

Chemische Digitalcomputer

Modell: Frequenzteiler 2:1; 3:1; 4:1

Friedrich-Schiller-Universität Jena

Erik Siebert, Eric Brzoska

Jena, 10.07.2024



**FRIEDRICH-SCHILLER-
UNIVERSITÄT
JENA**

Professor: Dr. Thomas Hinze

Modul: Molekulare Algorithmen – FMI-BI0050

Inhaltsverzeichnis

INHALTSVERZEICHNIS	II
ABBILDUNGSVERZEICHNIS	III
TABELLENVERZEICHNIS	IV
1. EINLEITUNG.....	1
1.1. AUFGABENSTELLUNG.....	1
2. KONSTRUKTION DER GATTERSCHALTUNGEN	2
2.0. VORÜBERLEGUNGEN	2
2.1. BESTANDTEILE UNSERER GATTERSCHALTUNGEN	3
2.1.1. <i>Gatterschaltung für 2:1 und 4:1 Frequenzteiler</i>	4
2.1.2. <i>Gatterschaltung für 3:1 Frequenzteiler</i>	6
3. COPASI IMPLEMENTIERUNG.....	10
3.1. ALLGEMEINES SETUP DES CHEMISCHEN DIGITALCOMPUTERS INNERHALB DER COPASI UMGEBUNG	10
3.2. DETAILLIERTE IMPLEMENTIERUNG DER SCHALTPLÄNE	10
4. SIMULATION UND RESULTATE.....	12
5. FAZIT.....	14
LITERATURVERZEICHNIS.....	V

Abbildungsverzeichnis

ABBILDUNG 1. VISUELLE REPRÄSENTATION DES EINGABESIGNALES, SOWIE DER DREI AUSGABESIGNALE DER JEWELIGEN ZU KONSTRUIERENDEN FREQUENZTEILER.	2
ABBILDUNG 2. VERALLGEMEINERTER SCHEMATISCHER AUFBAU DER LOGISCHEN GATTERSCHALTUNG FÜR DIE FREQUENZTEILER.....	3
ABBILDUNG 3. KARNAUGH MAPS FÜR DIE BIT-SIGNAL UMWANDLUNGSKOMPONENTEN B'1 , B'2 , UND B'3 , SOWIE DIE OUTPUT KOMPONENTEN FÜR DIE FREQUENZTEILER 2:1 UND 4:1.....	6
ABBILDUNG 4. KARNAUGH MAPS FÜR DIE BIT-SIGNAL UMWANDLUNGSKOMPONENTEN B'1 , B'2 , UND B'3 , SOWIE DIE OUTPUT KOMPONENTE FÜR DEN FREQUENZTEILER 3:1.	7
ABBILDUNG 5. VOLLSTÄNDIGE LOGISCHE GATTERSCHALTUNG FÜR DIE FREQUENZTEILER 2:1 UND 4:1.....	8
ABBILDUNG 6. VOLLSTÄNDIGE LOGISCHE GATTERSCHALTUNG FÜR DEN FREQUENZTEILER 3:1.	9
ABBILDUNG 7. SIMULATION DES 2:1 FREQUENZTEILER OUTPUTS.	12
ABBILDUNG 8. SIMULATION DES 3:1 FREQUENZTEILER OUTPUTS.	13
ABBILDUNG 9. SIMULATION DES 4:1 FREQUENZTEILER OUTPUTS.	13

Tabellenverzeichnis

TABELLE 1. ZUSTANDSÜBERGANGSTABELLE FÜR DIE FREQUENZTEILER 2:1 UND 4:1.	5
TABELLE 2. KONSTRUIERTE SCHALTKREISKOMPONENTEN UND BOOLESCHE SCHALTFUNKTIONEN FÜR DIE FREQUENZTEILER 2:1 UND 4:1.	6
TABELLE 3. ZUSTANDSÜBERGANGSTABELLE FÜR DEN FREQUENZTEILER 3:1.....	7
TABELLE 4. KONSTRUIERTE SCHALTKREISKOMPONENTEN UND DAZUGEHÖRIGE BOOLESCHE SCHALTFUNKTIONEN FÜR DEN FREQUENZTEILER 3:1.	7

1. Einleitung

Die Bioinformatik, eine interdisziplinäre Wissenschaft, setzt seit einigen Jahrzehnten computer- und Algorithmus-gestützte Methoden ein, um Probleme in den biologischen Wissenschaften zu lösen. Historisch gesehen hat die Bioinformatik bereits wesentliche Beiträge zur modernen Biologie und Medizin geleistet. Angesichts der zahlreichen Herausforderungen die innerhalb der Biologie emergieren, nutzt die Bioinformatik eine große Varianz an Rechenwerkzeugen zur Bewältigung dieser. Mit steigender Komplexität der zu lösenden Aufgaben aus immer spezifischeren biologischen Fachbereichen haben sich im Laufe der Zeit mehrere spezialisierte Bereiche der Bioinformatik entwickelt. Ein aufsteigendes Beispiel hierfür ist der Fachbereich der molekularen Bioinformatik. Dieser Bereich umfasst Themen wie molekulare Algorithmen, durch welche die technischen Prinzipien von Computern in chemischen Modellen abbilden werden.

Durch die Anwendung molekularer Algorithmen und computergestützter Analysemethoden konnten beispielsweise neue Therapien und Diagnosetools in der Medizin entwickelt werden. Diese Ansätze haben insbesondere in der Krebsforschung und bei der Entwicklung personalisierter Behandlungsstrategien zu neuen Erkenntnissen geführt. Insgesamt zeigt die Entwicklung der Bioinformatik, wie entscheidend die Integration von Computerwissenschaften in die Biowissenschaften ist. Sie bietet innovative Lösungen für komplexe biologische Probleme und treibt die wissenschaftliche Forschung kontinuierlich voran. ¹

Das Fachgebiet der molekularen Algorithmen versucht chemische Modelle zu nutzen, um verschiedene technische Konzepte von Computern darzustellen. Ein entscheidender Beweggrund ist die Überlegenheit die biologischer Systeme gegenüber konventioneller Rechentechnik bieten. Dies soll als Grundlage dienen, um mittels dieser Hausarbeit einen deterministischen, endlichen Automaten als chemischen Digitalcomputer zu implementieren. ²

1.1. Aufgabenstellung

Die Aufgabe dieser Projektarbeit besteht grundlegend darin, erfolgreich einen Frequenzteiler als Chemisches Digitalcomputermodell in der chemischen Simulationssoftware COPASI zu implementieren und mit Hilfe dieser Implementierung einen 2:1, 3:1, und 4:1 Frequenzteiler zu simulieren. Ein annähernd rechteckiges Taktsignal soll dem Frequenzteiler als Eingabe dienen, wobei dieser dieses eine Eingabesignal in drei individuelle ebenfalls annähernd rechteckige Ausgabesignale umwandeln soll. Jedes der drei Ausgabesignale repräsentiert eine entsprechende

¹ (Wikipedia - Bioinformatik, 2024)

² (Hinze, Computer der Natur, 2013)

Frequenzteilung des Eingabetaktsignales, wobei je nach Frequenzteiler der logische Wert des Ausgabesignales aller zwei, drei, oder vier Taktflanken zwischen 0 und 1 wechselt. Zur Verdeutlichung der zu realisierenden Ausgabesignale dient **Abbildung 1**.

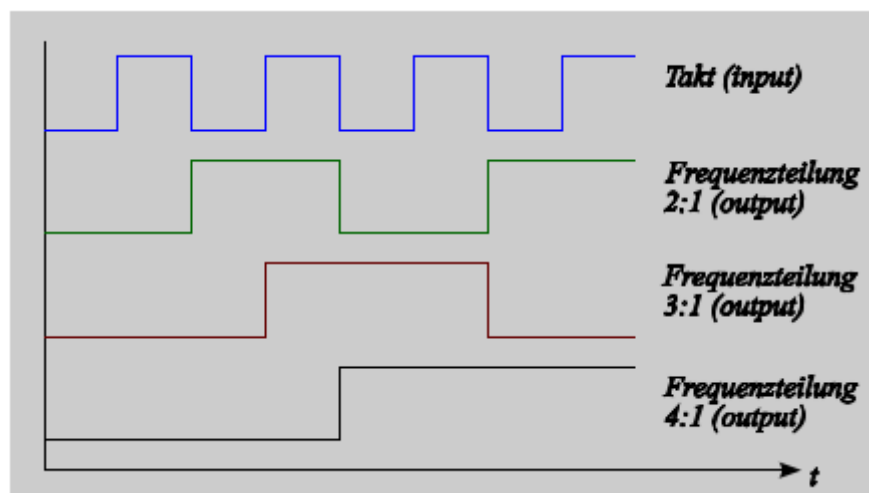


Abbildung 1. Visuelle Repräsentation des Eingabesignales, sowie der drei Ausgabesignale der jeweiligen zu konstruierenden Frequenzteiler. **Quelle:** Vorlesung: Molekulare Algorithmen - SS 2024; Dr. Thomas Hinze.

Für die erfolgreiche Umsetzung der Frequenzteiler wird für jeden dieser eine entsprechende logische Frequenzteiler-Gatter-Schaltung konstruiert, welche dann jeweils in ein chemisches Digitalcomputermodell überführt wird, und als chemisches Modell in COPASI implementiert wird. Abschließend wird eine Simulationsstudie durchgeführt, bei welcher über mind. 30 Eingabetakte hinweg die Funktionsfähigkeit der jeweiligen chemischen Modelle demonstriert werden soll. Im bevorstehenden Kapitel folgt eine detaillierte Ausführung zu dem Prozess der Konstruktion der benötigten Schaltkreise.

2. Konstruktion der Gatterschaltungen

2.0. Vorüberlegungen

Für jeweils jeden der drei Frequenzteiler wird grundlegend eine individuelle logische Gatterschaltung benötigt, welche sich dennoch zu einem gewissen Grade mit den Schaltungen der anderen Teiler gleichen können bzw. überschneiden dürfen, wenn dadurch die insgesamt Komplexität und Größe aller Gatterschaltungen auf ein effizientes Minimum gebracht werden. Eine genauere Erklärung für diese Herangehensweise und den daraus folgenden Implikationen folgt später im Kapitel.

Einer der ersten Schritte für die erfolgreiche Umsetzung der Frequenzteiler als logische Gatterschaltung, ist die konkrete Ausformulierung des umzusetzenden Konzeptes. Anders formuliert, der erste Schritt besteht aus der Planung der logischen Schaltung, also welche Komponenten in der Schaltung enthalten sein sollen, und welche Inputs und

Outputs in die Schaltung rein- und/oder rausfließen. Sobald Komponenten, Inputs, Outputs, und Struktur der Schaltung definiert wurden, wird für jede Schaltkomponente, bei der es notwendig ist, eine boolesche Schaltfunktion erstellt, die als Vorlage für die Anordnung und Nutzung der einzelnen Logikkomponente dienen wird. Die Erstellung dieser Schaltfunktionen erfolgt in dieser Arbeit mittels Karnaugh Maps, welche in einem späteren Abschnitt ebenfalls genauer beleuchtet werden. Grundlegend helfen diese dabei die effizienteste Gatterschaltung für mehrere zusammenhängende Zustandsabhängige Inputs und Outputs zu konstruieren. Dies ist ein essenzieller Schritt, da für die spätere Umsetzung der Schaltung in COPASI ein maximal effizienter und optimaler Schaltkreis benötigt wird. Andernfalls steigt die Quantität der einzelnen chemischen Spezies und deren korrelierenden Reaktionen in das unermessliche bei dem Versuch einen unnötig komplexen Schaltkreis in ein chemisches Reaktionsnetzwerk umzusetzen.

2.1. Bestandteile unserer Gatterschaltungen

In diesem Unter Kapitel wollen wir uns auf die konkrete Ausformulierung der benötigten Inputs, Outputs, und Schaltungskomponenten unserer Frequenzteiler, sowie deren Umsetzung mittels Zustandsübergangstabellen, Booleschen Schaltfunktionen und Karnaugh Maps fokussieren.

Um einen Frequenzteiler zu realisieren, welcher ein eingehendes Taktsignal in ein entweder um zwei, drei, oder vier Takte verzögertes Ausgabetaktsignal umwandelt, haben wir uns dafür entschieden den Ansatz eines internen RAM-Speichers zu wählen. Dieser soll in der Lage sein 3-Bit an Information und somit insgesamt bis zu acht

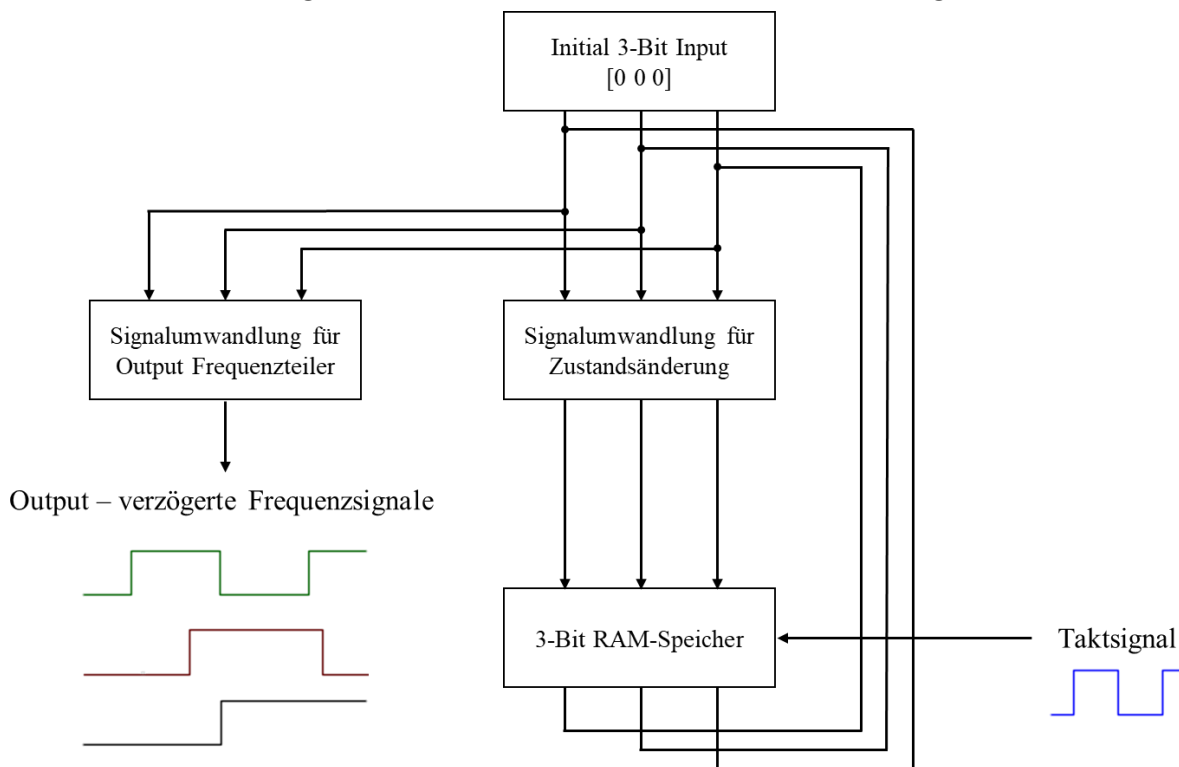


Abbildung 2. Verallgemeinerter schematischer Aufbau der logischen Gatterschaltung für die Frequenzteiler.

Zustände kurzzeitig zu speichern. Damit ist es möglich mit jeder Änderung des Taktsignales innerhalb unserer Schaltung zwischen diesen acht Zuständen zu wechseln, oder anders formuliert, es ist uns möglich von 0 bis 8 zu zählen. Um einen 3-Bit RAM-Speicher in eine Schaltung zu realisieren, benötigt die Schaltung drei Inputs, welche sich mit Wechsel des Taktsignales ebenfalls ändern, damit der RAM-Speicher parallel von 0 bis 8 hochzählt. Die für den RAM-Speicher notwendigen drei 1-Bit Speicherelemente werden mittels RS-Flip-Flop Schaltkomponenten umgesetzt. Für jedes dieser drei Speicherelemente ist die Konstruktion einer Gatterschaltung notwendig, welche je nach Zuständen der drei Inputs einen bestimmten Output liefert, welcher daraufhin in dem Speicherelement zwischengelagert wird.

Der Taktgenerator wird nach dem Vorbild aus der Vorlesung Molekulare Algorithmen von Dr. Hinze, welche zu großen Teilen auf seinem Buch „Computer der Natur“ basiert, implementiert.

Abschließend werden noch Output-Komponenten konstruiert, welche je nach Kombination der drei Inputs das zu erzielende verzögerte 2:1, 3:1, oder 4:1 Signal ausgeben sollen.

Zum besseren Verständnis der gesamten Schaltungen und ihrer Komponenten ist in **Abbildung 2** ein allgemeiner schematischer Aufbau der Schaltungen vorzufinden, welcher die Funktionsweise hervorheben soll.

2.1.1. Gatterschaltung für 2:1 und 4:1 Frequenzteiler

Die Konstruktion der Gatterschaltung für die beiden Frequenzteiler 2:1 und 4:1 bietet den enormen Vorteil, dass für beide dieser Frequenzteiler die gleiche Gatterschaltung verwendet werden kann und sich ausschließlich die Schaltungskomponenten unterscheiden, welche die Input Signale in den jeweiligen 2:1 und 4:1 Taktoutput umwandeln. Der Grund dafür liegt in der Natur der beiden Zahlen 2 und 4. Da 2 ein Teiler von 4 ist, lässt sich diese mit der 2 ohne Rest teilen. Betrachten wir nun dieses Verhältnis in Bezug auf die Schaltung. Die ersten vier Zustände bzw. die ersten vier Taktänderungen würden bei dem Teiler 4:1 zu einem Output von 0 führen. Dementsprechend würden die darauffolgenden vier Zustände zu einem Output von 1 führen, und damit der Frequenzteiler 4:1 innerhalb eines Durchlaufes der 8 Zustände genau einmal ein Zyklus seines Taktsignales ausgeben. Der Frequenzteiler 2:1 benötigt nur die Hälfte an Zuständen, um insgesamt genau einmal einen Zyklus seines Taktsignales zu durchlaufen. Demnach würde dieser zwei Zyklen durchlaufen, während der 4:1 Teiler einen durchläuft. Abschließend würden beide Teiler nach dem Reset auf den Ausgangszustand, welcher nach der achten Zustandsänderung durchgeführt wird, wieder bei ihrem Ausgangszustand, also ihr initialer Output von 0, beginnen.

Diese Form der Synchronisierung existiert nicht zwischen dem Frequenzteiler 3:1 und 2:1 und/oder 4:1. Ein Zyklus des 3:1 Teilers benötigt insgesamt sechs

Zustandsänderungen und würde demnach, folgend auf den sechsten Zustand, auf seinen Ausgangszustand zurücksetzen müssen. Würde man nun versuchen die gleichen Schaltelemente für alle drei Frequenzteiler oder eine Kombination aus dem 3:1 und 2:1 und/oder 4:1 Teiler zu verwenden, dann würde diese asynchron sein. Dies würde zur Folge haben, dass die Gatterschaltung für eine der beiden Teiler Gruppen nicht ordentlich schaltet. Somit betrachten wir als erstes die Konstruktion der gemeinsamen Gatterschaltung für die Frequenzteiler 2:1 und 4:1 und die des Frequenzteilers 3:1 darauffolgend im Einzelnen.

Um die möglichst effizienteste Schaltung für die Teiler aufzustellen, wird als erstes eine Zustandsübergangstabelle aufgestellt. Diese enthält als Zeilen die verschiedenen Zustände, zwischen denen gewechselt wird, und als Spalten die unterschiedlichen Inputs und Outputs, welche sich in Zusammenhang der Änderung der Zustände ebenfalls ändern. Für die erste Schaltung enthält diese Tabelle 8 Zustände als Zeilen, die drei Input Channels **b1**, **b2**, und **b3**, die drei Bit-Signal Umwandlungselemente **b'1**, **b'2**, und **b'3** (diese erhalten jeweils das Input Signal aller drei Input Channels des aktuellen Zustandes und wandeln dieses in das Input Signal für den neuen Zustand um), und das Output Signal für den jeweiligen Frequenzteiler. (**Tabelle 1**)

Zählwert	b1	b2	b3	b'1	b'2	b'3	2:1	4:1
1	0	0	0	0	0	1	0	0
2	0	0	1	0	1	1	0	0
3	0	1	1	0	1	0	1	0
4	0	1	0	1	1	0	1	0
5	1	1	0	1	1	1	0	1
6	1	1	1	1	0	1	0	1
7	1	0	1	1	0	0	1	1
8	1	0	0	0	0	0	1	1

Tabelle 1. Zustandsübergangstabelle für die Frequenzteiler 2:1 und 4:1. Dargestellt sind die Binärrepräsentation der Zustandsübergänge für die jeweiligen Input Bits **b1**, **b2**, und **b3**, die Output Bits der Bit-Signal Umwandlungskomponenten **b'1**, **b'2**, und **b'3**, und der Output Bits der zwei Frequenzteiler 2:1 und 4:1. Alle Input und Output Bits werden über einen Zyklus von 8 Zustandsänderungen dargestellt. Verwendung eines 3-Bit-Gray-Code, welcher je Zustandsänderung nur jeweils 1 Bit ändert.

Ein wichtiges Detail bei der Aufstellung der Zustandsübergangstabelle ist die Verwendung des Bit-Gray-Codes, welcher dafür sorgt, dass sich betrachtet auf die Bit-Kombination der drei Input Channels jeweils nur 1 Bit je Zustandsübergang ändert. Diese Herangehensweise ist eine Voraussetzung für die darauffolgende Nutzung der Zustandsübergangstabelle für Karnaugh Maps. Nutzt man den Bit-Gray-Code nicht, dann resultieren die darauf aufbauenden Karnaugh Maps auch nicht in den effizientesten Logikschaltungen. Die aufgestellten Karnaugh Maps für die drei Bit-Signal Umwandlungskomponenten und die Output Komponenten der 2:1 und 4:1 Teiler sind in **Abbildung 3** dargestellt.

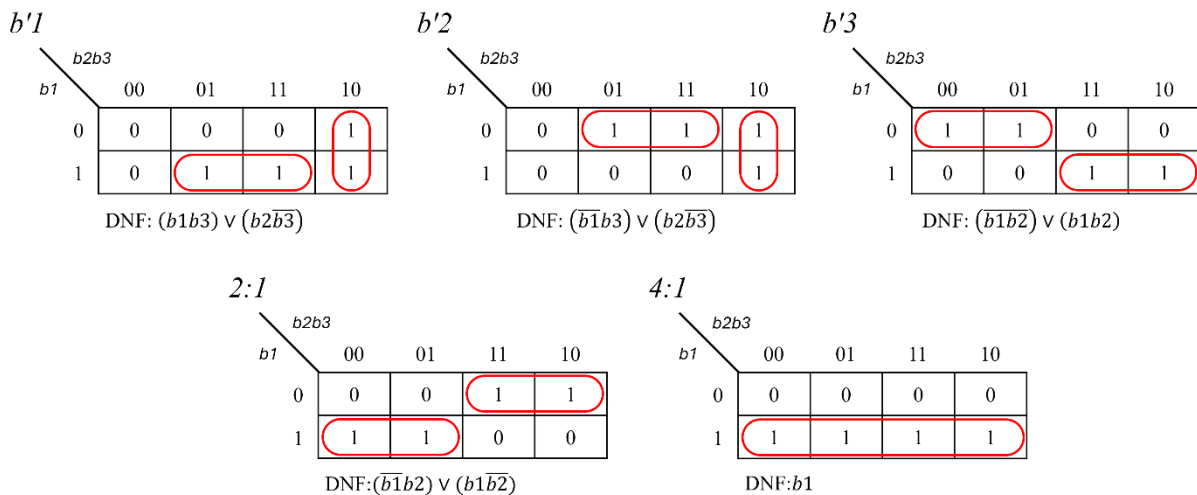


Abbildung 3. Karnaugh Maps für die Bit-Signal Umwandlungskomponenten $b'1$, $b'2$, und $b'3$, sowie die Output Komponenten für die Frequenzteiler 2:1 und 4:1. Eine einzelne Karnaugh Map repräsentiert eine alternative Schreibweise zu der Zustandsübergangstabelle, welche die jeweilig gewünschten Outputs zu jeder möglichen Kombination an Inputs der Input Channels wiedergibt. Durch Gruppierung (rote Umrandung) der high voltage Outputs (Binär 1) lassen sich boolesche Schaltfunktionen aufstellen, welche die effizienteste logische Schaltung für die Kombination aus den jeweiligen Inputs mit dem bestimmten Output widerspiegelt.

Sobald die Karnaugh Maps für alle in der Schaltung enthaltenen Komponenten aufgestellt wurden, liest man aus diesen die booleschen Schaltfunktionen ab. Diese geben Auskunft, welche Logik Gatter die verschiedenen Inputs miteinander kombiniert werden müssen, damit man die gewünschten Outputs erhält. Die jeweiligen booleschen Schaltfunktionen für die Frequenzteiler 2:1 und 4:1 sind übersichtlich in **Tabelle 2** vorzufinden.

Schaltkreiskomponente	Boolesche Schaltfunktion
$b'1$	$(b1b2) \vee (b1b3) \vee (b2\bar{b}3)$
$b'2$	$(\bar{b}1b3) \vee (\bar{b}1b2) \vee (b2\bar{b}3)$
$b'3$	$(\bar{b}1\bar{b}2) \vee (b1b2)$
Out_2:1	$(\bar{b}1b2) \vee (b1\bar{b}2)$
Out_4:1	$b1$

Tabelle 2. Konstruierte Schaltkreiskomponenten und boolesche Schaltfunktionen für die Frequenzteiler 2:1 und 4:1. Die booleschen Schaltfunktionen sind aus den Karnaugh Maps aus **Abbildung 3** abgeleitet.

Die aus den booleschen Schaltfunktionen abgeleitete vollständige Logik Gatterschaltung für die Frequenzteiler 2:1 und 4:1 ist der **Abbildung 5** zu entnehmen.³

2.1.2. Gatterschaltung für 3:1 Frequenzteiler

Die Konstruktion der Gatterschaltung für den Frequenzteiler 3:1 folgt dem gleichen Prinzip wie die für die vorherigen 2:1 und 4:1 Teiler. Als erstes wird die Zustandsübergangstabelle aufgestellt. (**Tabelle 3**) Diese enthält nur 6 verschiedene Zustände, da ein Zyklus des 3:1 Teilers nach dem sechsten Zustand in seinen Ausgangszustand zurückgeht.

³ (ALL ABOUT CIRCUITS, 2024)

Anschließend auf die Erstellung der Zustandsübergangstabelle werden erneut für alle noch zu konstruierenden Komponenten die Karnaugh Maps aufgestellt (**Abbildung 4**), aus denen dann die booleschen Schaltfunktionen abgelesen werden. (**Tabelle 4**)

Die aus den booleschen Schaltfunktionen abgeleitete vollständige Logik Gatterschaltung für den Frequenzteiler 3:1 ist der **Abbildung 6** zu entnehmen.

Zählwert	b1	b2	b3	b'1	b'2	b'3	3:1
1	0	0	0	0	0	1	0
2	0	0	1	0	1	1	0
3	0	1	1	0	1	0	0
4	0	1	0	1	1	0	1
5	1	1	0	1	1	1	1
6	1	1	1	0	0	0	1

Tabelle 3. Zustandsübergangstabelle für den Frequenzteiler 3:1. Dargestellt sind die Binärrepräsentation der Zustandsübergänge für die jeweiligen Input Bits **b1**, **b2**, und **b3**, die Output Bits der Bit-Signal Umwandlungskomponenten **b'1**, **b'2**, und **b'3**, und der Output Bits der zwei Frequenzteiler 2:1 und 4:1. Alle Input und Output Bits werden über einen Zyklus von 8 Zustandsänderungen dargestellt. Verwendung eines 3-Bit-Gray-Code, welcher je Zustandsänderung nur jeweils 1 Bit ändert.

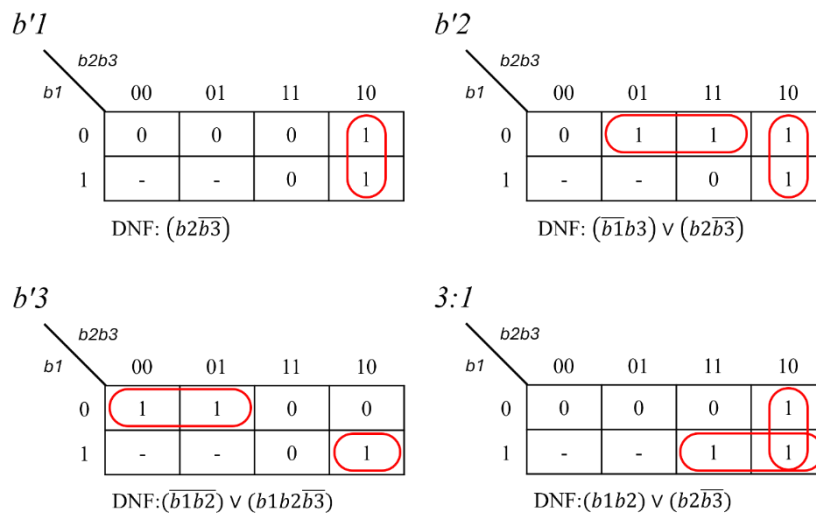


Abbildung 4. Karnaugh Maps für die Bit-Signal Umwandlungskomponenten **b'1**, **b'2**, und **b'3**, sowie die Output Komponente für den Frequenzteiler 3:1.

Schaltkreiskomponente	Boolesche Schaltfunktion
b'1	$(b2\bar{b}3)$
b'2	$(\bar{b}1b3) \vee (b2\bar{b}3)$
b'3	$(\bar{b}1b2) \vee (b1b2\bar{b}3)$
Out_3:1	$(\bar{b}1b2) \vee (b2\bar{b}3)$

Tabelle 4. Konstruierte Schaltkreiskomponenten und dazugehörige boolesche Schaltfunktionen für den Frequenzteiler 3:1. Die booleschen Schaltfunktionen sind aus den Karnaugh Maps aus **Abbildung 4** abgeleitet.

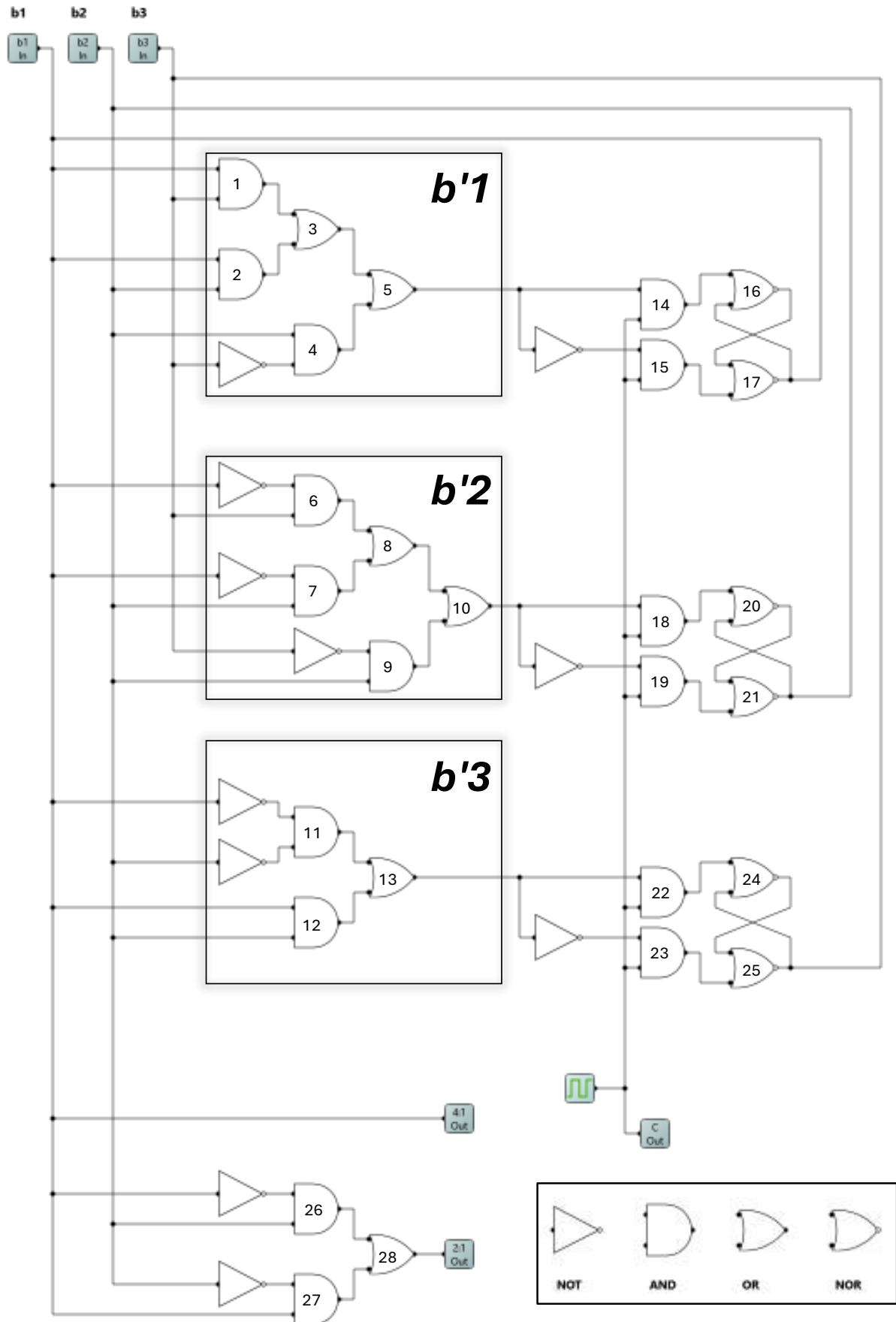


Abbildung 5. Vollständige logische Gatterschaltung für die Frequenzteiler 2:1 und 4:1.

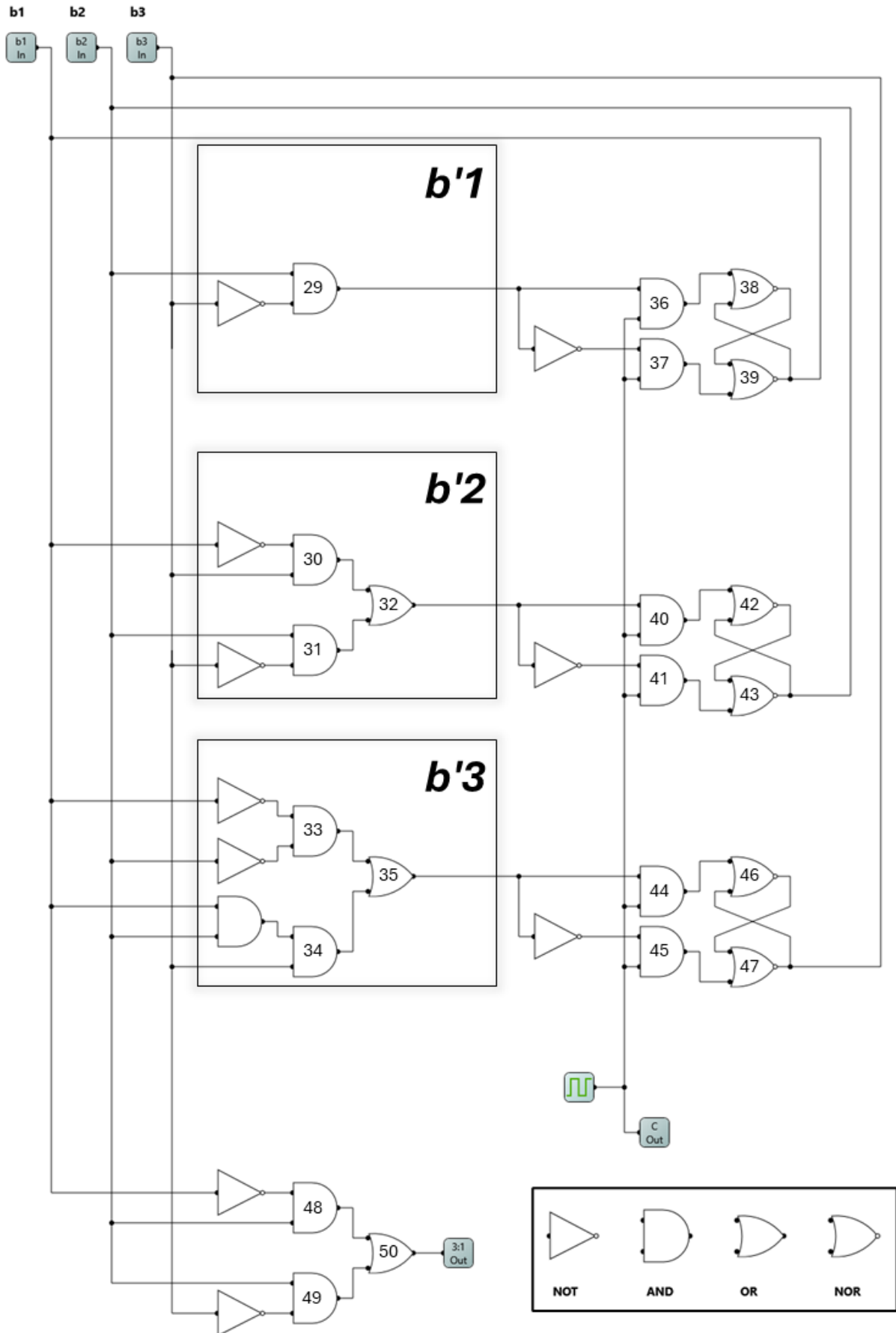


Abbildung 6. Vollständige logische Gatterschaltung für den Frequenzteiler 3:1.

3. COPASI Implementierung

3.1. Allgemeines Setup des chemischen Digitalcomputers innerhalb der COPASI Umgebung

Nachdem die logischen Gatterschaltungen für alle Frequenzteiler aufgestellt wurden, folgt daraufhin die Umsetzung dieser Schaltpläne in chemische Digitalcomputer. Für diese Umsetzung wird die Software COPASI verwendet. Die einzelnen Modelle der Frequenzteiler arbeiten innerhalb dieser Software ausschließlich über Massenwirkungskinetik. Somit ist der Fluss, welcher zwischen den verschiedenen Reaktionen fließt, direkt proportional zu der Anzahl der dazugehörigen Substratmoleküle.

Die jeweiligen Einheiten können vernachlässigt werden, da diese für die Funktionsweise des Systems keine Rolle spielen. Deshalb werden im Folgendem die Einheiten der angegebenen Konzentrationen ignoriert und weggelassen.

Für die vollständige Implementierung unserer logischen Schaltungspläne als chemischen Digitalcomputer, müssen die einzelnen Schalungskomponenten in ein chemisches Äquivalent umgewandelt werden. Dies erzielen wir mit der Nutzung von chemischen Spezies, welche repräsentativ für die einzelnen Inputs und Outputs stehen, die innerhalb unserer Schaltungen existieren. Es werden insgesamt zwei Spezies für jeweils jeden Input und jeden Output benötigt, da eine Spezies genau einen logischen Wert widerspiegeln soll. Dabei soll sich die Konzentration jeder dieser Spezies immer zwischen einem Wert von 0 und 1 bewegen. Gekennzeichnet werden die Spezies durch den ausschlaggebenden Namen der Schalungskomponenten, von welcher sie abstammen und einem Suffix, entweder T (True) oder F (False). Somit wird beispielsweise das Input Bit **b₁** durch die zwei Spezies **b₁T** und **b₁F** repräsentiert. Dabei fungiert die Spezies **b₁T** als logischer Wert 1 und die Spezies **b₁F** als logischer Wert 0. Welchen logischen Wert das Input Bit **b₁** zu einem gegebenen Zeitpunkt hat, wird dadurch bestimmt, welche der beiden Spezies eine Konzentration von annähernd 1 und 0 hat. Wichtig ist dabei, dass die Implementierung so umgesetzt wurde, dass nur eine der beiden Spezies zu einem gegebenen Zeitpunkt eine Konzentration von annähernd 1 haben kann und niemals beide.

3.2 Detaillierte Implementierung der Schaltpläne

Die Initialen Konzentrationen vieler Spezies werden ebenfalls auf entweder 0 oder 1 gesetzt. Bis auf wenige Ausnahmen werden für so gut wie alle Schalungskomponenten deren False Spezies auf 1 und deren True Spezies auf 0 gesetzt. Somit sind beispielsweise die initialen Konzentrationen für die drei Input Bits **b₁b₂b₃** = 000 jeweils wie folgt: **b_iT** = 0 und **b_iF** = 1. Grundlegend werden alle Spezies so initialisiert,

dass sie den logischen Werten entsprechen, welche diese äquivalent im Schaltplan im Zustand **b1b2b3** = 000 und Taktgeber = 0 annehmen würden. Die Funktionsweise der Output Spezies für die jeweiligen Frequenzteiler verhält sich ähnlich in dem Sinne, dass die Spezies repräsentativ für die logischen Werte 1 eine Konzentration von annähernd 1 haben, wenn das ausgegebene verzögerte Taktsignal dieses Frequenzteilers gleich logisch 1 sein soll.

Das implementierte COPASI Netzwerk kann in verschiedene funktionelle Bereiche zerlegt werden. Das binäre Eingabesignal (Taktsignal C) wird mit den Kompartimenten „Level 0“, „Level 1“, „Level 2“ und „Level 3“ simuliert. Diese bauen entsprechend ihrer aufsteigenden Nummerierung aufeinander auf und sorgen für eine Signalkaskade. Diese Signalkaskade soll in Form einer stetigen oszillierenden Reaktion simulieren und uns als Taktgenerator innerhalb unseres chemischen Digitalcomputers dienen.

Das zweite Kompartiment „Memory“ umfasst alle chemischen Spezies, die für Konstruktion unserer 6 Flipflop-Schaltungen benötigt werden. Diese speichern insgesamt 6 Bits (**b1-b6**), welche es dem chemischen Digitalcomputer erlauben den Rhythmus des Eingabesignals zu verfolgen und entsprechend mittels der verzögerten Ausgabesignale zu reagieren.

Der mit Abstand größte Bereich ist die „Main Reaction“. Bestandteile dieses Kompartiments erfüllen die Aufgabe die binären Eingabesignale **b1** bis **b6** in das jeweilige Binärmuster des nächstfolgenden Zustandes umzuwandeln. Insgesamt existieren sechs Cluster an Gattern, namentlich **b'1** bis **b'6** (**Abbildung 5, 6**), die jeweils drei Input Bits annehmen und ein einzelnes Ausgabesignal ausgeben, welches dann an einen 1 Bit RAM-Speichermodul weitergegeben wird.

Für jeden Frequenzteiler wurde eine Ausgabespezies erstellt, wobei die Spezies **O3T** die Ausgabe des normalen Taktsignales widerspiegelt. Für den Frequenzteiler 2:1 wird **R28T** als Ausgabespezies verwendet. Der Frequenzteiler 3:1 nutzt die Spezies **R40T**. Und für den Teiler 4:1 war es möglich die Spezies **b1T** wieder zu verwenden, da diese über alle acht Zustände hinweg das gleiche Binärmuster ausweist wie der zu erzielende Output des Frequenzteilers 4:1 (**Tabelle 1**).

Abschließend lässt sich bezüglich der Implementierung noch hinzufügen, dass bei der Umsetzung der logischen Gatter jeweils vier chemische Reaktionen je Gatter erstellt wurde, welche die jeweiligen Kombinationen der beiden Inputs widerspiegeln, und jeweils zwei autokatalytische Reaktionen je Gatter erstellt wurden, welche die Konzentrationsveränderungen beschleunigen sollen und verhindern sollen, dass ein Gatter ausstirbt.

4. Simulation und Resultate

Die vollzogenen Simulationen fanden über eine Dauer von 2000 Sekunden statt und erzeugten in einem Messpausen-Intervall von 0.05 Sekunden jeweils 40000 Messungen.

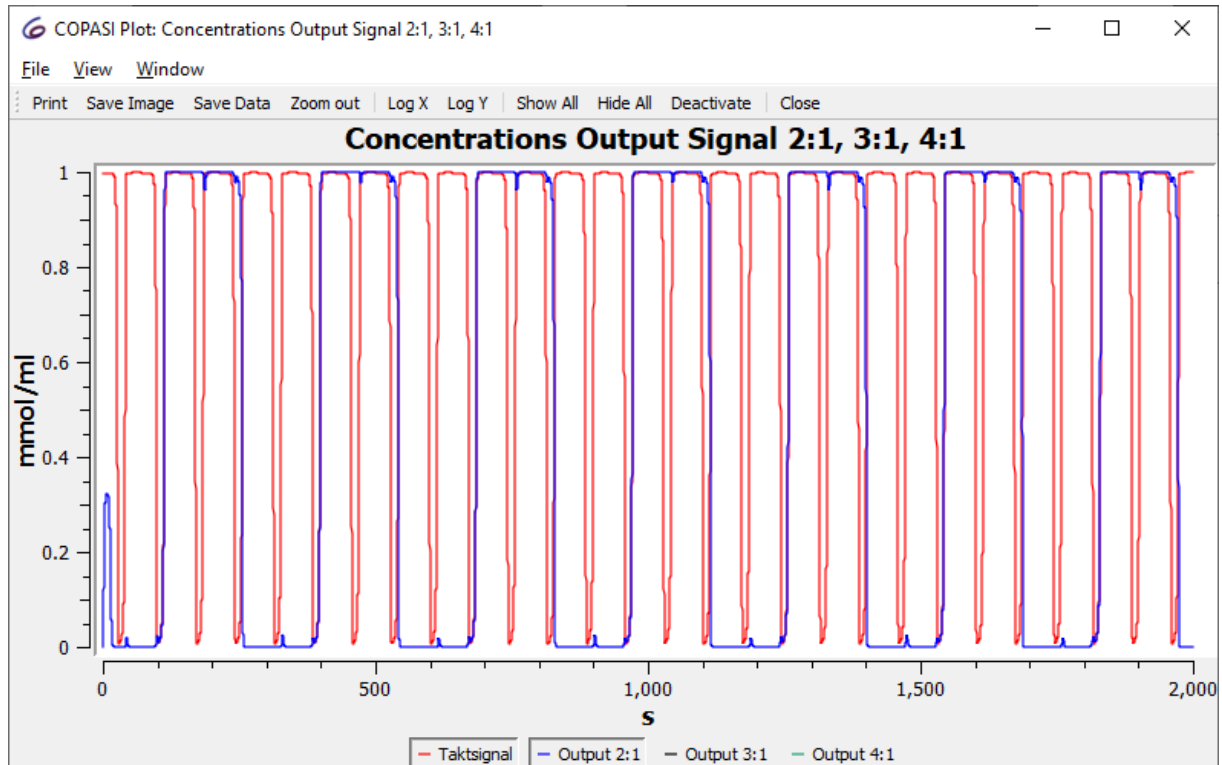


Abbildung 7. Simulation des 2:1 Frequenzteiler Outputs. Der Output der Frequenzteilers ist als blauer Kurvenverlauf abgebildet. Der Output des Taktsignales ist als roter Kurvenverlauf abgebildet.

Abbildung 7 zeigt den Ausgabe Plot für die Simulation des 2:1 Frequenzteilers. Wie erwartet oszilliert das Ausgabesignal des Teilers mit 2:1-facher Verzögerung verglichen mit dem Taktsignal. Somit deutet das zu sehende Ergebnis der Simulation auf eine fehlerfreie Funktionsweise der beteiligten Schaltungselemente und den korrespondierenden chemischen Spezies hin. Gut zu erkennen sind die fast rechteckigen Schwingungen der einzelnen Frequenzen. Dahingehend kann man annehmen, dass die Implementierung dieses Frequenzteiler erfolgreich war. Anmerken muss man dennoch, dass innerhalb des Ausgabesignales Artefakte zu vermerken sind, auch wenn diese von kleiner Ausprägung sind. Diese treten in rhythmischen Abständen, synchron zu dem Ausgabesignal des 2:1 Teilers auf.

Das Ergebnis für die Simulation des 3:1 Frequenzteilers ist in Abbildung 8 einzusehen. Ähnlich wie bei der vorherigen Simulation ist auch hier das Ausgabesignal der verzögerten 3:1 Frequenz zum einen annähernd rechteckig und zum anderen in einer 3:1-Verzögerung verglichen mit dem initialen Taktsignal. Demnach lässt sich auch hier von einer erfolgreichen Implementierung des Frequenzteilers ausgehen. Ebenfalls zu erkennen sind hier erneut die bereits in Abbildung 7 erwähnten Artefakte.

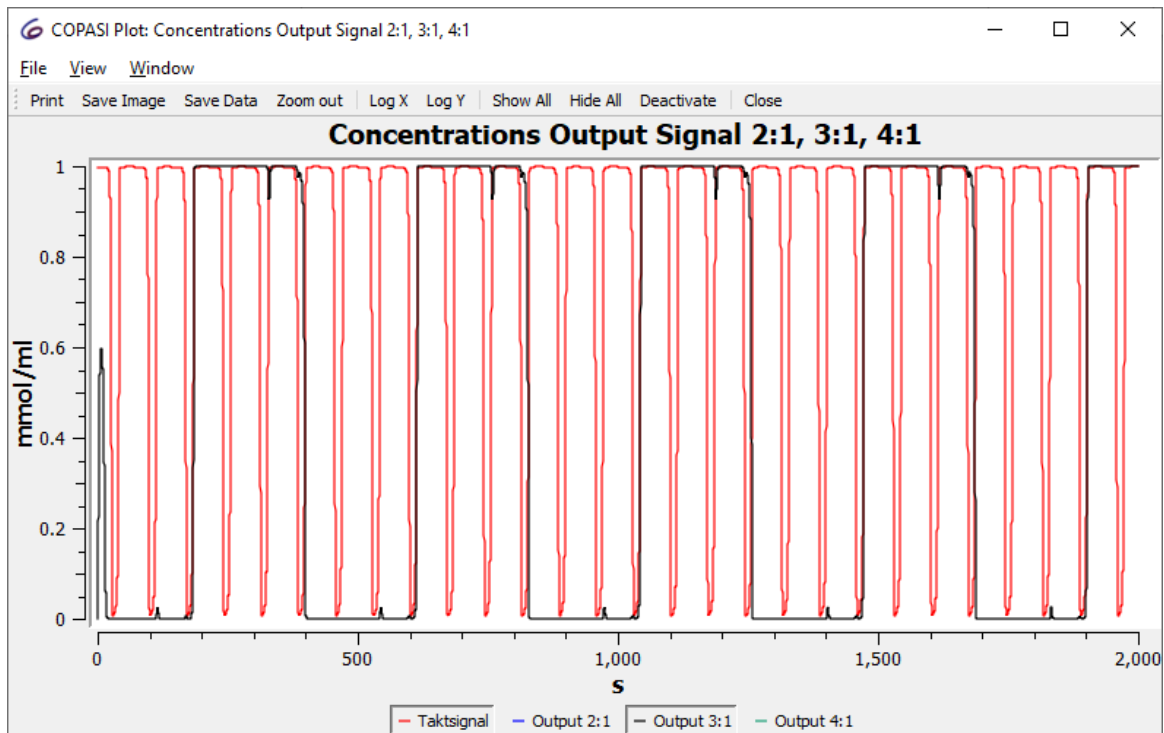


Abbildung 8. Simulation des 3:1 Frequenzteiler Outputs. Der Output der Frequenzteilers ist als schwarzer Kurvenverlauf abgebildet. Der Output des Taktsignales ist als roter Kurvenverlauf abgebildet.

Es ist ein klarer, verhältnismäßig glatter Übergang zwischen den binären Phasen erkennbar. Auch im Vergleich zum Originalsignal ist erkennbar, dass das Verhältnis 4:1 erfolgreich erzeugt und gehalten wird (**Abbildung 9**). Wie auch schon bei beiden Simulationen davor, lässt sich hier eine erfolgreiche Implementierung interpretieren.

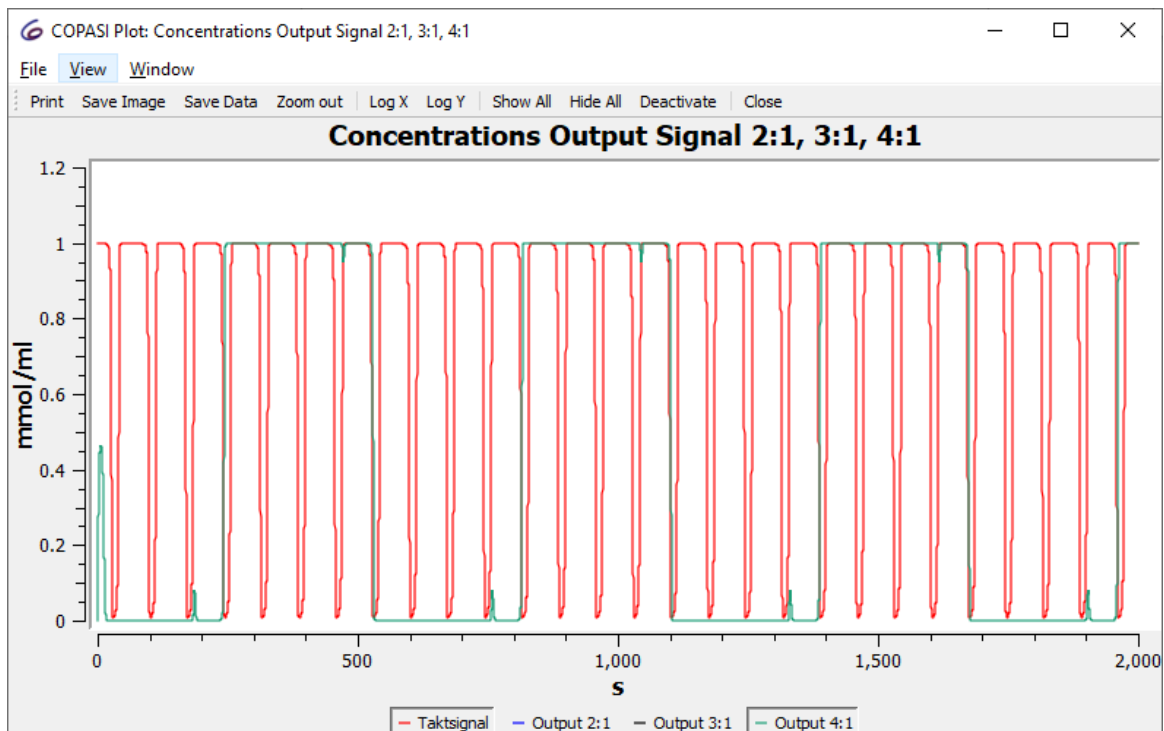


Abbildung 9. Simulation des 4:1 Frequenzteiler Output. Der Output der Frequenzteilers ist als türkiser Kurvenverlauf abgebildet. Der Output des Taktsignales ist als roter Kurvenverlauf abgebildet.

5. Fazit

Zusammenfassend konnten die durchgeführten Simulationen zeigen, dass die Frequenzteiler 2:1, 3:1 und 4:1 entsprechend der Anforderungen funktionieren.

Des Weiteren konnte gezeigt werden, dass es mit dem erstellten Modell auch auf molekularer Ebene möglich ist ein Signal zu erhalten, es zu strecken und auf messbare Art und Weise wieder auszugeben.

Die Umsetzung ermöglicht es nun die Vorteile eines analogen, chemischen Digitalcomputers auf das Problem anzuwenden und so die bestehenden Grenzen der digitalen Verarbeitung zu überwinden. Somit wurde die Aufgabenstellung erfolgreich umgesetzt, indem die drei Frequenzteiler problemlos als chemische Digitalcomputer in COPASI implementiert wurden.

Auch ist denkbar, dass für jedes beliebige Verhältnis Frequenzteiler erzeugt werden können. Dies müsste jedoch zukünftig in weiteren Versuchen bestätigt werden.

Literaturverzeichnis

ALL ABOUT CIRCUITS. (17. Juni 2024). Von

<https://www.allaboutcircuits.com/textbook/digital/chpt-8/logic-simplification-karnaugh-maps/> abgerufen

Hinze, T. (2013). *Computer der Natur*. Cottbus.

Hinze, T. (2024). *Vorlesung - Molekulare Algorithmen*. Von <https://users.fmi.uni-jena.de/~hinze/mr/themen.html> abgerufen

Wikipedia - Bioinformatik. (02. Juli 2024). Von <https://de.wikipedia.org/wiki/Bioinformatik> abgerufen