liebe sich mit Herr- und-Sklave-Flipflop übersetzen). In **Bild 7** ist so ein MS-Flipflop gezeigt. Die Eingänge J und K bestimmen, was beim nächsten Taktimpuls geschieht. Der Ablauf ist etwas kompliziert, daher zeigen wir es ganz langsam:

1. Der Takt beginnt mit einer 0-1-Flanke. Durch den Taktinverter werden die Eingänge des Hauptspeichers gesperrt (G6 und G7 sorgen dafür).
2. Hat der Impuls dann den 1-Pegel er-

reicht, wird die Information von den Eingängen J und K in den Master übernommen.

3. Fällt die Impulsflanke wieder, werden die Eingänge wieder gesperrt, wenn der 1-Pegel unterschritten ist, der Master speichert die Information.

4. Hat die Taktflanke den 0-Pegel erreicht, wird die Information an den Slave weitergegeben. Durch die Rückkopplung der Slave-Ausgänge auf den Master wird zudem immer ein Eingangsgatter gesperrt, so daß auch bei gleichem Signal auf J und K ein eindeutiger Ausgangszustand erzeugt wird.

Zähler und Schieberegister

Wenn Sie den Takteingang eines MS-Flipflops mit dem Ausgang des vorhergehenden verbinden, erhalten Sie einen Dualzähler. Die Eingänge J und K müssen dabei auf 1 liegen. **Bild 8** zeigt einen 4-Bit-Zähler, mit dem man von 0 bis 15 zählen kann. Wie die Ausgänge beim Zählen aussehen, zeigt das Impulsdigramm in **Bild 8**. So einen 4-Bit-Zähler gibt es als fertigen Baustein zu kaufen, er heißt 7493. Mit den Zählern kann man dann Uhren, Frequenzzähler, Taktgeneratoren und Schieberegister bauen. Beim Schieberegister wird nicht der

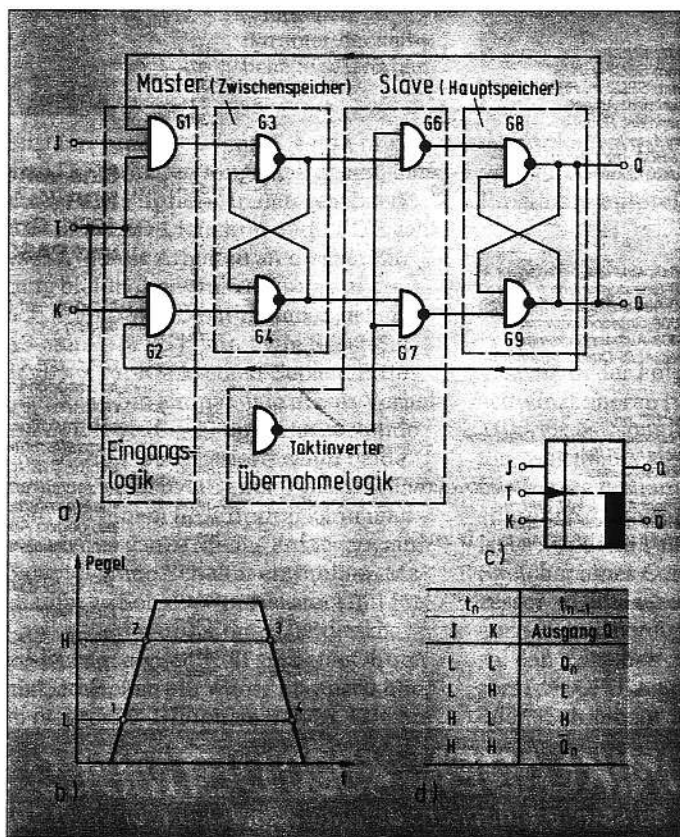


Bild 7: Herr und Sklave – nicht gerade vornehm, aber überdeutlich.

Takteingang eines Flipflops mit dem Ausgang des vorhergehenden verbunden, sondern die Eingänge mit den Ausgängen. Der Takt liegt bei allen Flipflops an der selben Leitung. **Bild 9** zeigt ein Schieberegister mit vier D-Flipflops und das Impulsdiagramm dazu.

Decoder und Vergleicher

Gleich bei der SBC-II-Karte tauchen sie auf, und sie sind auf fast jeder Karte des

NDR-Klein-Computers zu finden, die Adreßdecoder 74 LS 138 und die Vergleicher 74 LS 85. Hier soll ein wenig beleuchtet werden, was in diesen Bausteinen so vorgeht. Beide Bausteine enthalten jede Menge von Logik-Gliedern, die in ihrem Zusammenwirken eben die komplexen Funktionen liefern. Ein Decoder liefert für jede Dualzahl am Eingang an einem bestimmten Ausgang ein Signal. Bei einem Bit hat er zwei Ausgänge, 0 und 1, bei zwei Bit vier (0...3), bei drei Bit acht (0...7) und bei vier Bit

schon 16. Allgemein hat ein n-Bit-Decoder 2 hoch n Ausgangsleitungen. In **Bild 10** ist die Schaltung eines Decoders für zwei Bit gezeichnet. Zum Glück gibt es die Decoder als fertige Bausteine, sonst würden die Platinen aus allen Nähten platzen. **Bild 11** zeigt, wie man mit einem Zähler und einem 4-Bit-Decoder eine Lauflichtsteuerung realisieren kann. Auf den Computerplatinen sorgen die Decoder dafür, daß, abhängig von den Adressen, die jeweils gewünschten RAM-, EPROM- oder ROM-Bausteine ausgewählt werden. In den RAM-, EPROM- oder ROM-Bausteinen befinden sich natürlich auch Decoder, die entsprechend den Adreß-Signalen die richtige Speicherzelle auswählen. Will man die Adresse einer Karte beliebig umstellen, braucht man einen Vergleicher-Baustein, der die Adreßbits vom Prozessor mit den auf der Karte eingestellten Adreßbits vergleicht. **Bild 12** zeigt Vergleicher für ein und zwei Bits. Der Baustein 74 LS 85 enthält einen 4-Bit-Vergleicher, der auch noch Ausgänge für $A < B$ und $A > B$ enthält – wieder jede Menge Logik-Glieder. Da die Vergleichereingänge im offenen Zustand über Widerstände auf 1 gelegt werden, muß man Brücken für diejenigen Bits anbringen, die 0 sein sollen. Diese – gewissermaßen invertierte – Einstellung führt manchmal zu Fehlern. Man kann auf einer Karte auch Decodierer und Vergleicher sparen, indem man die

Bild 8: Mit vier MS-Flipflops können Sie bereits von 0 bis 15 zählen.

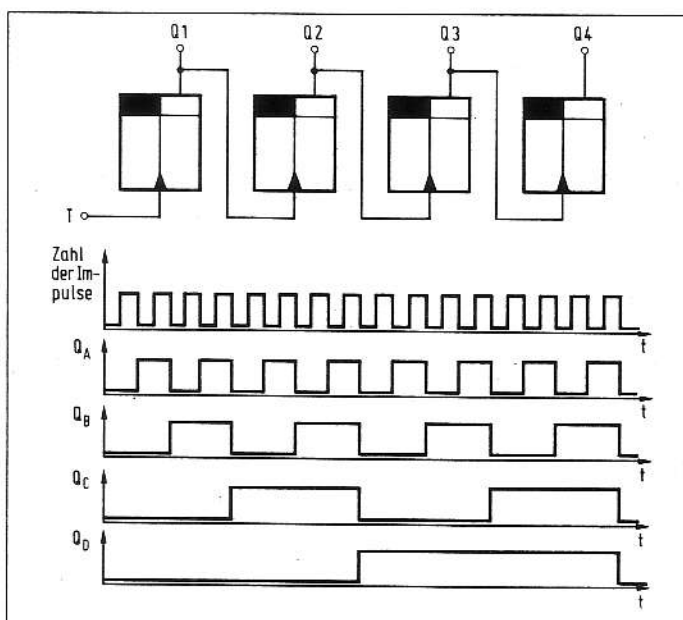


Bild 9: Und hier wird geschoben – ganz legal.

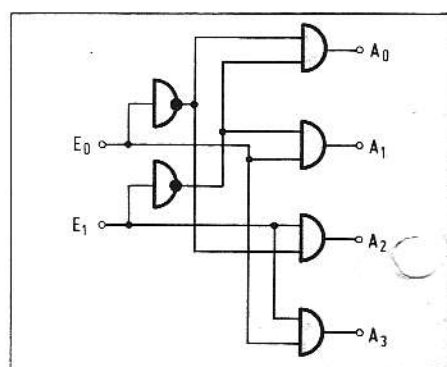
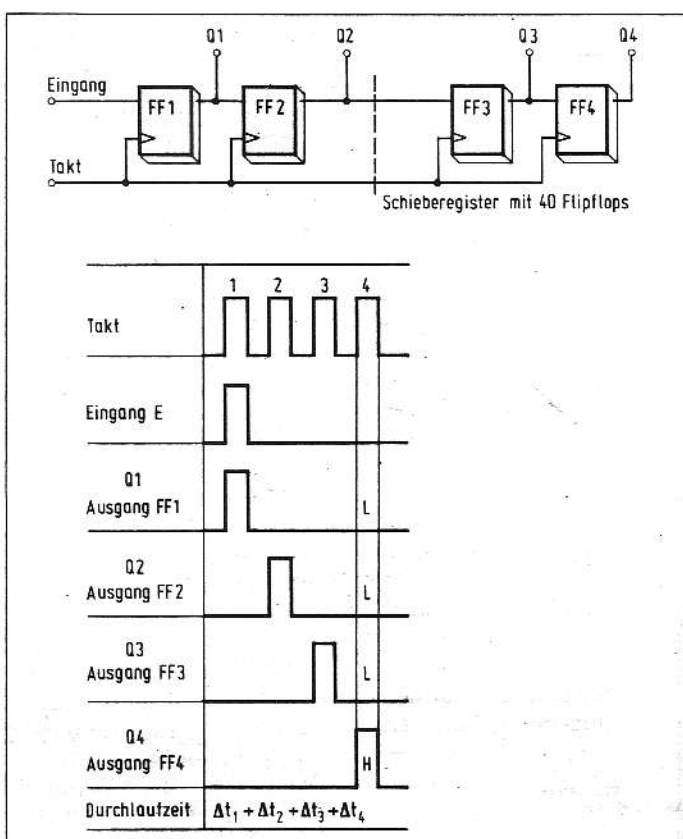


Bild 10: Diese Schaltung decodiert 2 Bits.

Adressen nicht voll decodiert. Dann taucht ein und dieselbe Karte an mehreren Stellen im Adreßraum auf, was so lange nichts macht, solange man an diesen doppelt belegten Adressen keine anderen Speicher oder Peripheriekarten unterbringt.

Druckfehler

Leider haben sich im Begleitbuch „Mikrocomputer selbstgebaut und program-

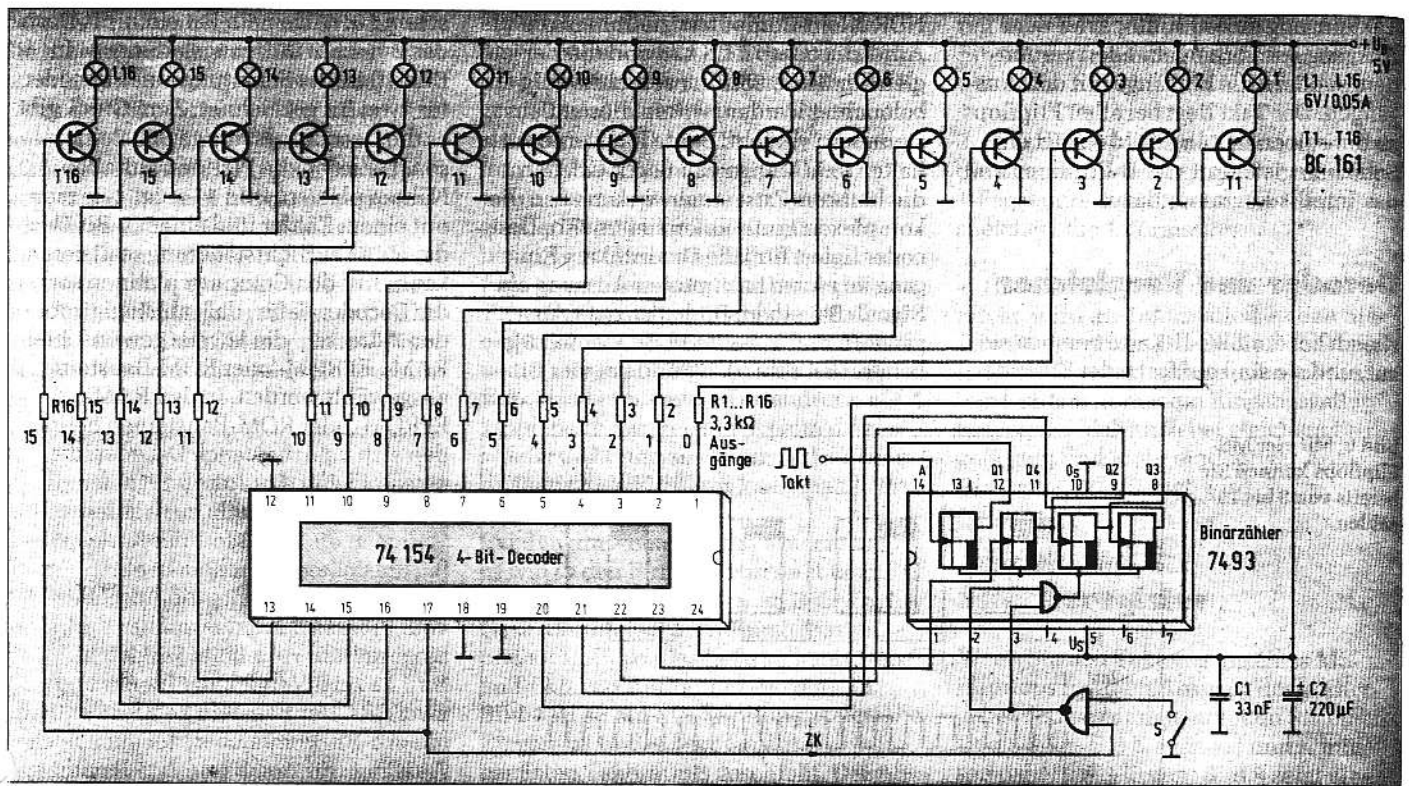


Bild 11: Und hier steuert ein 4-Bit-Decoder ein Lauflicht.

miert“, das übrigens schon lange vor der Fernsehreihe auf dem Markt war und daher in Reihenfolge und einigen Layouts nicht mit den Bausätzen übereinstimmt, einige Druckfehler eingeschlichen. Hier nun die Korrekturen zur zweiten Auflage.

S. 25/Abb. 2.3.3: 7406 anstatt 7606.

S. 27/Abb. 2.3.7: Die Steuereingänge am 74 245 sind nicht invertiert.

S. 81/Abb. 4.5.13: Der Ausgang 7 des 74 LS 138 liegt an Pin 7 und nicht wie gezeichnet an Pin 8.

S. 86/Abb. 4.6.1: Die Gatter 17 und 18 sind aus dem 7400 und nicht aus dem 7404.

S. 91/Abb. 4.7.1: IC B1 Pin 1 ist DIR, Pin 19 ist Enable.

S. 104/Zeile 6: C3 1223.

S. 112/Zeile 13: ...Leitung PHI2 (Pin 27)...

S. 124/Abb. 5.2.1: Der Widerstand gegen +5 V ganz oben sollte statt 1 kΩ einen Wert von 3,3 kΩ haben, wie der Bestückungsplan angibt.

S. 136/Abb. 5.3.9: Bezeichnungen J1 und J2 sind vertauscht. Beim 2716 sind die Bezeichnungen CS und OE vertauscht. Eingang D (Pin 6) von Z1 muß an +5 V liegen. Bei Gen1 ist der Masseanschluß Pin 7 (nicht 8) und der +5-V-Anschluß Pin 14 (nicht 16). R1 = 390 Ω, R5 = 390 Ω.

S. 161/Abb. 5.3.28: Der Kondensator mit 10 nF links oben ist falsch gepolt, der Pluspol muß nach rechts zeigen.

S. 165/letzte Zeile: Streichen.

S. 166/Zeile 2: Anfügen: ...ergeben sich 4 Bildseiten 256 × 512 Bildpunkte...

S. 167/Abb. 5.3.34: Unter b4 b3: delta y, bei Dimension: delta x or delta y.

S. 171: EF 9366 statt EF 0366.

S. 185/Zeile 12: e1 statt e0, e2 statt e1.

S. 191/Abb. 5.6.7: R10 hat 1 kΩ statt 10 kΩ.

S. 324 ff.: Bei Grundprogramm Version 1.2 arbeiten die Funktionen HEBE und SENKE nicht korrekt. Im Handel ist Version 2.0, die korrekt arbeitet; anstelle „T=Turtle“ erscheint „F=Flip“.

S. 373/Abb. 7.4.3: Anstelle 74 LS 327 nimmt man besser 74 LS 627. Poti Mot2 geht an Pin 12, Poti Mot1 an Pin 2 des 74 LS 627.

S. 374/Abb. 7.4.4: Statt 1 N 2002 heißt es 1 N 2004. Die Diode vom Kollektor des obersten Transistors TIP 120 muß an +Spg.Motor und nicht an +5 V liegen.

S. 406: Die Firma Albatros existiert nicht.

Jürgen Plate

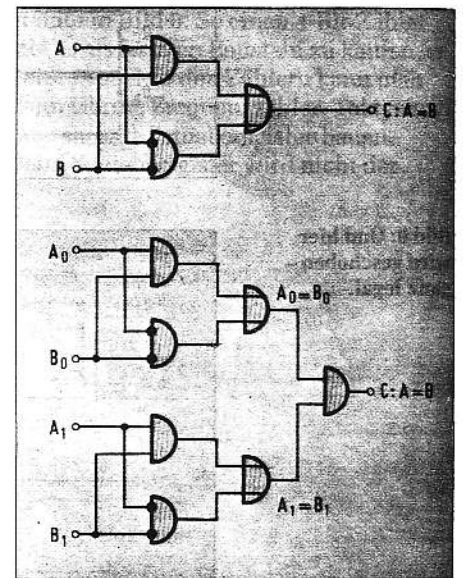


Bild 12: Im Endausbau identifizieren sie Fingerabdrücke! Diese beiden Vergleicher erkennen nur 1 Bit (oben) und 2 Bits.

Durch einen bedauerlichen Irrtum wurde in Heft 9 nur ein Bausatzlieferant genannt. Wir bitten um Entschuldigung!

Derzeit sind die Platinen und Bausätze bei folgenden Stellen erhältlich:

Hauptlizenznehmer:
Graf Elektronik Systeme GmbH
Magnusstr. 13
Postfach 1610
8960 Kempten
Tel.: (08 31) 62 11
Untertizenznehmer:
Elektronikladen Detmold
Eggestr. 70
4930 Detmold 18
Tel.: (0 52 32) 81 31